

Fu, Junchen (2026) *Adapting multimodal foundation models for recommendation.* PhD thesis.

https://theses.gla.ac.uk/86308/

Copyright and moral rights for this work are retained by the author

A copy can be downloaded for personal non-commercial research or study, without prior permission or charge

This work cannot be reproduced or quoted extensively from without first obtaining permission from the author

The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given

Enlighten: Theses
https://theses.gla.ac.uk/
research-enlighten@glasgow.ac.uk

Adapting Multimodal Foundation Models for Recommendation

Junchen Fu

Submitted in fulfilment of the requirements for the Degree of
Doctor of Philosophy

School of Computing Science
College of Science and Engineering
University of Glasgow



June 2026

Abstract

Multimodal foundation models (MFMs) now show strong representation and generation capabilities. Adapting them to recommender systems is a natural and increasingly central research direction, since items often contain rich multimodal signals such as text, images, videos, and audio. Yet MFMs are designed for general representation and generation tasks, making specific recommendation-oriented adaptation a key challenge. This thesis addresses this challenge from two connected perspectives: representation adaptation and generation adaptation.

The first perspective studies end-to-end adaptation of continuous and discrete representations. Continuous representation adaptation jointly trains the MFM representation-learning pipeline and recommender through real-valued representations, making efficiency a central challenge. IISAN uses cached decoupled parameter-efficient fine-tuning to reduce computation overhead. Building on IISAN, we investigate adaptation depth and breadth. IISAN-Versa extends it to deeper, larger, asymmetric multimodal encoders, particularly pairing a large LLM-based text encoder with a smaller vision encoder. CROSSAN broadens it to multiple modality-specific foundation models through cross-modal gated side adaptation. We then consider unified single-tower embedding models, whose general-domain capabilities pose a distinct challenge: existing multi-tower methods often become shallow side adapters and struggle with deep-layer adaptation. Stresa addresses this limitation through stream-aware residual adaptation. For discrete representation adaptation, jointly learning semantic-ID construction and the generative recommender introduces challenges, particularly non-differentiability and codebook collapse. DIGER tackles these using differentiable semantic IDs and an exploration mechanism that promotes codebook utilisation while maintaining end-to-end adaptation performance.

The second perspective addresses a gap in the literature: recommendation-oriented generation adaptation remains underexplored from platform and user perspectives. On the platform side, although generation can complete missing modalities for item presentation and recommendation model training, its effectiveness in recommendation settings lacks evaluation. MM-PCBench addresses this gap by benchmarking MFMs on multimodal product content completion. On the user side, generated content can enter the recommendation ecosystem, yet little is known about optimising it for engagement. LLMPopcorn investigates the popularity of MFM-generated content and explores LLMs for popularity-oriented micro-video creation. Across seven works, this thesis studies MFM adaptation across representation and generation.

Contents

Abstract **i**

Acknowledgements **xii**

Declaration **xiv**

1 Introduction **1**

1.1 Terminology and Scope 3
1.2 Thesis Statement 4
1.3 Research Questions 5
1.4 Research Progression and Connections 6
1.5 Contributions 8
1.5.1 Recommendation-Oriented Representation Adaptation 8
1.5.2 Recommendation-Oriented Generation Adaptation 9
1.6 Source Publications, Thesis Format and Author Contributions 9
1.6.1 Source Material and Adaptation to the Thesis 9
1.6.2 Contribution Statement 10
1.6.3 Publication-by-Publication Outline 10
1.6.4 Changes Made for the Minor Corrections 14
1.7 Thesis Organisation 15

2 Background and Related Work **16**

2.1 Recommender Systems and Content-Rich Items 17
2.2 Multimodal Foundation Models 18
2.3 Multimodal Recommendation Systems 20
2.4 Continuous Representation Adaptation for Recommendation 22
2.5 Discrete Representation Adaptation for Generative Recommendation 24
2.6 Generation Adaptation for Recommender Systems 26
2.7 Datasets, Biases, and Evaluation Metrics 28

I Recommendation-Oriented Representation Adaptation 33

3 IISAN 34

3.1 Introduction . . . 34
3.2 Related Work . . . 39
3.3 Methodology . . . 40
3.3.1 IISAN . . . 41
3.3.2 New Efficiency metric: TPME . . . 43
3.4 Analysis of Efficiency . . . 44
3.4.1 Training-time Efficiency . . . 44
3.4.2 Parameter Efficiency . . . 45
3.4.3 GPU Memory Efficiency . . . 45
3.5 Experiment Setup . . . 48
3.6 Experiment . . . 48
3.6.1 Efficiency-Performance Balance (RQ3.1) . . . 49
3.6.2 Robustness Evaluation (RQ3.2) . . . 50
3.6.3 Ablation Study (RQ3.3) . . . 50
3.6.4 Multimodality vs. Unimodality (RQ3.4) . . . 52
3.7 Conclusion and Future Work . . . 53

4 IISAN-Versa 54

4.1 Introduction . . . 54
4.2 Related Work . . . 57
4.3 Methodology . . . 59
4.3.1 Overview . . . 60
4.3.2 Problem Formulation . . . 60
4.3.3 Intra-model Side Adapted Network . . . 62
4.3.4 Inter-model Side Adapted Network . . . 62
4.3.5 LayerDrop . . . 63
4.3.6 Caching Strategies . . . 64
4.3.7 Training Objectives . . . 64
4.4 Efficiency Analysis . . . 65
4.5 Experiment Setup . . . 67
4.5.1 Dataset . . . 67
4.5.2 Multimodal Text Generation . . . 67
4.5.3 Performance Evaluations. . . . 68
4.5.4 Implementation Details. . . . 69
4.6 Experiment . . . 69
4.6.1 IISAN-Versa Performance with larger text encoders (RQ4.1) . . . 70

4.6.2 Scaling Effect (RQ4.2) . . . 71
4.6.3 Performance on Public Benchmark (RQ4.3) . . . 74
4.6.4 Efficiency Analysis (RQ4.4) . . . 75
4.6.5 Ablation Studies (RQ4.5) . . . 79
4.6.6 Performance on Multimodal Text (RQ4.6) . . . 81
4.6.7 Hyper-parameter Sensitivity Analysis (RQ4.7) . . . 82
4.7 Conclusion and Future Work . . . 83

5 CROSSAN 84
5.1 Introduction . . . 84
5.2 Related Work . . . 86
5.3 Preliminary study: Cross or Independent? . . . 88
5.4 Theoretical Analysis . . . 88
5.4.1 Mutual Information as an Evaluation Metric . . . 88
5.4.2 Independent vs. Cross-Modal Adapters for Item Representation . . . 89
5.5 Methodology . . . 90
5.6 Experiment Setup . . . 92
5.7 Experiment . . . 94
5.7.1 Effectiveness Evaluation (RQ5.1) . . . 94
5.7.2 Efficiency Evaluation (RQ5.2) . . . 96
5.7.3 Ablation Study (RQ5.3) . . . 98
5.7.4 Hyperparameter Analysis (RQ5.4) . . . 100
5.8 More evaluations . . . 102
5.9 Conclusion . . . 103

6 Stresa 104
6.1 Introduction . . . 104
6.2 Related Work . . . 108
6.3 Method . . . 109
6.3.1 Problem Formulation . . . 109
6.3.2 Overview of Stresa . . . 110
6.3.3 Stream-aware Hidden-Adapter Fusion (SHAF) . . . 111
6.3.4 Residual Stream Adapter (ReSA) . . . 113
6.3.5 Complexity and Deployment . . . 114
6.3.6 Structural Observations on SHAF and ReSA . . . 114
6.4 Experimental Setup . . . 115
6.5 Experimental results . . . 116
6.5.1 Comparison with Side Adapters (RQ6.1) . . . 116
6.5.2 Comparison with SOTA baselines (RQ6.2) . . . 118

6.5.3 Ablation Study (RQ6.3) . . . 120
6.5.4 Visualisation (RQ6.4) . . . 120
6.6 Efficiency . . . 121
6.7 Conclusion and Future Work . . . 121

7 DIGER 122
7.1 Introduction . . . 122
7.2 Related Work . . . 125
7.3 Preliminaries . . . 128
7.4 Our Method: DIGER . . . 128
7.4.1 DRIL: Differentiable Semantic ID with Exploratory Learning . . . 129
7.4.2 Two Uncertainty Decay Strategies . . . 130
7.4.3 Model Training . . . 132
7.5 Experimental Setup . . . 133
7.6 Experimental results . . . 134
7.6.1 Comparison with Conventional and Differentiable SID Training (RQ7.1) 134
7.6.2 Comparison with SOTA baselines (RQ7.2) . . . 135
7.7 Ablation Study (RQ7.3) . . . 136
7.8 Semantic ID Dynamics and Stability (RQ7.4) . . . 138
7.9 Theoretical Analysis . . . 140
7.10 Derivation of the optimal σ . . . 141
7.11 Conclusions . . . 142

II Recommendation-Oriented Generation Adaptation 143

8 MMPCBench 144
8.1 Introduction . . . 144
8.2 Related Work . . . 146
8.3 Methodology . . . 148
8.3.1 Problem Formulation . . . 148
8.3.2 MMPCBench . . . 150
8.3.3 GRPO Training . . . 152
8.4 Experiments . . . 153
8.4.1 Implementation Details . . . 153
8.4.2 Benchmark Analysis . . . 154
8.4.3 GRPO Results . . . 158
8.5 Conclusion and Future Work . . . 159
8.6 Additional Details . . . 160

8.6.1 More Results of GRPO . . . 160
8.6.2 More Analysis of the Results . . . 161
8.6.3 Additional Implementation Details for Text→Image . . . 163

9 LLMPopcorn **169**
9.1 Introduction . . . 169
9.2 The Proposed *LLMPopcorn* . . . 171
9.2.1 Problem Formulation . . . 171
9.2.2 Pipeline . . . 172
9.2.3 Prompt Enhancement . . . 172
9.3 Experiment Setup . . . 173
9.4 Popular Micro-videos Generation (RQ9.1) . . . 175
9.5 Popularity Enhancement (RQ9.2) . . . 176
9.6 Benchmarking Models (RQ9.3) . . . 177
9.7 Visualisation . . . 178
9.8 Conclusion . . . 179

10 Conclusion and Future Work **180**
10.1 Conclusion . . . 180
10.1.1 RQ1: Efficient Continuous Representation Adaptation . . . 180
10.1.2 RQ2: Scaling to Larger and Asymmetric Encoders . . . 180
10.1.3 RQ3: Adapting Multiple Modality-Specific Foundation Models . . . 181
10.1.4 RQ4: Deep Adaptation of Unified Single-Tower Models . . . 181
10.1.5 RQ5: Joint Optimisation of Discrete Representations . . . 181
10.1.6 RQ6: Platform-Side Missing-Modality Completion . . . 182
10.1.7 RQ7: User-Side Popularity-Oriented Generation . . . 182
10.1.8 Overall Answer to the Thesis Question . . . 182
10.2 Limitations . . . 183
10.3 Future Work . . . 184
10.3.1 Unified Representation and Generation Adaptation . . . 184
10.3.2 Online and Long-Term Evaluation . . . 185
10.3.3 Continual and Cache-Aware Adaptation . . . 185
10.3.4 Broader Multimodal Architectures and Modalities . . . 186
10.3.5 Trustworthy Recommendation-Oriented Generation . . . 186
10.3.6 Towards Recommendation-Oriented Foundation Models . . . 187

List of Tables

3.1 Efficiency Comparison of FFT and PEFT 44
3.2 Dataset Description 47
3.3 Comparisons of efficiency-performance balance 47
3.4 Ablation study for IISAN on Scientific Dataset 51
3.5 LayerDrop in IISAN on Scientific Dataset 51
3.6 Implementation of SANB (Side Adapted Network Block) on Scientific Dataset 52
3.7 Multimodality vs. Unimodality 53

4.1 Selected layers S for various total depths L (with $L^v = 6$) 63
4.2 Dataset Description 67
4.3 Description of the pre-trained models used in this research 69
4.4 Performance (%) comparison across methods on three datasets 71
4.5 IISAN-Versa Encoders Scaling 72
4.6 Performance comparison of IISAN-Versa on a large dataset over 1M interactions 72
4.7 Performance comparison of IISAN-Versa on a multimodal public benchmark MicroLens 73
4.8 Efficiency comparison with optimal SASRec and IISAN-Versa 74
4.9 Ablation study for IISAN-Versa and IISAN-VA on Scientific Dataset 78
4.10 LayerDrop in symmetrical IISAN-Versa on Scientific Dataset 78
4.11 Asymmetrical IISAN-Versa LayerDrop ablation 79
4.12 Performance on Multimodal Text 80

5.1 Performance comparison between INDSAN (Independent Side Adapter) and CROSSAN (Cross-Modal Side Adapter) 88
5.2 Dataset Description 93
5.3 CROSSAN performance on MicroLens datasets 94
5.4 State-of-the-art comparison on MicroLens 96
5.5 Efficiency Comparison 96
5.6 Ablation Study on Fusion Method 97
5.7 Cross-modal vs. independent-modal ablation 97
5.8 Top-K experts for CROSSAN 99

5.9 Performance (%) comparison across methods on three datasets 100
5.10 Performance comparison on MicroLens-100K 101
5.11 Performance Comparison of Independent vs. Cross-Modal Architectures with Cross-Attention Fusion on the MicroLens-100K Dataset 101
5.12 Performance comparison of Independent and Cross-modal methods 101

6.1 Dataset Description 115
6.2 Main results under frozen-backbone adaptation 117
6.3 Main results on three datasets 117
6.4 Ablation on Qwen3-VL-8B 118
6.5 Ablation on Identity Preservation 119
6.6 Ablation on stream-aware residual 119
6.7 Efficiency comparison between Stresa and Side Adapter 120

7.1 Dataset statistics 132
7.2 Comparison of conventional two-stage generative recommendation and training with differentiable semantic IDs 134
7.3 Overall performance comparison between the baselines on generative recommender models on three datasets 135
7.4 Ablation of DIGER components on B-Shop Dataset 137
7.5 Effect of codebook capacity (K) and SID length (m) on B-Shop Dataset 138

8.1 Recommendation Benchmark Category Description 149
8.2 Average Text Quality Metrics (mean $\pm$ std) Across All Categories for I$\rightarrow$T tasks on Content Quality Benchmark 154
8.3 Image quality metrics (mean $\pm$ std) across all categories 154
8.4 Category-wise Average Text Quality Metrics (mean $\pm$ std) Across All Models for Image-to-Text tasks on Content Quality Benchmark 154
8.5 Category-wise average image quality metrics (mean $\pm$ std) across all models . 154
8.6 Human evaluation results (5 annotators, 54 items, and 9 categories) 155
8.7 Benchmark of MLLMs on Image-to-Text (I$\rightarrow$T) and Text-to-Image (T$\rightarrow$I) . . . 156
8.8 Summary of paired Wilcoxon signed-rank tests (n=9 datasets, α=0.05) 157
8.9 Average Text Quality Metrics (mean $\pm$ std) Across All Categories for GRPO Training for Qwen2.5-VL-7B-Instruct 158
8.10 Average Image Quality Metrics (mean $\pm$ std) Across All categories for GRPO Training 159
8.11 Text Quality Metrics (mean $\pm$ std) by Method and Category 161
8.12 Text Quality Metrics (mean $\pm$ std) by Model and Category 164
8.13 Image quality metrics (mean $\pm$ std) by model and category 165

8.14 Performance comparison for model VBPR, combining Image-to-Text and Text-to-Image 166
8.15 Performance comparison for model BM3, combining Image-to-Text and Text-to-Image 167
8.16 Performance comparison for model FREEDOM, combining Image-to-Text and Text-to-Image 168

9.1 Comparison of popularity across different categories and LLMs 174
9.2 Comparison of popularity medians for video generation tasks (using CogVideoX-5b as the video generator) 174
9.3 Comparison of popularity for Prompt Enhancement's win rates across different LLMs 175
9.4 User study for Prompt Enhancement vs. Basic LLMPopcorn prompt 175
9.5 Different LLMs' Win Rate comparisons across concrete and abstract datasets . 176
9.6 Different video generation models' Win Rate comparisons across concrete and abstract Datasets 177

List of Figures

3.1 Comparisons among Full Fine-tuning (FFT), Embedded PEFT and Decoupled PEFT for feature representation learning 35
3.2 An Overview of the IISAN for sequential recommendation 38
3.3 Caching strategies comparison 38
3.4 Performance comparisons between FFT and IISAN with different multimodal backbones on Scientific dataset 49

4.1 Text and visual transformer model sizes 56
4.2 An Overview of the IISAN-Versa for Sequential Recommendation 61
4.3 An Overview of Caption Generation 68
4.4 Scaling effect of IISAN-Versa's text encoder on Scientific dataset with Hit@10 metric 70
4.5 Comparison of GPU Memory Usage and Performance Between SASRec and IISAN-Versa 74
4.6 GPU Memory and Training time comparison 75
4.7 On-Disk Storage Requirements for Caching Strategy, by Model Size 76
4.8 GPU memory and Training time usage for IISAN-Versa with different text encoders 76
4.9 Visualisation of inter-modality gates in IISAN-Versa 78
4.10 Learning rate and Embedding Dimension of IISAN-Versa 82

5.1 Independent vs. Cross-modal Side Adapter 86
5.2 CROSSAN Overview 90
5.3 Optimal Embedding dimension for IISAN-E and CROSSAN 95
5.4 The heatmap visualisation of CROSSAN's gates, where deeper colours indicate higher values 98
5.5 CROSSAN embedding dimension and learning-rate adaptation 99

6.1 Conventional Side Adapter vs. Stresa 105
6.2 Overview of the proposed stream-aware fusion framework 108
6.3 Stresa performance across adapted layers 118

6.4 Layer-wise representation similarity 121

7.1 Conventional vs. Generative recommendation with Differentiable SID 123
7.2 Comparison of DIGER and STE on B-Shop Dataset 125
7.3 Comparison of STE vs. DIGER 127
7.4 Two uncertainty decay strategies for exploration–exploitation 127
7.5 Hyperparameter analysis on B-Shop and I-Shop 137
7.6 Semantic ID assignment dynamics 138
7.7 Code usage across quantisation layers 139

8.1 Missing-Modality Completion for E-Commerce Products 145
8.2 Overview of MMPCBench 148
8.3 Missing-modality completion framework 148
8.4 Content quality radar charts 153
8.5 MLLM completion-method heatmap 157
8.6 Reward Curves of different tasks 160
8.7 Qualitative missing-modality completion cases 162

9.1 Overview of the LLMPopcorn pipeline 171
9.2 An overview of the Prompt Enhancement (PE) process 172
9.3 Predicted popularity distributions for different LLMs 175
9.4 Video examples from LLMPopcorn, shown from left to right 178
9.5 Video examples from LLMPopcorn with Prompt Enhancement, shown from left to right 179

Acknowledgements

My PhD journey has been a wonderful and transformative experience, filled with success, frustration, persistence, failure, and much more. There were difficult moments, especially when facing repeated setbacks, but there were also deeply rewarding moments when the results finally came together and the methods I developed showed the potential to benefit the broader research community. These rewarding experiences have strengthened my belief in the value of meaningful research. I hope to dedicate my future research career to work that benefits not only the research community, but also wider society.

Throughout this journey, I have grown from a junior student into an independent researcher. This growth would not have been possible without the support, guidance, and encouragement of many people, to whom I would like to express my heartfelt gratitude.

First and foremost, I would like to thank my primary supervisor, Prof. Joemon M. Jose. Joemon has been a remarkably supportive supervisor, mentor, and friend. Thank you for your constant support, patience, and consideration throughout my PhD. Whenever I encountered difficulties, whether academic or personal, Joemon was always there to help. When I faced failures, he encouraged me and helped me find ways forward. One of the most important lessons I have learned from Joemon is that research should always be driven by strong research questions. A good research paper must begin with good research questions, and this principle will continue to guide me throughout my research career.

Second, I would like to express my sincere thanks to two of my mentors, Dr. Alexandros Karatzoglou and Dr. Ioannis Arapakis. Alex is an extremely insightful researcher with a sharp intuition and clear vision for research directions. He consistently helped guide my work to the next level. Ioannis is a very supportive and responsible mentor who provided important resources for our group, enabling us to pursue more advanced research. I am very grateful to both of them.

Third, I would like to express my sincere gratitude to Prof. Zhaochun Ren for his support during my research visit to Leiden University and ever since. During my visit, he shared many insightful perspectives on the research field and encouraged me to shift my research direction from traditional sequential recommendation to the generative recommendation paradigm. This change meant a great deal to me and has since proved to be the right choice. I would also like to thank Prof. Suzan Verberne, from whom I learned a great deal about academic writing and

formulation. She has been very kind and supportive.

Fourth, I would like to thank Dr. Kleomenis Katevas for his support during my internship at Brave. I am also grateful to Dr. Sofía Celi, Dr. Vandana Rajan, Prof. Hamed Haddadi, and everyone else at the Brave London office for making my internship at the end of my PhD such a wonderful experience.

Fifth, I would like to thank my girlfriend, Qing Jiang, for her unwavering support. During the most difficult period of my PhD, when I faced rejection after rejection for an entire year, she continued to believe in me, encourage me, and help me persevere. I am also deeply grateful to her for maintaining our long-distance relationship between the UK and China with such patience, trust, and care.

Sixth, I would like to thank all my collaborators, colleagues, and friends for their help, companionship, and the enjoyable time we spent together. I am grateful to my colleagues from GAIR-Lab, including Xuri Ge, Kaiwen Zheng, Wenhao Deng, and many others. I would also like to thank my roommate, Chengli Zhai, for his help throughout these years. My sincere thanks also go to the warm and kind people from LIACS, including Yumeng Wang, David Lindevelt, Farrukh Baratov, Jujia Zhao, I-Fan Lin, Shuaiqun Pan, and many others. I would also like to thank Yongxin Ni, Youhua Li, Hanwen Du, Jiaqi Zhang, and Yu Song for discussing many interesting ideas with me. I am also grateful to everyone in the "Glasgow Board Game WeChat Group" for all the fun and memorable moments, and to my friends at the UvA IRLab for their warm welcome and many insightful academic discussions, including Yongkang Li, Jingwei Kang, and many others. In addition, I am deeply grateful for the opportunity to have collaborated with many excellent and hardworking MSc/Undergraduate students. In addition, there are many friends whose names I have not mentioned individually, but to whom I am deeply grateful for their companionship, support, and help throughout this journey.

Seventh, I would like to thank my secondary supervisor, Dr. Ke Yuan, and Dr. Paul Henderson for providing so much useful feedback during my APR.

Eighth, I would like to thank the many people who shaped my academic journey before my PhD. I am grateful to Prof. Zhongxiang Chen and Dr. Liubin Zhang, who first introduced me to machine learning; Prof. Zhaohui Qi, who supervised my undergraduate thesis and encouraged me to publish my first research paper; and Prof. Yufei Tao, whose course at CUHK helped me appreciate the beauty of machine learning and whose "small but sweet" algorithmic philosophy continues to inspire my research. I am especially grateful to Prof. Fajie Yuan, my supervisor at Westlake University, who taught me to think big, pursue impactful research, and encouraged me to apply to Joemon's group in Glasgow. Without them, I might never have chosen to pursue a PhD, and I am deeply grateful for their support.

Last but not least, I would like to thank my family for their unconditional support. I am deeply grateful that they have always respected my choices, supported my decision to pursue my own career path, and encouraged me throughout this journey.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work, under the supervision of Professor Joemon M. Jose.

Chapter 1

Introduction

Recommender systems have become essential infrastructure for modern digital platforms, alleviating information overload by recommending personalised items from large-scale user interaction histories and rapidly expanding pools of user-generated and AI-generated content. Modern recommender systems increasingly operate in content-rich environments, where products, micro-videos, music, news articles, and services are described and presented through text, images, videos, audio, attributes, and their combinations. Conventional recommender models, however, have typically been built around ID-based interaction embeddings, which makes it difficult to fully exploit the rich semantic information contained in complex textual, visual, and multimodal item content (X. He et al., 2017; Hidasi, Karatzoglou, Baltrunas, & Tikk, 2015). Recent multimodal foundation models (MFMs), including large language models, vision transformers, vision–language models, and multimodal large language models, have demonstrated strong transferable representation and generation capabilities (Alayrac et al., 2022; Brown et al., 2020; Devlin, Chang, Lee, & Toutanova, 2018; Dosovitskiy, Beyer, et al., 2020; Jia et al., 2021; C. Li et al., 2024; J. Li, Li, Savarese, & Hoi, 2023; Y. Liu et al., 2019; Radford, Kim, Hallacy, et al., 2021). These capabilities make MFMs a natural interface between multimodal item content and recommender systems: they can encode heterogeneous item content into transferable representations and generate new multimodal content for recommendation-oriented applications.

Their relevance to recommendation arises from two capabilities. First, MFMs can encode item text, images, videos, audio, and multimodal inputs into continuous representations in a real-valued embedding space for personalised ranking and sequential prediction. Second, MFMs can generate multimodal content, including product descriptions, captions, images, prompts, and videos. These generative capabilities support recommendation from both platform-side and user-side perspectives. On the platform side, MFMs can complete incomplete product records by inferring missing modalities from available ones, improving item presentation and providing additional content signals for recommendation-model training. On the user side, especially on recommendation-driven social media platforms, MFMs can assist users in creating videos and other content intended to attract engagement and enter the recommendation pool (HaCohen et

al., 2024; Kondratyuk et al., 2023; Ramesh, Dhariwal, Nichol, Chu, & Chen, 2022; Ramesh et al., 2021; Z. Yang et al., 2024; P. Zhou et al., 2024).

The direct use of MFMs creates a recommendation-oriented adaptation gap. MFMs are usually pretrained with broad objectives such as masked language modelling, contrastive image–text alignment, next-token prediction, visual instruction following, retrieval, or generative likelihood. Recommender systems optimise task-specific objectives such as next-item prediction and personalised ranking. A representation learned for semantic alignment may organise items according to general visual or textual similarity. A recommendation representation must organise content according to the distinctions that drive user choice, sequence transitions, and ranking outcomes. Two products with close visual semantics may differ in function, price, brand, or usage context. Two micro-videos with similar topics may receive different engagement because of pacing, editing style, audio, or creator-side presentation. Recommendation-oriented adaptation is required to make MFM-derived item representations and generated output serve personalised objectives.

The representation adaptation gap is also a systems problem. Full fine-tuning can align foundation encoders with downstream recommendation losses, but it becomes computationally expensive when item catalogues are large, user histories repeatedly reuse the same items, and multiple raw modalities must be processed. Parameter-efficient fine-tuning methods such as adapters, LoRA, and BitFit reduce the number of trainable parameters (Houlsby et al., 2019; E. J. Hu et al., 2021; Zaken, Ravfogel, & Goldberg, 2021), and recent recommender-system studies have adopted PEFT for large-model adaptation (Bao et al., 2023; Fu, Yuan, et al., 2024; Peng, Burns, Chen, Parthasarathy, & Ning, 2023). However, adapter- and LoRA-based embedded PEFT methods still require backpropagation through the large backbone computation graph, leaving training time and GPU memory as major bottlenecks. This thesis addresses this efficiency challenge through a series of decoupled PEFT methods, including IISAN, IISAN-Versa, CROSSAN, and Stresa. These methods separate trainable side networks from frozen multimodal foundation backbones, enabling cached hidden-state reuse and reducing the practical cost of end-to-end representation adaptation. The representation adaptation gap also appears in the emerging generative recommendation paradigm (Rajput et al., 2023). In this setting, items are represented by discrete semantic identifiers, which are usually derived by encoding multimodal item content with MFMs, mapping it into real-valued embeddings, and then applying quantisation. Directly adapting this semantic-ID construction pipeline is challenging because quantisation blocks gradient propagation, while directly adopting the default straight-through estimator (Bengio, Léonard, & Courville, 2013) can lead to codebook collapse. To address this bottleneck, this thesis further develops DIGER, which introduces differentiable semantic identifiers with exploratory codebook learning. DIGER enables end-to-end adaptation of semantic IDs for generative recommendation while improving the stability and expressiveness of semantic identifier learning.

The generation capability of MFMs introduces a second, application-driven adaptation problem. Unlike representation adaptation, which mainly focuses on optimising MFM representations toward recommendation objectives, generation adaptation concerns how MFM-generated content can support different applications within recommendation platforms. This thesis studies this problem from both platform-side and user-side perspectives. On the platform side, MFMs can be used for multimodal product content completion, such as generating product descriptions from images or generating product images from textual information. When the generated content is used for product presentation, the key requirement is faithfulness: the generated modality should recover item information from the available modalities and remain consistent with the original product. When it is used for recommendation-model training, the key requirement is downstream utility: the generated content should provide effective multimodal signals that improve recommendation performance. Missing-modality and incomplete-content problems in e-commerce therefore motivate recommendation-oriented product content completion, where generation must be evaluated through both content fidelity and recommendation effectiveness (H. Li et al., 2020; F. Liu, Cheng, Zhu, Liu, & Nie, 2022; Malitesta, Rossi, Pomo, Di Noia, & Malliaros, 2024; K. Wang, Shao, Zhang, Chen, & Huo, 2023; B. Zhao, Zhang, & Zou, 2025). On the user side, generation adaptation focuses on popular short-video creation in recommendation-driven social media platforms, where MFMs can assist users in producing titles, prompts, and videos that enter the recommendation pool. We propose LLMPopcorn, an empirical study that explores MFM-assisted generation of popular micro-videos.

The central question of this thesis is:

> How can the broad representation and generation capabilities of multimodal foundation models be adapted for recommender systems?

This thesis addresses the question through two connected perspectives: *recommendation-oriented representation adaptation* and *recommendation-oriented generation adaptation*. The first perspective studies how MFM-derived item representations can be learned against recommendation objectives. It covers continuous representation adaptation of MFMs and discrete representation adaptation, both guided by the recommendation objective. The second perspective studies how MFM generation can support recommendation from both platform and user sides. On the platform side, generation completes missing product modalities for item presentation and recommendation-model training. On the user side, generation creates content that may enter a recommendation platform and compete for engagement.

1.1 Terminology and Scope

Representation adaptation. This thesis studies recommendation-driven adaptation of MFM-based representation learning in two settings: *continuous representation adaptation* and *dis-*

crete representation adaptation. Both pursue joint optimisation of the trainable representation-learning components and the recommender.

Continuous representation adaptation. The MFM-based representation-learning pipeline is trained under a recommendation objective and produces real-valued embeddings $\mathbf{e}_v \in \mathbb{R}^d$ for item v. The term *continuous* refers to this real-valued representation space. The recommendation loss is backpropagated through the differentiable representation-learning path to jointly optimise the trainable parameters of the representation pipeline and the recommender. In Chapters 3–6, this is implemented through parameter-efficient side adaptation: the MFM backbones remain frozen, while the side-adaptation modules and recommender are jointly trained end to end.

Discrete representation adaptation. Discrete representation adaptation jointly learns the semantic-ID construction module and the generative recommender under recommendation supervision. In Chapter 7, item content is encoded and quantised into a finite sequence of codebook indices, $\mathbf{c}_v = (c_{v,1}, \ldots, c_{v,L})$, where $c_{v,\ell} \in \{1, \ldots, K_\ell\}$ and K_ℓ is the number of entries in codebook ℓ. Each integer selects a codebook entry, and the resulting sequence forms the item's semantic identifier (SID). The recommender receives histories expressed using these IDs and predicts the IDs of the next item. DIGER addresses the non-differentiable assignments and codebook collapse through differentiable approximations and exploratory codebook learning, retaining discrete code indices at inference.

Generative recommendation and content generation. Generative recommendation predicts identifiers of existing catalogue items, as in Chapter 7. Content generation produces text, images or videos, as in Chapters 8 and 9. Generating a semantic identifier therefore has a different output and evaluation target from generating product content or a micro-video. These two meanings of generation are distinguished throughout the thesis.

Recommendation-oriented adaptation. In this thesis, adaptation means making a foundation model's representations or generated content useful for a recommendation-related objective. Representation adaptation changes the trainable components that map item content to recommendation representations. Generation adaptation includes the evaluation, prompting and training strategies studied for missing product modalities and content creation.

1.2 Thesis Statement

The thesis advances the following statement:

> This thesis argues that recommender-system ecosystems can benefit from multimodal foundation models when their representation and generation capabilities are

adapted to recommendation-oriented objectives.

This statement has two implications. First, this thesis argues that effective adaptation of multimodal foundation models for recommendation requires aligning both representation and generation with recommendation-oriented objectives. Representation adaptation must account for diverse item representation forms, MFM architectures, encoder designs, and modality settings, while generation adaptation must ensure that MFM-generated content serves its specific role within the recommendation platform. On the platform side, generated content should support faithful product content completion and improve downstream recommendation-model training. On the user side, generated content should support content creation for recommendation-driven platforms, where popularity, engagement, and exposure are central outcomes. Together, these two implications define adaptation as the process of making MFM representation and generation capabilities useful for recommender systems.

1.3 Research Questions

Following the two-perspective structure, this thesis addresses seven research questions.

RQ1: How can multimodal foundation models be efficiently adapted through end-to-end joint optimisation with a recommender in the continuous representation setting? *Addressed in Chapter 3.*

RQ2: How can continuous representation adaptation be scaled from paired encoders to larger and asymmetric foundation encoders? *Addressed in Chapter 4.*

RQ3: How can multiple modality-specific foundation models be jointly adapted with a recommender in the continuous representation setting? *Addressed in Chapter 5.*

RQ4: How can deep unified single-tower multimodal embedding models be adapted for recommendation? *Addressed in Chapter 6.*

RQ5: How can semantic-ID construction and a generative recommender be effectively aligned through joint optimisation? *Addressed in Chapter 7.*

RQ6: How should multimodal foundation models be evaluated and adapted for platform-side missing-modality completion, supporting both item presentation and recommendation-model training? *Addressed in Chapter 8.*

RQ7: How can foundation-model-assisted content generation be studied and improved on the user side when the generated content is intended to gain popularity? *Addressed in Chapter 9.*

1.4 Research Progression and Connections

Starting from efficient continuous representation adaptation. The research began with continuous representation adaptation of multimodal foundation models (MFMs). The objective was to adapt the MFM-based representation-learning pipeline end to end with the recommender, using recommendation gradients to shape its real-valued representations. The first obstacle identified was efficiency: updating large multimodal encoders through the recommendation loss was expensive in both training time and GPU memory. IISAN (Chapter 3) addresses this obstacle through decoupled side adaptation and cached frozen-backbone states, allowing the trainable adaptation modules to be jointly optimised with the recommender.

Extending IISAN to different backbone architectures. IISAN then motivated three complementary extensions, each targeting a different change in the available foundation-model backbones:

- **Scale and asymmetry: IISAN-Versa (Chapter 4).** The increasing use of large language models motivated adaptation of larger text encoders alongside smaller visual encoders. IISAN-Versa extends the architecture to these asymmetric backbones, handling differences in depth and hidden dimensions.
- **Modality breadth: CROSSAN (Chapter 5).** CROSSAN extends adaptation from an image–text pair to four modalities—text, image, video and audio—with separate modality-specific backbones. It studies how to preserve cross-modal interaction while adapting this wider collection of encoders efficiently.
- **Deep unified encoders: Stresa (Chapter 6).** As embedding models moved towards larger LLM-based single-tower architectures that jointly encode multimodal content, the adaptation problem changed from connecting separate towers to using deep hidden states within a unified model. Stresa develops stream-aware side adaptation to preserve and update information across those layers.

These studies share the principle of joint optimisation under recommendation supervision. Their overlapping background reflects this common starting point, while their research questions address scale and asymmetry, modality breadth, and deep unified architectures respectively.

Extending joint optimisation to semantic-ID generative recommendation. In recent years, semantic-ID-based generative recommendation has gradually become a mainstream research direction, with representative developments such as OneRec (Deng et al., 2025) and ETEGRec (E. Liu et al., 2024). This shift motivated extending the same joint-optimisation objective to the evolving recommendation paradigm. In this setting, real-valued content vectors are converted into discrete codebook indices, and the recommender generates the IDs of the next item.

The research focus therefore expanded from efficient continuous representation adaptation to enabling joint optimisation when a semantic-ID construction module lies between the content encoder and the recommender.

Two obstacles arise at this interface: hard code assignments are non-differentiable, and a naive straight-through estimator can cause codebook collapse during joint training. DIGER (Chapter 7) addresses these obstacles through differentiable semantic-ID learning and exploratory codebook optimisation, aligning semantic-ID construction directly with the recommendation objective. This lays a methodological foundation for a unified large recommendation model trained end to end across the following pipeline:

$$\text{MFM content encoder / adaptation} \longrightarrow \text{semantic IDs} \longrightarrow \text{generative recommender.}$$

The longer-term goal is to jointly optimise content encoding, semantic-ID construction and generative recommendation within one model. Stable differentiable training through the semantic-ID interface is necessary for the recommendation loss to reach the upstream trainable MFM components. DIGER tackles the fundamental differentiability and codebook-stability problems at this interface; building and evaluating the complete unified model remains future work. Within this broader direction, combining DIGER with IISAN-style side tuning is one possible route to efficient adaptation of the upstream content features.

Exploring MFM generation within the recommendation ecosystem. Having studied representation adaptation, the research also considered how the generation capabilities of MFMs could serve recommendation applications. This required application-level problem definitions and empirical evidence about the usefulness of generated content. The thesis therefore presents two empirical studies organised around two roles in the recommendation ecosystem.

From the **platform side**, MMPCBench (Chapter 8) selects missing-modality completion as a representative application: a platform uses the available product information to generate a missing description or image. The study defines this problem and evaluates both the fidelity of the completed content and its usefulness for downstream recommendation. From the **user side**, LLMPopcorn (Chapter 9) selects popular content creation as a representative application: a creator uses MFM-generated prompts and micro-videos to produce content intended to attract engagement. These studies explore how generation serves the different goals of platforms maintaining item catalogues and users creating content.

The thesis consequently develops two connected perspectives: recommendation-oriented representation adaptation in Chapters 3–7, and recommendation-oriented generation adaptation in Chapters 8–9. Within the first perspective, continuous representation adaptation and discrete representation adaptation share the objective of joint optimisation; DIGER addresses the differentiability and training-stability obstacles introduced by semantic IDs, contributing foundations for a future unified model trained end to end. The second perspective investigates how the same

class of foundation models can generate content for different participants in the recommendation ecosystem.

1.5 Contributions

The seven studies in this thesis form a cumulative progression. They are organised according to the two perspectives that structure the thesis argument.

1.5.1 Recommendation-Oriented Representation Adaptation

Continuous representation adaptation: establishing efficient adaptation. Chapter 3 introduces IISAN, which decouples trainable side-adapted networks from frozen multimodal backbones. By caching backbone representations and excluding the backbone from the backward computation graph, IISAN establishes a practical form of parameter-efficient adaptation for recommendation-driven continuous representation learning. This work provides the efficiency basis for the subsequent representation chapters and exposes limitations in architectural depth and breadth.

Continuous representation adaptation: supporting larger and asymmetric encoders. Chapter 4 introduces IISAN-Versa. It extends cached decoupled adaptation to modern multimodal encoder pairs whose depths, dimensions, and representation capacities are unbalanced, including settings with a large LLM-based text encoder and a smaller vision encoder. This chapter moves the thesis from a relatively fixed two-encoder setting to scalable and asymmetric MFM adaptation.

Continuous representation adaptation: extending to multiple modality-specific foundation models. Chapter 5 introduces CROSSAN. It expands the framework from image–text encoder pairs to settings containing multiple modality-specific foundation models, including text, image, video, and audio encoders. CROSSAN uses cross-modal gated side adaptation to coordinate modality-specific representations and preserve useful cross-modal interaction.

Continuous representation adaptation: adapting unified single-tower embedding models. Chapter 6 introduces Stresa for large unified multimodal embedding models. In this architecture, side-adapter implementations developed for multi-tower models transfer poorly and may degenerate into shallow adapters that underuse deep hidden representations. Stresa introduces stream-aware residual mechanisms to preserve and update representation streams across depth, enabling more effective deep-layer adaptation.

Discrete representation adaptation: differentiable code learning. Chapter 7 introduces DIGER, which extends recommendation-oriented representation adaptation to the learning of discrete representations. Generative recommendation has motivated token-like item representations and semantic coding schemes (Rajput et al., 2023). DIGER makes semantic codes differentiable so that the discrete indexing space can be shaped by the generative recommendation objective. Its design addresses both gradient blocking caused by discrete assignments and codebook collapse during joint learning.

1.5.2 Recommendation-Oriented Generation Adaptation

Platform-side generation. Chapter 8 introduces MMPCBench, a benchmark for multimodal product content completion. It studies whether MFMs can reconstruct a missing text or image modality from the available modality and evaluates the resulting content in relation to both product presentation and downstream recommendation-model training. The chapter positions missing-modality generation as a platform-level adaptation and evaluation problem.

User-side generation. Chapter 9 introduces LLMPopcorn, which studies generated content at the point of creation. It investigates the popularity of MFM-generated micro-videos and explores how LLM-generated titles and prompts, including popularity-oriented prompt enhancement, can support the creation of content intended for engagement. The chapter extends recommendation-oriented generation from catalogue maintenance to content production.

Taken together, these contributions provide a structured study of MFM adaptation for content-native recommendation: across continuous representation adaptation and discrete representation adaptation, from multi-tower to single-tower architectures, from paired modalities to multiple modality-specific foundation models, and from platform-side content completion to user-side content creation.

1.6 Source Publications, Thesis Format and Author Contributions

1.6.1 Source Material and Adaptation to the Thesis

Chapters 3–9 originate from seven separate co-authored publications or manuscripts: six accepted or published papers and the CROSSAN manuscript under review. Each chapter presents a self-contained study with its own motivation, research questions, methods and evaluation. IISAN-Versa (Chapter 4) is a journal extension of the IISAN conference paper (Chapter 3). Chapters 1, 2 and 10 provide the thesis-level framing, background and synthesis.

This publication-based organisation retains some repetition of background, literature reviews and methodological context, especially across Chapters 3–5. This allows each study to remain understandable as an individual publication, while Section 1.4 explains the progression of ideas and the different motivations of the papers.

The research content was incorporated directly from the source papers and manuscript when the original thesis was assembled. The original paper abstracts were not rewritten. Thesis-specific presentation includes chapter titles, publication notes, chapter-positioning paragraphs, chapter-based numbering and cross-references, and a shared bibliography. The chapters do not present the paper abstracts as separate abstract sections. The changes made during the minor revision are set out in Section 1.6.4, including the movement of Related Work in Chapters 3 and 5.

1.6.2 Contribution Statement

Across the publications in this thesis, Junchen Fu's contributions include: Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; and Data curation. These describe his contributions to the research design, methods, investigations and validation, implementation, project coordination, figures, data preparation and initial manuscripts. The entries below specify the exact roles for each individual study.

The following statements specify each author's contributions to the individual publications. CRediT role[1] names retain their standard spelling.

1.6.3 Publication-by-Publication Outline

Lengths below refer to the identified publication or manuscript version, not the number of pages occupied by its thesis chapter. Where only a preprint length is verified, it is labelled explicitly and is not presented as the final publisher page count.

Chapter 3: IISAN (Fu, Ge, et al., 2024)

Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Jie Wang and Joemon M. Jose. "IISAN: Efficiently Adapting Multimodal Representation for Sequential Recommendation with Decoupled PEFT." Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2024. (Chapter 3)
Type, venue and length. Full paper, Research Track, SIGIR 2024, pp. 687–697 (11 pages). DOI: `https://doi.org/10.1145/3626772.3657725`.

[1] `http://elsevier.com/en-gb/researcher/author/policies-and-guidelines/credit-author-statement`

Study focus. Efficient continuous representation adaptation of image–text MFMs through decoupled side networks jointly trained with the recommender and cached backbone states.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Xuri Ge. Writing – review & editing; Methodology.
Xin Xin. Writing – review & editing.
Alexandros Karatzoglou. Writing – review & editing; Supervision.
Ioannis Arapakis. Writing – review & editing; Supervision.
Jie Wang. Writing – review & editing.
Joemon M. Jose. Supervision; Resources; Writing – review & editing.

Chapter 4: IISAN-Versa (Fu, Ge, Xin, Karatzoglou, et al., 2025)

Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Kaiwen Zheng, Yongxin Ni and Joemon M. Jose. "Efficient and Effective Adaptation of Multimodal Foundation Models in Sequential Recommendation." IEEE Transactions on Knowledge and Data Engineering (TKDE), 2025. (Chapter 4)
Type, venue and length. Regular journal paper (conference extension of IISAN), IEEE Transactions on Knowledge and Data Engineering, 37(12), 2025, pp. 7076–7089 (14 pages). DOI: `https://doi.org/10.1109/TKDE.2025.3608071`.
Study focus. Scalable adaptation for larger and asymmetric text and image encoders, extending the earlier IISAN study.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Xuri Ge. Writing – review & editing.
Xin Xin. Writing – review & editing.
Alexandros Karatzoglou. Writing – review & editing; Supervision.
Ioannis Arapakis. Writing – review & editing; Supervision.
Kaiwen Zheng. Writing – review & editing.
Yongxin Ni. Writing – review & editing.
Joemon M. Jose. Supervision; Resources; Writing – review & editing.

Chapter 5: CROSSAN (Fu, Ni, et al., 2025)

Junchen Fu, Yongxin Ni, Joemon M. Jose, Ioannis Arapakis, Kaiwen Zheng, Youhua Li and Xuri Ge. "CROSSAN: Towards Efficient and Effective Adaptation of Multiple Multimodal Foundation Models for Sequential Recommendation." arXiv preprint, 2025. (Under review for IEEE Transactions on Knowledge and Data Engineering.) (Chapter 5)

Type, venue and length. Regular journal paper submitted to IEEE Transactions on Knowledge and Data Engineering; currently under review. The available manuscript, arXiv:2504.10307v2 (2025), has 12 pages. Source: `https://arxiv.org/abs/2504.10307v2`.
Study focus. Efficient adaptation of multiple modality-specific encoders through cross-modal side adaptation and modality-expert fusion.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Yongxin Ni. Writing – review & editing.
Joemon M. Jose. Writing – review & editing; Supervision; Resources.
Ioannis Arapakis. Writing – review & editing; Supervision.
Kaiwen Zheng. Writing – review & editing.
Youhua Li. Writing – review & editing.
Xuri Ge. Writing – review & editing.

Chapter 6: Stresa (Fu, Zheng, et al., 2026)

Junchen Fu, Kaiwen Zheng, Ioannis Arapakis, Wenhao Deng, Xin Xin, Joemon M. Jose and Xuri Ge. "Stream-aware Side Adaptation for Large Pre-trained Multimodal Embedding Models in Sequential Recommendation." Proceedings of the 34th ACM International Conference on Multimedia (ACM MM), 2026. (Accepted) (Chapter 6)
Type, venue and length. Regular paper, Research Track, accepted by ACM Multimedia 2026. The available manuscript, arXiv:2607.10909v1, has 10 pages; these are manuscript pages, not assigned proceedings pages. Source: `https://arxiv.org/abs/2607.10909v1`.
Study focus. Deep side adaptation for unified multimodal embedding models, using stream-aware fusion and residual stream adapters.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Kaiwen Zheng. Writing – review & editing.
Ioannis Arapakis. Writing – review & editing; Supervision.
Wenhao Deng. Writing – review & editing.
Xin Xin. Writing – review & editing.
Joemon M. Jose. Writing – review & editing; Supervision; Resources.
Xuri Ge. Writing – review & editing.

Chapter 7: DIGER (Fu, Ge, Karatzoglou, et al., 2026)

Junchen Fu, Xuri Ge, Alexandros Karatzoglou, Ioannis Arapakis, Suzan Verberne, Joemon M. Jose and Zhaochun Ren. "Differentiable Semantic ID for Generative Recommendation." Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2026. (Chapter 7)

Type, venue and length. Full paper, Research Track, SIGIR 2026, pp. 369–379 (11 pages). DOI: https://doi.org/10.1145/3805712.3809641.
Study focus. Differentiable learning of discrete semantic identifiers for generative recommendation, with exploratory code learning and uncertainty decay.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Xuri Ge. Writing – review & editing.
Alexandros Karatzoglou. Writing – review & editing; Supervision.
Ioannis Arapakis. Writing – review & editing; Supervision.
Suzan Verberne. Supervision; Writing – review & editing.
Joemon M. Jose. Supervision; Resources; Writing – review & editing.
Zhaochun Ren. Supervision; Resources; Writing – review & editing.

Chapter 8: MMPCBench (Fu, Deng, et al., 2026)

Junchen Fu, Wenhao Deng, Kaiwen Zheng, Ioannis Arapakis, Yu Ye, Yongxin Ni, Joemon M. Jose and Xuri Ge. "Benchmarking Multimodal Large Language Models for Missing Modality Completion in Product Catalogues." Pattern Recognition, 180(A):114020, 2026. (Chapter 8)
Type, venue and length. Regular journal research paper, Pattern Recognition, volume 180, Part A, article 114020, 2026 (16 pages). DOI: https://doi.org/10.1016/j.patcog.2026.114020.
Study focus. Evaluation of missing product text and image completion, considering both content fidelity and downstream recommendation.
Junchen Fu. Conceptualization; Methodology; Investigation; Formal analysis; Data curation; Software; Resources; Project administration; Writing – original draft; Writing – review & editing.
Wenhao Deng. Writing – review & editing; Visualization.
Kaiwen Zheng. Writing – review & editing.
Ioannis Arapakis. Writing – review & editing.
Yu Ye. Writing – review & editing.
Yongxin Ni. Writing – review & editing.
Joemon M. Jose. Writing – review & editing; Supervision.
Xuri Ge. Writing – review & editing.

Chapter 9: LLMPopcorn (Fu, Ge, Zheng, et al., 2026)

Junchen Fu, Xuri Ge, Kaiwen Zheng, Alexandros Karatzoglou, Ioannis Arapakis, Xin Xin, Yongxin Ni and Joemon M. Jose. "LLMPopcorn: Exploring LLMs as Assistants for Popular Micro-video Generation." IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2026. (Chapter 9)

Type, venue and length. Regular paper, research programme, ICASSP 2026, pp. 12007–12011 (5 pages). DOI: https://doi.org/10.1109/ICASSP55912.2026.11463708.
Study focus. An empirical study of LLM-assisted micro-video creation and prompt enhancement, assessed through predicted popularity and a user study.
Junchen Fu. Conceptualization; Methodology; Investigation; Validation; Writing – original draft; Software; Project administration; Visualization; Data curation.
Xuri Ge. Writing – review & editing.
Kaiwen Zheng. Writing – review & editing.
Alexandros Karatzoglou. Writing – review & editing; Supervision.
Ioannis Arapakis. Writing – review & editing; Supervision.
Xin Xin. Writing – review & editing.
Yongxin Ni. Writing – review & editing.
Joemon M. Jose. Writing – review & editing; Supervision; Resources.

1.6.4 Changes Made for the Minor Corrections

The following chapter-by-chapter record identifies changes from the source publications and manuscript. The revision adds thesis-level definitions, the research narrative, publication details and author-contribution statements to Chapter 1, and aligns the framing in the abstract, Chapter 2, the introduction to Part II and Chapter 10. Spelling has been standardised to UK English throughout the thesis.

Chapter 3: IISAN
Related Work is moved before Methodology and the experiments, to Section 3.2. The Chapter Positioning paragraph uses continuous representation adaptation to describe end-to-end adaptation of MFM-based representation learning.

Chapter 4: IISAN-Versa
The LayerDrop caption, ablation discussion and hyperparameter-analysis punctuation receive minor editorial corrections.

Chapter 5: CROSSAN
Related Work is moved before the preliminary study and technical development, to Section 5.2. The independent-variant experiment table (Table 5.12) is kept with its experimental discussion in Section 5.8. Generic backbone notation is introduced in Section 5.5; Equation 5.2 uses the current backbone layer i and previous adapter outputs at $i-1$. Section references, model/dataset names and the missing year of Wang et al. are corrected.

Chapter 6: Stresa
In Section 6.3.3, d_r is defined after Equation 6.11 as the routing query/key dimension, and the square-root scaling is explained.

Chapter 7: DIGER

The Chapter Positioning paragraph explains how DIGER addresses non-differentiable code assignments and codebook collapse, laying a methodological foundation for a future unified large recommendation model trained end to end. Loss subscripts in Section 7.4.3 and ablation notation in Section 7.7 are corrected. The unsupported training-time comparison is removed from Section 7.6.2.

Chapter 8: MMPCBench

The Publication Note gives the journal name in full; the model name and human-evaluation wording receive minor editorial corrections.

Chapter 9: LLMPopcorn

Section 9.3 clarifies that the thesis author, assisted by a co-author, manually checked all generated videos. Model naming, citation punctuation and redundant wording in the conclusion are corrected.

1.7 Thesis Organisation

Chapter 2 develops the background and related-work structure required by the two thesis perspectives. It first reviews recommender systems, sequential recommendation, multimodal recommendation, and multimodal foundation models. It then discusses the recommendation-oriented adaptation gap, emphasising the difference between general MFM capabilities and recommendation-specific objectives, data, and deployment constraints.

The first technical part, Chapters 3–7, develops recommendation-oriented representation adaptation. Chapter 3 establishes efficient cached decoupled continuous representation adaptation of MFMs. Chapter 4 scales this idea to larger and asymmetric encoders. Chapter 5 extends the framework to multiple modality-specific foundation models. Chapter 6 studies deep stream-aware adaptation for unified single-tower embedding models. Chapter 7 studies joint learning of discrete representations and a generative recommender for generative recommendation.

The second technical part, Chapters 8 and 9, develops recommendation-oriented generation adaptation. Chapter 8 studies platform-side missing-modality completion for product content. Chapter 9 studies user-side popularity-oriented micro-video generation. Chapter 10 concludes the thesis and identifies future directions for integrating representation and generation adaptation in recommender systems.

Chapter 2

Background and Related Work

Recommender systems increasingly operate in environments where items are not only identifiers in an interaction log but also multimodal objects. A product may be described by a title, a detailed specification, a price, an image, reviews, and structured attributes, whereas a micro-video may combine visual frames, a cover, an audio track, captions, topics, and engagement traces. News articles, music, local services, and educational resources exhibit the same separation between behavioural interaction data and content that expresses item meaning. This chapter reviews the literature needed to explain why multimodal foundation models (MFMs) are useful in such settings and why their use requires recommendation-oriented adaptation.

The discussion follows the same two perspectives as the thesis: representation adaptation, in which MFMs encode item content into real-valued vectors used directly by recommendation models or quantised into semantic-ID sequences for generative recommendation; and generation adaptation, in which MFMs produce descriptions, images, prompts, videos, or other missing or newly created content. In both cases, an MFM becomes useful to a recommender only when its output supports the relevant platform function, such as presentation, retrieval, ranking, or engagement. The central issue is therefore shared across the two perspectives: a model trained for general semantic representation or generation does not automatically produce outputs aligned with user behaviour, item transitions, catalogue structure, and deployment constraints.

This chapter is organised to build that argument step by step. It first revisits recommender systems and content-rich items, then reviews multimodal foundation models and multimodal recommendation systems. The middle sections discuss continuous representation adaptation and discrete representation adaptation, following the definitions in Section 1.1. The final sections review generation adaptation and the datasets, biases, and evaluation metrics that shape evidence in this area.

2.1 Recommender Systems and Content-Rich Items

Recommender systems predict which items should be retrieved, ranked, or shown to a user. Early collaborative filtering work framed the problem through the user–item interaction matrix. Matrix factorisation represented users and items in a shared latent space and became a standard model for sparse preference prediction (Koren, Bell, & Volinsky, 2009). Neural collaborative filtering extended this formulation by replacing the inner product with nonlinear matching functions (X. He et al., 2017). Graph-based collaborative filtering then used the user–item bipartite graph directly; LightGCN simplified graph convolution and showed that much of the gain in graph recommendation comes from neighbourhood propagation (X. He et al., 2020). These methods established the core idea that user preference can be learned from interaction structure.

At industrial scale, this interaction-centric view expands into multi-stage recommendation systems. Covington et al. described the YouTube recommendation architecture as a candidate-generation and ranking pipeline, making clear that large-scale recommendation is a systems problem as well as a modelling problem (Covington, Adams, & Sargin, 2016). Yi et al. studied sampling-bias-corrected neural modelling for large item corpora, showing that the construction of negatives and sampled candidates can affect both training and evaluation (Yi et al., 2019). These works connected model design with throughput, latency, candidate selection, and training-data construction, a connection that is particularly important for MFM-based recommendation because large content encoders can improve item semantics while also increasing training and serving costs.

Sequential recommendation extends this formulation by modelling the temporal structure of user behaviour. Hidasi et al. introduced GRU4Rec for session-based recommendation and showed that recurrent neural networks can model short-term behavioural intent from ordered sessions (Hidasi et al., 2015). Tang and Wang proposed Caser, a convolutional sequence-embedding model that captures local sequential patterns (J. Tang & Wang, 2018); Kang and McAuley proposed SASRec, which uses self-attention to model user histories and became a widely used baseline for next-item prediction (Kang & McAuley, 2018); and Sun et al. introduced BERT4Rec by adapting bidirectional masked modelling to recommendation sequences (F. Sun et al., 2019). Zhou et al. further proposed S^3-Rec, using self-supervised objectives to improve sequential representation learning (K. Zhou et al., 2020). Together, these methods made sequence order a central modelling signal in modern recommendation.

Most classical and sequential recommenders rely on item IDs, whose embeddings are powerful because they absorb collaborative signals from user behaviour but remain limited when an item is new or rarely consumed. A new product or newly uploaded video may have too few interactions to support a reliable ID embedding, and the embedding does not explain whether two items are similar because of attributes, appearance, language, audio, or video content. Content-rich recommendation addresses these weaknesses by combining behavioural evidence with item-side information. He and McAuley introduced Visual Bayesian Personalised Ranking

(VBPR), showing that visual features can improve implicit-feedback recommendation (R. He & McAuley, 2016). In micro-video recommendation, Wei et al. proposed MMGCN to incorporate visual, acoustic, and textual signals through graph convolution (Y. Wei et al., 2019), while Wei et al. later proposed GRCN to refine multimedia recommendation with graph structures (Y. Wei, Wang, Nie, He, & Chua, 2020). Zhang et al. studied multimodal session recommendation beyond simple item co-occurrence (X. Zhang et al., 2023). These studies helped move recommendation from purely collaborative signals toward content-aware ranking.

Recent datasets and empirical studies made this shift more explicit. Ni et al. released MicroLens, a content-driven micro-video recommendation dataset containing multimodal content and user behaviour (Y. Ni et al., 2023), while Zhang et al. proposed NineRec as a benchmark suite for transferable recommendation across domains (J. Zhang et al., 2023). Yuan et al. revisited ID-based and modality-based recommenders and showed that each signal has a different strength: ID embeddings are strong when behavioural data are dense, whereas modality features improve transfer and cold-start settings (Yuan et al., 2023). Li et al. further studied ID-agnostic multimodal sequential recommenders and examined when content-only representations can be competitive (Y. Li et al., 2024). Collectively, these works discourage a simple replacement view: item content does not remove the need for collaborative learning, but supplies a complementary source of generalisation that must be aligned with user behaviour.

Accordingly, this thesis treats behavioural and content signals as complementary. Content-rich items make MFMs attractive because these models can encode text, images, video, audio, and their multimodal combinations. At the same time, recommendation objectives differ from general semantic objectives: a representation that clusters items by visual appearance may not capture substitutability, sequence transition, price sensitivity, brand preference, or engagement. The background problem is therefore not only how to obtain strong multimodal item embeddings, but also how to adapt them so that they support recommendation.

2.2 Multimodal Foundation Models

Foundation models are large pretrained models whose representations or generation capabilities can be reused across tasks. In language, BERT introduced bidirectional transformer pretraining with masked language modelling (Devlin et al., 2018). RoBERTa refined this paradigm through stronger pretraining design (Y. Liu et al., 2019), and DeBERTa improved contextual language representation through disentangled attention and enhanced pretraining (P. He, Gao, & Chen, 2021; P. He, Liu, Gao, & Chen, 2020). Autoregressive large language models then showed that next-token prediction at scale can produce broad few-shot and instruction-following abilities (Achiam et al., 2023; Brown et al., 2020; Touvron et al., 2023). These models supplied the language side of modern MFM pipelines: titles, descriptions, reviews, attributes, and user-facing text can all be encoded or generated with pretrained language models.

In vision, ResNet established deep residual learning as a standard backbone for visual representation (K. He, Zhang, Ren, & Sun, 2016). Vision Transformer (ViT) later showed that images can be modelled as patch token sequences with transformer architectures (Dosovitskiy et al., 2020). Swin Transformer introduced hierarchical shifted-window attention for scalable vision modelling (Z. Liu et al., 2021). Dehghani et al. studied the scaling of vision transformers to very large parameter counts (Dehghani et al., 2023). These works changed how item images can be encoded: product covers, fashion images, food images, and video frames can be represented by large pretrained visual backbones instead of handcrafted features or small CNNs.

Vision–language pretraining provides a shared representation interface between visual and textual item content. CLIP, proposed by Radford et al., learned visual representations from natural-language supervision through image–text contrastive learning and became a standard encoder for cross-modal retrieval and zero-shot transfer (Radford, Kim, Hallacy, et al., 2021). ALIGN scaled noisy image–text contrastive learning and showed that very large weakly supervised data can produce strong aligned representations (Jia et al., 2021); BLIP-2 connected frozen image encoders with large language models for image–text understanding and generation (J. Li et al., 2023); and Flamingo demonstrated few-shot visual-language learning with interleaved image and text inputs (Alayrac et al., 2022). Recent surveys summarise how multimodal pretraining, adaptation, and generation have become connected research directions for recommendation and related tasks (Q. Liu, Zhu, et al., 2024). For recommendation, this line of work provides a practical bridge between item media and semantic text descriptions.

Audio and video models extend MFM coverage to richer recommendation objects. wav2vec 2.0 learned speech representations through self-supervision (Baevski, Zhou, Mohamed, & Auli, 2020). The Audio Spectrogram Transformer (AST) applied transformer modelling to audio spectrograms and became a useful audio encoder (Gong, Chung, & Glass, 2021). SlowFast networks captured different temporal rates for action recognition (Feichtenhofer, Fan, Malik, & He, 2019), and VideoMAE used masked autoencoding for data-efficient video representation learning (Tong, Song, Wang, & Wang, 2022). These models matter for short-video recommendation because a video item is rarely explained by a single frame or title. Audio, rhythm, motion, and scene progression can all affect engagement.

Beyond encoding existing content, MFMs can generate new modalities and candidate items or media. DALL-E generated images from text prompts (Ramesh et al., 2021), and later text-to-image models improved image fidelity and controllability (Ramesh et al., 2022). VideoPoet, CogVideoX, HunyuanVideo, and LTX-Video represent recent progress in text-to-video and video generation (HaCohen et al., 2024; Kondratyuk et al., 2023; Kong et al., 2024; Z. Yang et al., 2024). These systems turn language instructions into visual or video content that can fill missing product modalities, create item-side descriptions, or help users produce content that enters a recommendation pool.

Recent embedding models further blur the boundary between representation learning and

ranking. Sentence-BERT made transformer sentence embeddings efficient for semantic similarity tasks (Reimers & Gurevych, 2019), and weakly supervised contrastive text embedding models strengthened general text retrieval (L. Wang et al., 2022). Qwen3-VL-Embedding and Qwen3-VL-Reranker integrate multimodal representation, retrieval, and ranking in a unified model family (M. Li et al., 2026). Such models are directly relevant to this thesis because they expose hidden states and embeddings that can be reused as item representations. Their strength also creates a new difficulty: a stronger general embedding model may be deeper, larger, and less aligned with a specific recommendation objective.

For this thesis, the term MFM covers modality-specific encoders, paired image–text encoders, multimodal large language models, unified embedding models, and multimodal generators. These models share one property: they process recommended-item content at a scale and generality beyond traditional task-specific feature extractors. Their capabilities enter recommendation through two interfaces: a representation interface, in which the MFM produces embeddings or hidden states, and a generation interface, in which it produces missing or newly created content. The remaining sections examine how these interfaces have been used and adapted in recommender systems.

2.3 Multimodal Recommendation Systems

Multimodal recommendation studies how item modalities should be represented, fused, and used for ranking. The early direction used pretrained visual or textual features as additional item inputs. VBPR showed the value of visual signals in implicit feedback settings (R. He & McAuley, 2016). MMGCN and GRCN extended this idea to graph-based multimodal recommendation by propagating modality features through user–item graphs (Y. Wei et al., 2020, 2019). These models had a clear impact on the community: they showed that visual and acoustic item content can improve recommendation even when the final objective is behavioural ranking.

Later work moved from simple feature concatenation to modality-aware representation learning. This progression includes TransRec for transferable recommendation from mixture-of-modality feedback (J. Wang et al., 2022), MISSRec for pretraining and transferring multimodal interest-aware sequence representations (J. Wang et al., 2023), multimodal self-supervised learning (W. Wei, Huang, Xia, & Zhang, 2023), bootstrapped latent representations (X. Zhou et al., 2023), and knowledge-perceived multimodal pretraining in e-commerce (Y. Zhu et al., 2021). These methods established that multimodal features should not merely be appended to ID embeddings, but can be pretrained, aligned, contrasted, and transferred.

Fusion is a central design problem because simple concatenation, although efficient and often strong, treats all modalities as uniformly reliable. Gated or attention-based fusion can instead model item-specific and user-specific modality importance. Hu et al. proposed adaptive multi-modality fusion for sequential recommendation (H. Hu, Guo, Liu, & Kan, 2023) and

later studied lightweight modality adaptation through correlation supervision (H. Hu, Liu, Li, & Kan, 2024); Li et al. examined dynamic user interests under modality imbalance (Z. Li et al., 2026); and Guo et al. proposed a mixture-of-modality-experts design for multimodal sequential recommendation (X. Guo, Zhang, Xue, et al., 2025). Together, these studies show that modalities are not always complementary: a modality may be noisy, missing, redundant, or weak for a particular item, so the model needs an explicit mechanism to control its contribution.

Benchmark development also shaped multimodal recommendation by changing which adaptation questions could be evaluated. MicroLens made raw video, audio, cover images, titles, and user interactions available for micro-video recommendation (Y. Ni et al., 2023, 2025); Cheng et al. provided an image dataset for benchmarking recommenders with raw pixels (Y. Cheng et al., 2023); and NineRec supported evaluation across domains and transfer settings (J. Zhang et al., 2023). Whereas earlier work often relied on frozen features extracted by CNNs or language encoders, these raw multimodal datasets make it possible to study end-to-end adaptation of the content encoder itself.

Recent multimodal recommendation surveys organise the area around modality representation, fusion, pretraining, adaptation, and generation (Q. Liu, Hu, Xiao, Gao, & Zhao, 2023; Q. Liu, Zhu, et al., 2024). This perspective is useful because the field now includes several overlapping directions, including graph-based and sequential multimodal recommendation, multimodal CTR prediction, multimodal pretraining, MLLM-based recommendation, and generative recommendation. Within this landscape, the thesis focuses not merely on fusion modules inside a recommender, but on adapting MFM encoders and generators themselves to recommendation-oriented objectives.

A key distinction is whether the multimodal encoder is frozen, fully fine-tuned, or adapted through a lightweight module. Frozen features are easy to cache and deploy, but they cannot change under recommendation loss. Full fine-tuning can align item representations with ranking, but it becomes expensive when each item contains images, video frames, audio, and text. End-to-end image-based fashion recommendation and raw-pixel recommendation studies show the benefit of directly training content encoders (Y. Cheng et al., 2023; Elsayed, Brinkmeyer, & Schmidt-Thieme, 2022). Li et al. and Yuan et al. showed that modality-based recommendation can be strong when content encoders are trained or adapted appropriately (Y. Li et al., 2023; Yuan et al., 2023). The practical problem is how to obtain this alignment without making training infeasible.

The MLLM-based recommendation literature adds another layer. VIP5 connected multimodal inputs with the P5 language-processing recommendation framework (Geng, Tan, Liu, Fu, & Zhang, 2023). Recent work studies how large vision-language models and multimodal large language models can be used for multimodal sequential recommendation (Z. Wang et al., 2025; Ye et al., 2025; P. Zhou et al., 2025). These studies show that recommendation can benefit from stronger perception and reasoning models, but they also expose cost and alignment problems. A

powerful MLLM can describe an item well, but recommendation requires scoring, ranking, and repeated computation over large catalogues.

The field has therefore progressed from appending visual or textual features to collaborative models, through modality-aware fusion and pretraining, to the current problem of adapting very large representation and generation models to recommendation-specific objectives and systems constraints. The next two sections examine continuous representation adaptation and discrete representation adaptation, both guided by recommendation supervision.

2.4 Continuous Representation Adaptation for Recommendation

Continuous representation adaptation jointly optimises the MFM-based representation-learning pipeline and recommender through a differentiable training path over real-valued representations. In a typical setting, an item encoder maps product text, image, video, audio, or multimodal content into a vector. A sequential model then predicts the next item using user history and candidate item embeddings. The simplest strategy is to freeze the item encoder and train only the recommender. This strategy is efficient, but the representation remains organised by the pretraining objective. A CLIP embedding, for example, may reflect image–text alignment, while the recommender needs a space where sequence transitions and user choices are easy to model.

Full fine-tuning provides direct alignment because the recommendation loss updates the MFM and reshapes item representations toward ranking, but its computational cost is high. Transformer encoders store activations for backpropagation, and multimodal recommendation repeatedly processes the same item content across many user histories. When the catalogue is large, recomputing or backpropagating through a large encoder at every training step wastes this repeated-use structure. Efficient adaptation is therefore not a secondary consideration; it determines whether MFM adaptation can be studied and deployed at scale.

Parameter-efficient fine-tuning (PEFT) methods reduce trainable parameters. Houlsby et al. proposed adapters as small bottleneck modules inserted into pretrained transformers (Houlsby et al., 2019). Hu et al. proposed LoRA, which updates low-rank matrices while freezing the main weights (E. J. Hu et al., 2021). Zaken et al. proposed BitFit, updating only bias terms (Zaken et al., 2021). Pfeiffer et al. developed AdapterHub as a framework for adapter-based transfer (Pfeiffer et al., 2020), and Karimi Mahabadi et al. proposed Compacter to further reduce adapter parameters (Karimi Mahabadi, Henderson, & Ruder, 2021). Dettmers et al. introduced QLoRA for efficient fine-tuning of quantised LLMs (Dettmers, Pagnoni, Holtzman, & Zettlemoyer, 2023). These methods had a large impact in NLP and vision because they made downstream adaptation cheaper and easier to store.

Recommendation research adopted PEFT rapidly. TALLRec aligns large language models

with recommendation tasks efficiently (Bao et al., 2023); M6-Rec shows that generative pre-trained language models can act as open-ended recommenders (Z. Cui, Ma, Zhou, Zhou, & Yang, 2022); and Peng et al. study efficient adaptation of large language models for sequential recommendation (Peng et al., 2023). Fu et al. further show that adapters can be effective for recommender systems but require careful design (Fu, Yuan, et al., 2024), while VIP5 uses adapters in a multimodal foundation-model recommender (Geng et al., 2023). These studies helped the community move beyond treating LLMs and MFMs solely as offline feature extractors.

However, trainable parameter count does not fully describe practical efficiency. Adapter and LoRA modules are usually embedded inside the backbone. The backbone remains in the forward and backward computation graph, so GPU memory and training time can remain high even when only a small number of parameters are updated. Cai et al. summarised this issue with the phrase "reduce memory, not parameters" and argued that activation memory is often the bottleneck (H. Cai, Gan, Zhu, & Han, 2020).

IISAN applies this decoupled principle to multimodal sequential recommendation by separating trainable intra-modal and inter-modal side-adapted networks from frozen multimodal backbones (Fu, Ge, et al., 2024). Cached hidden states avoid backpropagation through the full encoder, while the distinction between intra-modal adaptation and inter-modal adaptation preserves both modality-specific refinement and image–text interaction. Methodologically and practically, IISAN shows that multimodal item encoders can be aligned with a recommendation loss without paying the full cost of end-to-end fine-tuning.

IISAN-Versa extends this direction to larger and asymmetric encoders. Modern MFM pipelines rarely contain balanced towers. A recommender may use a large LLM-based text encoder and a smaller vision encoder, with different depths and hidden dimensions. We addressed this by introducing layer selection and dimensional alignment for asymmetric multimodal encoders (Fu, Ge, Xin, Karatzoglou, et al., 2025). The finding is significant for the field because scaling the text encoder can improve recommendation, but only when the adaptation method handles architectural mismatch. This shifts the adaptation question from "which foundation model is strongest?" to "how should different foundation models be connected to the recommendation objective?"

CROSSAN then moves from two modalities to multiple modality-specific foundation models. We proposed cross-modal side adapters and a mixture-of-modality-expert fusion mechanism for text, image, video, and audio encoders (Fu, Ni, et al., 2025). The method argues that independent side adapters can miss useful cross-modal dependencies. Cross-modal gating allows modality-specific side branches to exchange information while preserving the efficiency of decoupled adaptation. This contribution is important for content-rich recommendation because real item content often contains more than two modalities. A method designed only for text and image cannot fully exploit micro-video content, where visual motion and audio can affect user response.

Unified embedding models create another adaptation setting. A single-tower multimodal embedding model can encode text, image, document image, and video into one representation space (M. Li et al., 2026). Multi-tower side-adapter designs do not transfer directly because there are no separate modality towers to fuse. Deep adaptation also becomes delicate: adding more adapted layers can overwrite useful earlier representations or destabilise the side memory. Recent work on Hyper-Connections studies stream-aware residual designs in large models (Xie et al., 2025; D. Zhu et al., 2024). Stresa builds on this idea by treating deep side adaptation as a memory-preserving process. Instead of simple residual addition and sigmoid fusion, it uses stream-aware residual mechanisms to preserve and update representation streams across depth. This places depth stability alongside efficiency and modality interaction as a core design issue for continuous representation adaptation.

Continuous representation adaptation therefore involves several connected problems: aligning MFM embeddings with behavioural objectives, controlling training cost, handling asymmetric encoder architectures, coordinating multiple modalities, and preserving deep hidden information. The thesis chapters on IISAN, IISAN-Versa, CROSSAN, and Stresa follow this progression. They all treat frozen-backbone adaptation as a way to combine strong MFM content representations with the repeated-use and efficiency constraints of recommender systems.

2.5 Discrete Representation Adaptation for Generative Recommendation

Generative recommendation changes the representation interface. Instead of scoring each candidate item through a vector dot product, a generative recommender produces item identifiers as token sequences. This idea is related to neural retrieval and differentiable search indexes, where documents or items are retrieved by generating IDs (Lewis et al., 2020; Tay et al., 2022). Geng et al. proposed P5 and framed recommendation as language processing, unifying several recommendation tasks through prompts and text generation (Geng, Liu, Fu, Ge, & Zhang, 2022). Rajput et al. introduced generative retrieval for recommender systems and made semantic item IDs a central design component (Rajput et al., 2023). This line encouraged the recommendation community to treat item indexing as a learnable representation problem.

Item-ID inputs may use arbitrary IDs, collaborative IDs, content-derived IDs, or semantic IDs. Arbitrary IDs are easy to assign but do not expose item meaning to the model. Collaborative IDs can encode behaviour but struggle with cold-start items. Content-derived semantic IDs use item text, image, or multimodal representations to construct token-like identifiers. Hou et al. proposed learning vector-quantised item representations for transferable sequential recommenders (Y. Hou, He, McAuley, & Zhao, 2022). Hua et al. studied how to index item IDs for recommendation foundation models (Hua, Xu, Ge, & Zhang, 2023). These works established that item tokenisation is not a preprocessing detail; it changes what the generative model can

learn.

Recent work has explored stronger tokenisation schemes. Liu et al. proposed a generative recommender with end-to-end learnable item tokenisation (E. Liu et al., 2024). Wang et al. studied learnable item tokenisation for generative recommendation (W. Wang et al., 2024). Zhai et al. developed a contrastive framework for item tokenisation (Zhai et al., 2025). Lin et al. proposed order-agnostic identifiers for LLM-based generative recommendation (X. Lin et al., 2025). Zheng et al. studied pretraining generative recommenders with multi-identifier item tokenisation (Zheng et al., 2025). Zhang et al. converted semantic codebooks to textual document identifiers for generative search (Y. Zhang et al., 2025). Hou et al. generated long semantic IDs in parallel (Y. Hou, Li, et al., 2025). These methods helped make tokenisation a major research theme in generative recommendation.

Semantic IDs are especially relevant to MFM-based recommendation. An MFM can encode item text, image, video, and audio into a real-valued vector; a quantiser can then map the vector to a sequence of discrete codes. The resulting codes give a generative recommender a token vocabulary that reflects item content. This is useful for cold-start and transfer settings because new items can be assigned codes from their content. It also connects representation adaptation with generation: the item embedding space and the generated item-token space become linked.

The main difficulty is optimisation because quantisation is non-differentiable, preventing recommendation gradients from directly updating the discrete assignment. Bengio et al. studied gradient estimators for stochastic neurons, including straight-through estimators (Bengio et al., 2013), and Jang et al. proposed the Gumbel–Softmax trick for differentiable categorical sampling (Jang, Gu, & Poole, 2016). Residual and stochastic quantisation methods have also been used in representation learning and image generation (D. Lee, Kim, Kim, Cho, & Han, 2022; Takida et al., 2022). These tools are useful, but recommendation adds a further requirement: code usage must remain stable, expressive, and aligned with item ranking or generative retrieval.

Codebook collapse is one of the main risks. If only a small subset of codes is used, semantic IDs lose capacity and many items become difficult to distinguish. If the code assignments change too abruptly during training, the generative recommender sees an unstable target vocabulary. Recent generative recommendation work tries to address this through disentangled semantic-collaborative modelling, multimodal semantic IDs, or industrial-scale tokenisation frameworks (Han et al., 2025; C. Liu et al., 2025; Y. Wang et al., 2025; J. Zhu et al., 2025). These studies show that discrete item representation is not only a modelling problem; it also affects training stability, indexing, and deployment.

DIGER follows this line by asking how semantic IDs can be adapted under the recommendation objective itself. The key idea is to make semantic ID learning differentiable enough for joint optimisation while avoiding collapse. In the broader literature, this position sits between vector-quantised representation learning and generative recommendation. It keeps the MFM-derived

content signal, but lets the discrete code space respond to behavioural recommendation loss. This matters because a static content quantiser may group items by general semantic similarity, while a recommender needs codes that preserve distinctions users act on.

Continuous representation adaptation trains the MFM-based representation-learning pipeline and recommender through real-valued representations. Discrete representation adaptation extends the joint-learning objective to semantic-ID construction and generative recommendation. Both adapt trainable representation-learning components under recommendation supervision, with semantic-ID learning introducing the additional challenges of discrete assignments and codebook stability.

2.6 Generation Adaptation for Recommender Systems

MFMs can generate recommendation-relevant content and, in some settings, the candidate items themselves. Generation in recommendation is therefore not limited to explanations or conversational responses: it can complete catalogue fields, alter item presentation, supply content signals for model training, or create videos that enter the recommendation pool. The adaptation question is consequently role-specific. A generator may produce fluent text, plausible images, or visually appealing videos, but the recommender-system setting asks whether the output is faithful, useful for ranking, and appropriate for the engagement objective through which it will be evaluated.

Platform-side generation often starts from missing or incomplete item content, such as absent product descriptions, attributes, or images. Aspect-aware multimodal summarisation shows that product text generation should preserve item attributes and user-facing aspects (H. Li et al., 2020); attribute-aware graph models and MPKGAC address missing attributes in e-commerce (F. Liu et al., 2022; K. Wang et al., 2023); and missing-modality studies show that affected items should not simply be removed from multimodal recommendation datasets (Malitesta et al., 2024; B. Zhao et al., 2025). Kim et al. further study disentangling and generating modalities when product modalities are missing (J. Kim, Kang, Kim, Kim, & Park, 2025). Together, these works establish product completion as a recommendation problem rather than merely a data-cleaning operation.

Classical imputation methods such as MICE and MissForest are useful for tabular fields (Stekhoven & Bühlmann, 2012; van Buuren & Groothuis-Oudshoorn, 2011), but they are not designed to reconstruct a faithful product image from a description or to write a description from an image. Multimodal generators are more suitable for these free-form modalities: a text-to-image model can synthesise visual content, an image-to-text model can describe a product, and an MLLM can connect attributes, images, and text. Their flexibility also creates a hallucination risk. A description that invents a material, size, or function may be fluent but damage user trust and downstream ranking, while an image that changes the product may appear attractive while misrepresenting the item. Recommendation-oriented generation must therefore evaluate factual

consistency and downstream utility together.

MMPCBench is motivated by this platform-side completion setting and treats missing product text and missing product images as complementary forms of catalogue repair. The task requires the generated modality to recover item information from the available modality, support product presentation, and provide useful signals for recommendation-model training. This evaluation framing connects content fidelity with recommendation effectiveness and addresses a gap in general MFM benchmarks, which rarely ask whether generated content improves ranking when inserted into a recommender pipeline. User-side generation concerns candidate content—including titles, prompts, and micro-videos—created by users or creators before it enters a recommendation ecosystem. Once generated, a video and its associated title become candidate items that the platform may rank, expose, and use to collect feedback for future training. Micro-video analysis surveys describe how short-video content depends on visual, audio, textual, temporal, and social signals (J. Guo et al., 2024); popularity-prediction studies use multimodal features and retrieval augmentation to estimate engagement (J. Lu et al., 2024; Zhong et al., 2024); and advertising or traffic-optimisation studies show that exposure and revenue objectives shape short-video ecosystems (Ran et al., 2022). These works provide the behavioural foundation for user-oriented generation.

The content-creation side is enabled by generative video models such as VideoPoet, CogVideoX, HunyuanVideo, and LTX-Video, which show progress toward higher temporal coherence and lower latency (HaCohen et al., 2024; Kondratyuk et al., 2023; Kong et al., 2024; Z. Yang et al., 2024). Surveys of generative AI for video summarise rapid advances in video generation, understanding, and streaming (P. Zhou et al., 2024). For recommender systems, these models make it possible to generate not only descriptions of items but also the items themselves, allowing an AI-generated micro-video to compete for exposure with human-created content.

LLMPopcorn studies this user-side setting by asking whether LLMs and video generators can assist popularity-oriented micro-video creation through title generation, prompt construction, and prompt enhancement. Unlike platform-side product completion, the central objective is engagement under a platform-like popularity model rather than product faithfulness alone, which makes evaluation more difficult. Popularity predictors are trained on historical exposure and interaction data and therefore inherit platform biases; a generated video that receives a high predicted score may exploit shallow cues or dataset artefacts. User-side generation consequently requires careful offline and online evaluation of diversity, authenticity, creator incentives, and long-term user satisfaction.

Generation adaptation thus extends the role of MFMs across two positions in the recommendation ecosystem. On the platform side, generation repairs incomplete catalogues and supplies missing modalities for representation learning; on the user side, it changes the supply of candidate content. In both cases, the generator participates in a feedback loop with ranking,

exposure, user behaviour, and future data collection rather than operating as an isolated model. This ecosystem role is why generation adaptation in this thesis is treated as a recommendation-oriented problem.

2.7 Datasets, Biases, and Evaluation Metrics

Evaluation is where the different adaptation problems in this thesis become separable. A real-valued MFM-derived item vector is evaluated by whether it improves ranking under a fixed recommendation protocol. A discrete semantic ID is evaluated by whether it forms a useful item vocabulary for generative recommendation and remains stable during joint optimisation. A product-completion generator is evaluated by whether the completed modality is faithful enough for presentation and useful enough for downstream recommendation. A micro-video generator is evaluated by whether the generated content is likely to receive popularity or engagement. These settings share the same recommendation data regime, but they should not be compressed into a single list of metrics.

All four evaluation settings share the same starting point: implicit feedback, in which a click, view, purchase, like, comment, or watch event records an observed user–item interaction after exposure without revealing every item the user would have liked. Unobserved interactions are therefore not clean negative labels, and offline test sets inherit exposure bias, popularity bias, position bias, platform-policy bias, and previous-recommender bias. Yi et al. studied sampling bias in large-corpus item recommendation, and Krichene and Rendle showed that sampled ranking metrics may be misleading when the sampled candidate set does not match the full-ranking task (Krichene & Rendle, 2020; Yi et al., 2019). For this reason, the representation chapters use a standard leave-one-out protocol where possible: the last item in each sequence is reserved for testing, the penultimate item for validation, and earlier interactions for training. HR@K, Recall@K, NDCG@K, and related top-K metrics then provide reproducible evidence for short-term next-item ranking, but not a complete measure of user satisfaction.

The available dataset determines which adaptation question can be studied credibly and which modality interactions can be observed. Amazon product data combines user interactions with product text and images (J. Ni, Li, & McAuley, 2019); MicroLens provides user behaviour together with titles, cover images, videos, and audio for content-driven micro-video recommendation (Y. Ni et al., 2023, 2025); NineRec supports transfer and larger-scale domain-generalisation studies (J. Zhang et al., 2023); and raw-pixel datasets test whether recommenders can benefit from images without reducing the modality to fixed shallow features (Y. Cheng et al., 2023). These datasets are not interchangeable: product recommendation tests catalogue semantics and incomplete product information, micro-video recommendation tests richer temporal and audio-visual signals, and transfer benchmarks test generalisation beyond one domain.

Continuous representation adaptation. For continuous representation adaptation, the evaluation target is an adapted item embedding. The central question is whether raw content encoded through an MFM, after adaptation, improves next-item ranking compared with ID-only models, frozen content features, full fine-tuning, and embedded PEFT. In the IISAN and IISAN-Versa settings, Amazon categories such as Industrial and Scientific, Musical Instruments, and Office Products provide paired text–image product data. In CROSSAN, MicroLens makes it possible to evaluate more than two modality-specific foundation models, including text, image, video, and audio. In Stresa, unified multimodal embedding backbones such as Qwen3-VL-Embedding motivate evaluation on single-tower embedding models (M. Li et al., 2026).

The main performance metrics for this part are HR@10 and NDCG@10, sometimes extended to HR@20 and NDCG@20 when broader ranking lists are reported. HR measures whether the target item is retrieved within the top-K results, while NDCG gives more credit when the target appears nearer the top of the list. These metrics directly match the representation objective and allow fair comparison across encoder choices when the downstream sequential encoder, data split, and candidate set are fixed. Representation quality alone is not sufficient evidence, however, because efficiency is part of the evaluation standard for MFM-based recommenders. IISAN distinguishes trainable-parameter efficiency from training-system efficiency through training time, trainable parameters, GPU memory, and the composite TPME metric (Fu, Ge, et al., 2024); IISAN-Versa and CROSSAN further evaluate cached hidden-state storage, feasible batch size, and the cost of adapting larger or multiple encoders (Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025). A small NDCG gain is less meaningful if the method cannot be trained or refreshed on a large catalogue.

The continuous representation setting also exposes modality bias. Text often carries dense product semantics, while images, videos, and audio may be noisy, redundant, or missing. A modality can appear useful simply because it is more complete for popular items or for professionally maintained categories. CROSSAN-style evaluation therefore needs modality ablations, fusion ablations, and efficiency comparisons in addition to final HR/NDCG values. Stresa adds another diagnostic: when adapting deeper hidden states, the evaluation should test whether more depth actually helps or whether conventional side adapters degrade because earlier information is overwritten. In other words, evaluation of continuous representation adaptation asks not only "which model ranks better?", but also which modality, layer, adapter path, and caching strategy produces that result.

Discrete representation adaptation for generative recommendation. Generative recommendation changes the evaluation object from a learned continuous representation to a sequence of discrete item-code indices. In models such as TIGER, items are mapped into semantic IDs and the recommender predicts the next item by generating an ID sequence (Rajput et al., 2023). Later work studies learnable item tokenisation, multi-identifier tokenisation, and semantic-ID-

based recommendation, showing that the item vocabulary itself has become a modelling component (Y. Hou, Li, et al., 2025; X. Li et al., 2025; E. Liu et al., 2024; W. Wang et al., 2024; Zheng et al., 2025). For this setting, ranking metrics such as Recall@K and NDCG@K remain necessary, because the final system still recommends items. They are not sufficient, because a semantic ID can produce acceptable ranking for the wrong reason while using a collapsed or unstable codebook.

Discrete adaptation therefore requires representation-level diagnostics in addition to ranking metrics. Code usage reveals whether the codebook is balanced or dominated by a small subset of entries; codebook capacity and semantic-ID length test whether the identifier space is too small, too large, or unnecessarily long; SID drift measures how often item identifiers change during training; and train–inference agreement measures whether stochastic training assignments match deterministic inference-time IDs. These diagnostics are especially important for differentiable semantic IDs because gradient-based learning must pass through a discrete bottleneck. A method that improves Recall@10 while collapsing most items into a few codes has not solved the representation problem; it has hidden the problem inside the item-token space. DIGER is therefore evaluated through both downstream recommendation performance and codebook stability.

Platform-side generation adaptation. For platform-side generation, the output is a missing product modality. The evaluation question is no longer whether an embedding ranks well by itself, but whether a generated description or product image can serve the role of the missing modality. Missing-modality studies show that incomplete or imbalanced modalities are common and should not be handled only by dropping affected items (Malitesta et al., 2024; C. Wang, Niepert, & Li, 2018; B. Zhao et al., 2025). Product attribute completion work also shows the practical importance of reconstructing incomplete e-commerce information (K. Wang et al., 2023). MMPCBench follows this line but evaluates open-ended multimodal product completion instead of only feature-level imputation.

This setting requires two complementary evaluation layers, the first of which measures content quality. For Image-to-Text completion, lexical and semantic metrics such as token overlap, METEOR, BERTScore, MoverScore, SBERT similarity, and TF–IDF-based similarity measure whether the generated description is close to the reference and semantically aligned (Banerjee & Lavie, 2005; T. Zhang, Kishore, Wu, Weinberger, & Artzi, 2019; W. Zhao et al., 2019). For Text-to-Image completion, PSNR, SSIM, MSE, LPIPS, FID, and CLIP-based similarity measure reconstruction quality, perceptual similarity, distributional realism, and semantic alignment (Heusel, Ramsauer, Unterthiner, Nessler, & Hochreiter, 2017; Hore & Ziou, 2010; Radford, Kim, Hallacy, et al., 2021; Z. Wang, Bovik, Sheikh, & Simoncelli, 2004; R. Zhang, Isola, Efros, Shechtman, & Wang, 2018). Human evaluation remains necessary because automatic metrics do not fully capture product faithfulness, attribute preservation, visual plausibility, commercial

readiness, or whether a description is useful to a shopper (Van der Lee, Gatt, Van Miltenburg, & Krahmer, 2021). The human dimensions in this thesis therefore reflect the product role: factual accuracy, completeness, and marketing appeal for text, and visual plausibility, attribute fidelity, and commercial readiness for images.

The second layer measures recommendation utility, because a generated modality can look plausible while remaining unhelpful for ranking. Conversely, a generated description may differ from the reference at the word level while preserving the semantic attributes used by the recommender. RecBench therefore evaluates generated modalities as substitutes in downstream multimodal recommendation, using VBPR, BM3, and FREEDOM and reporting Recall@K and NDCG@K (R. He & McAuley, 2016; X. Zhou & Shen, 2023; X. Zhou et al., 2023). This separation is important because CQBench measures whether the content is faithful and plausible, whereas RecBench measures whether it carries sufficient recommendation signal. A weak correlation between content-quality and recommendation metrics is not a contradiction; it indicates that recommendation models may rely on semantic-level signals and remain robust to some low-level generation noise.

User-side popularity generation. For user-side generation, the output is not a completed product field but a candidate micro-video that may enter the recommendation pool, so the evaluation target becomes popularity or engagement potential. General video-generation datasets and quality metrics are useful for checking visual plausibility, temporal coherence, and prompt following (Chivileva, Lynch, Ward, & Smeaton, 2024; P. Zhou et al., 2024), but they do not show whether a generated video is likely to attract engagement on a short-video platform. Micro-video popularity prediction provides a more suitable proxy by modelling engagement from multimodal content and behavioural traces (J. Lu et al., 2024; Zhong et al., 2024). LLMPopcorn adopts this view by using a pretrained MMRA popularity evaluator (Zhong et al., 2024) to score generated title–video pairs, operationalising popularity through an engagement proxy such as the median number of comments.

The popularity setting also benefits from pairwise evaluation because absolute predicted scores can be sensitive to the evaluator scale. Win rate compares two videos generated for the same prompt and asks which is predicted or judged to be more popular, supporting comparisons among LLMs, prompt-enhancement strategies, and video generators. Blind human comparison provides an additional check. Nevertheless, offline popularity remains a proxy learned from historical exposure and interaction data that encode platform bias, creator bias, and content-distribution bias. Popularity-oriented generation must therefore be interpreted cautiously: optimising predicted popularity can improve engagement signals while also encouraging repetitive styles, overfitting shallow cues, or reducing content diversity.

Bias takes different forms across the four evaluation settings. In continuous representation adaptation, popularity and missingness can make a modality appear stronger than it is; in

semantic-ID learning, codebook collapse can reduce the expressiveness of the item vocabulary even when ranking metrics remain acceptable; in platform-side generation, automatic metrics can reward surface similarity without ensuring product faithfulness or downstream usefulness; and in user-side generation, popularity predictors can reproduce platform exposure feedback loops. These limitations do not make offline evaluation useless, but they define the claims that offline evidence can and cannot support.

The evaluation principle followed in this thesis is therefore role-specific: an MFM used as an encoder should be evaluated through ranking accuracy and practical adaptation cost; an MFM used to build discrete item tokens should be evaluated through both recommendation performance and token-space stability; a model that completes missing product content should be evaluated through content fidelity, human judgement, and downstream recommendation utility; and a model that generates micro-videos should be evaluated through predicted popularity, pairwise win rate, and human preference, while acknowledging that online exposure remains the final test. This role-specific view keeps the evidence aligned with the adaptation problem and connects the datasets, biases, and metrics directly to the methods developed in the rest of the thesis.

I. Recommendation-Oriented Representation Adaptation

The first technical part of the thesis develops recommendation-oriented representation adaptation. It treats multimodal foundation models not as offline semantic feature extractors, but as representation infrastructures that must be adapted within an end-to-end joint-optimised pipeline to align with recommendation objectives.

IISAN first addresses the efficiency of continuous representation adaptation, jointly training MFM side-adaptation modules and the recommender under recommendation supervision. IISAN-Versa, CROSSAN and Stresa extend this approach along three architectural directions: larger asymmetric encoders, broader modality coverage, and deep unified single-tower embedding models. DIGER carries the same joint-optimisation objective into semantic-ID-based generative recommendation. By addressing non-differentiable code assignments and codebook collapse, it lays a methodological foundation for a unified large recommendation model that can jointly optimise MFM-based content encoding, semantic-ID construction and generative recommendation end to end. Developing this full model, including possible use of efficient side tuning, remains future work.

Chapter 3

IISAN: Efficient Decoupled Adaptation of Multimodal Representations

Publication Note

This chapter is adapted from: Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Jie Wang, and Joemon M. Jose. IISAN: Efficiently Adapting Multimodal Representation for Sequential Recommendation with Decoupled PEFT. SIGIR 2024.

Chapter Positioning

This chapter establishes the first step in representation adaptation in the thesis. It asks how a recommendation model can use raw image and text signals from multimodal foundation models without incurring the cost of full fine-tuning. IISAN introduces decoupled PEFT, separates frozen backbones from trainable side networks, and proposes a caching strategy to further enhance performance, thereby providing the efficiency foundation for the later chapters on continuous representation adaptation.

3.1 Introduction

Large foundation models such as GPT-4 (Achiam et al., 2023), DALL-E (Ramesh et al., 2022), LLaMA (Touvron et al., 2023), and CLIP (Radford, Kim, et al., 2021), are at the forefront of AI innovation, captivating the entire AI community. These models have become pivotal in AI research areas such as Natural Language Processing, Computer Vision, and Multimodal Learning Tasks.

In particular, their superior ability to generate universal representations is highly advantageous for sequential recommendation tasks that have recently shifted from traditional reliance

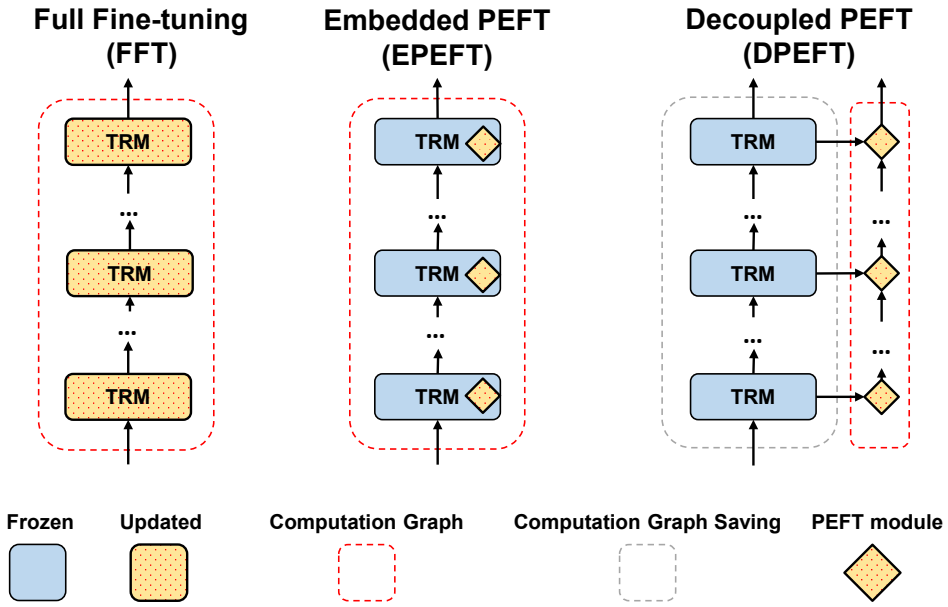


Figure 3.1: Comparisons among Full Fine-tuning (FFT), Embedded PEFT and Decoupled PEFT for feature representation learning. The traditional Embedded PEFT (EPEFT), e.g., Adapter (Houlsby et al., 2019) and LoRA (E. J. Hu et al., 2021), embed the additional trainable parameters into the foundation models, reducing trainable parameters but still having heavy computation graph during backpropagation. The proposed IISAN belongs to Decoupled PEFT (DPEFT), which significantly reduces the size of the computation graph by decoupling the PEFT from backbones and maintains the latest trainable parameters by freezing backbones.

on IDs (Identifiers) (X. He et al., 2017; Hidasi et al., 2015) to multimodal item content (Text, Images, etc.)

While this paradigm shift is undeniably successful, it introduces significant challenges. Large foundation models, characterised by their extensive parameterisation, incur substantial costs when applying traditional full fine-tuning (FFT) methods (Devlin et al., 2018). This partly explains why much research focuses on text modality (Y. Hou, He, et al., 2022; Y. Hou, Mu, et al., 2022; C. Wu, Wu, Qi, & Huang, 2021a; L. Wu et al., 2020) – the training cost for visual modality is even higher. For instance, vision encoders like ViT have long sequence lengths (up to 197), compared to less than 50 in typical text scenarios.

Longer sequence length can significantly increase the activations in GPU memory, which we will explain in Section 3.4.3. This high training cost is even more pronounced in multimodal scenarios, combining text and visual information, which significantly increases model size and data input. Nevertheless, despite such efficiency issues, the intuitive advantage of multimodal representation lies in its ability to comprehensively integrate information, thereby enhancing overall performance. Therefore, optimising efficiency in multimodal recommendation is of paramount importance.

We identified two additional key challenges in the costly training of large foundation models: **GPU memory** and **training time**. The substantial expense of GPU memory, such as that

required for advanced GPUs like A100 and H100 80G, poses a barrier for researchers and engineers interested in developing large foundation models but lack access to these resources. Moreover, the issue of training time encompasses not only extended waiting periods but also the escalation of electricity expenses. Crucially, the energy consumed during lengthy training processes directly contributes to higher carbon emissions, a matter of significant global environmental concern (L. Chen et al., 2022; Williams et al., 2021).

In response to the FFT's efficiency problem, many researchers have turned to parameter-efficient fine-tuning (PEFT) methods, such as LoRA (E. J. Hu et al., 2021), Adapter (Houlsby et al., 2019) and Bitfit (Zaken et al., 2021), etc. Note that, adapter-based approaches have shown performance comparable to FFT in sequential RS, as discussed in (Fu, Yuan, et al., 2024). However, a critical question arises: **Does the parameter efficiency represent the practical model efficiency, i.e. lower parameter $\rightarrow$ lower GPU memory & faster training speed?** PEFT was initially proposed to reduce trainable parameters (Houlsby et al., 2019), aiming to save storage by reducing the need for multiple copies of foundation models for various tasks (Fu, Yuan, et al., 2024; Houlsby et al., 2019). For instance, as shown in Figure 3.1, popular PEFT methods (Houlsby et al., 2019; E. J. Hu et al., 2021; Karimi Mahabadi et al., 2021; Pfeiffer et al., 2020) embed new trainable parameters into the foundation models, namely Embedded PEFT (EPEFT). However, the computation graph of these EPEFTs, such as Adapter and LoRA, shows no significant reduction (H. Cai et al., 2020; Sung, Cho, & Bansal, 2022). Consequently, the cost of backward propagation, i.e. GPU memory and training time, does not decrease appreciably, which continues to be a primary bottleneck during the training stage, as detailed in Section 3.4.

As mentioned before, we argue that two intuitive key issues exist in PEFT research:

- Current embedded PEFT methods are inherently flawed, i.e., the heavy computation graph in backpropagation.
- In the PEFT research methodologies, there is a misconception that parameter efficiency directly translates to overall efficiency, and the model efficiency evaluation (only focusing on the size of trainable parameters) is lopsided.

Current research in multimodal sequential recommendation is suffering from these critical issues (Z. Cui et al., 2022; Fu, Yuan, et al., 2024; Geng et al., 2023). Motivated by this, we conduct our research. Specifically, we develop a *simple yet efficient* Intra- and Inter-modal Side Adapted Network (IISAN) to adapt the multimodal foundation models more efficiently. Compared to traditional EPEFT (Adapter and LoRA, etc.), IISAN's innovations encompass three aspects: (1) We adopt the advanced Decoupled PEFT (DPEFT) structure, described in Figure 3.1, to largely reduce the computation graph; (2) Drawing from our insights into the characteristics of the DPEFT, we further propose to improve efficiency by implementing a caching strategy, as detailed in Section 3.3.1; (3) Based on the traits of multimodality, we embrace the capabilities of both unimodality (intra-SAN) and multimodality interactions (inter-SAN) for better multimodal

representation learning.

Additionally, we correct the misconception that "parameter efficiency is equal to model practical efficiency" in two aspects. First, we undertake a thorough yet accessible analysis in Section 3.4. This analysis focuses on the practical efficiency of various prevalent approaches, including full fine-tuning (FFT), EPEFTs (such as LoRA (E. J. Hu et al., 2021) and Adapter (Houlsby et al., 2019), etc.), and DPEFT (IISAN). The evaluation encompasses three key facets: training speed, trainable parameters, and GPU memory usage. Second, to unify the point of evaluating the practical model efficiency, we introduce a metric named TPME (Training-time, Parameter, and GPU Memory Efficiency), which comprehensively considers the above three critical factors to address the second aforementioned issue in PEFT research.

Overall, the major contributions of our paper are threefold:

- We introduce a novel Intra- and Inter-modal Side Adapted Network (IISAN) following the decoupled PEFT paradigm for end-to-end sequential recommendation tasks based on pre-trained multimodal foundation models. IISAN is allowed to employ caching strategy to further improve efficiency due to its DPEFT property.

- We propose a new practical efficiency metric (TPME), harmonising the costs of training time, trainable parameters, and GPU memory, which offers a holistic measure of model efficiency.

- We provide a detailed and accessible analysis for understanding the high efficiency of the PEFT mechanism. It positions the IISAN as a theoretically superior architecture in terms of practical efficiency compared to prevalent PEFT.

Extensive experiments validate that IISAN achieves comparable performance over the FFT and the state-of-the-art PEFT methods on three widely-used multimodal recommendation datasets, while also delivering a notable enhancement in GPU memory and training time efficiency.

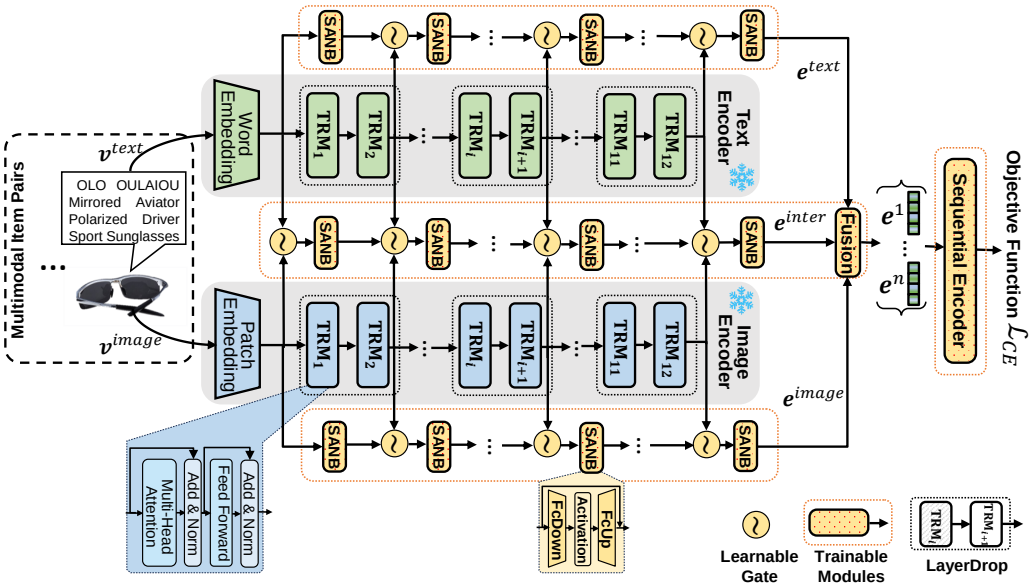


Figure 3.2: An Overview of the IISAN for sequential recommendation. The framework takes the pre-trained text encoder BERT (Devlin et al., 2018) and image encoder ViT (Dosovitskiy et al., 2020) as an example, which contains 12 Transformer-blocks (TRMs) respectively. IISAN proposes intra- and inter-modal side adapted networks, where the intra-modal SANs mainly construct independent adaptive representation learning within two modalities and the inter-modal SAN focuses on the efficient multimodal interactions between layer hidden states in multimodal networks. SANs consist of multiple SAN blocks (SANBs) and learnable fusion gates. Each SANB receives the hidden states from the corresponding layers and makes an adaptive learning optimisation for the final recommendation task by a unified objective function. Notably, we leverage LayerDrop to further omit redundancy.

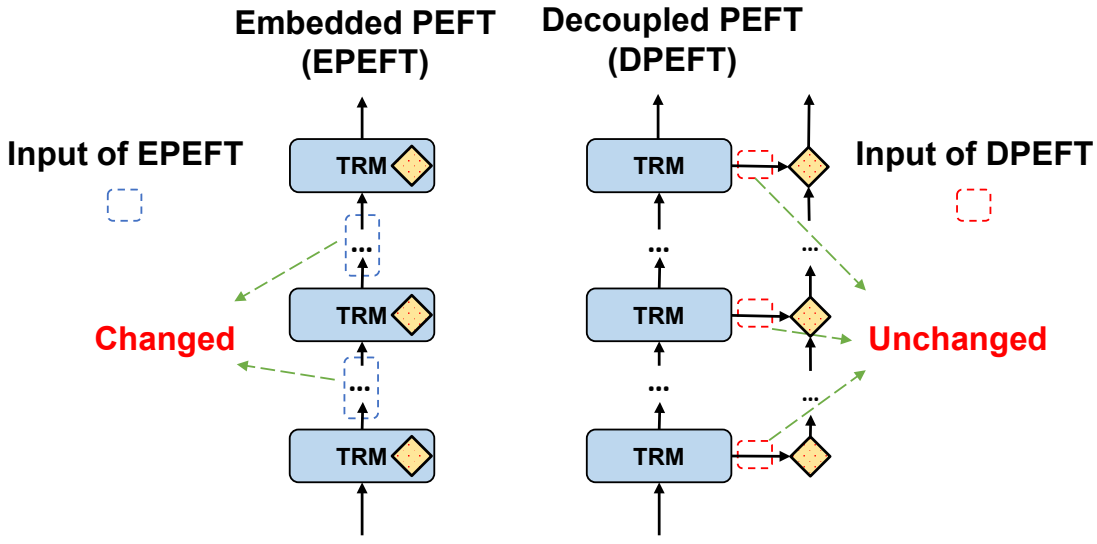


Figure 3.3: Caching strategies comparison. The input for the DPEFT remains constant and, in theory, can be cached. On the other hand, the input for the EPEFT is subject to change as it is influenced by parameter updates in the last block.

3.2 Related Work

Modality-based Sequential Recommendation (MoRec). The recommendation with various modality information has increasingly fascinated the Recommender System community (Y. Cheng et al., 2023; H. Hu et al., 2024; R. Li et al., 2025; Q. Liu, Chen, Sakai, & Wu, 2024; Y. Liu et al., 2023; Y. Ni et al., 2023; Z. Qu et al., 2023; R. Sun et al., 2020; J. Wang et al., 2022, 2023; W. Wei et al., 2023; C. Wu, Wu, Qi, & Huang, 2021b; Yuan et al., 2023; J. Zhang et al., 2023). They deploy the large-scale pre-trained foundation models Brown et al. (2020); Devlin et al. (2018); K. He et al. (2016); P. He et al. (2021, 2020); Radford, Kim, et al. (2021) from NLP and CV (Dosovitskiy et al., 2020; Z. Liu et al., 2021), and CLIP (Radford, Kim, et al., 2021) to encode texts and images. The sequential encoder stays unchanged as the traditional sequential architectures, including SASRec (Kang & McAuley, 2018), GRU4Rec (Hidasi et al., 2015), and BERT4Rec (F. Sun et al., 2019), etc. In addition, IDA-SR (Mu, Hou, Zhao, Li, & Ding, 2022), UniSRec (Y. Hou, Mu, et al., 2022), and VQ-Rec (Y. Hou, He, et al., 2022) realised MoRec by learning item representation from the foundation model in NLP. In terms of image representation in RS, (Y. Wei et al., 2019) and (L. Meng, Feng, He, Gao, & Chua, 2020) fed the RS model with image features extracted from the ResNet-50 (K. He et al., 2016). On the other hand, (Elsayed et al., 2022; Y. Li et al., 2023; Y. Ni et al., 2023; Yuan et al., 2023) have collectively demonstrated that the MoRec framework, when applied through end-to-end learning, significantly enhances performance over previous methods that rely on offline feature extractors. Specifically, (Y. Li et al., 2024, 2023) highlighted that end-to-end training, which integrates both image and text modalities, considerably surpasses systems based solely on a single modality.

However, a significant limitation observed in these end-to-end research studies is their continued reliance on full fine-tuning of large-scale multimodal encoders. This approach often results in a lack of efficiency.

Parameter-efficient Fine-tuning (PEFT). In the domains of NLP and CV, significant efforts are being devoted to investigating PEFT methods. These approaches aim to address the challenge of managing the extensive number of trainable parameters present in large-scale pre-trained models. Foundational works in this area include Adapter (Houlsby et al., 2019), LoRA (E. J. Hu et al., 2021), and BitFit (Zaken et al., 2021). Building on this paradigm, a variety of alternative strategies have been proposed, as evidenced by studies such as (Z. Chen et al., 2022; P. Gao et al., 2023; X. He, Li, Zhang, Yang, & Wang, 2022; Karimi Mahabadi et al., 2021; Pfeiffer et al., 2020; R. Wang et al., 2020; D. Yang et al., 2023; R. Zhang et al., 2021). Nevertheless, these methods predominantly employ Embedded PEFT (EPEFT) and concentrate chiefly on parameter efficiency, rather than practical efficiency factors like training speed and GPU memory consumption.

To address these limitations, (H. Cai et al., 2020) introduced the concept of "Reduce Memory, Not Parameters", suggesting a reduction in activations. LST (Sung et al., 2022) introduces

a memory efficiency approach that tries to decouple PEFT modules from the encoder-decoder T5 (Raffel et al., 2020). This emerging trend in NLP and CV research is increasingly pivoting from the earlier focus on EPEFT to DPEFT, as exemplified by recent studies (Y.-B. Lin, Sung, Lei, Bansal, & Bertasius, 2023; M. Xu, Zhang, Wei, Hu, & Bai, 2023a, 2023b).

For the usage of PEFT in the realm of modality-based sequential recommender systems (RS), significant progress has been demonstrated by M6-Rec (Z. Cui et al., 2022), Tallrec (Bao et al., 2023), and SSNA (Peng et al., 2023). These studies show that PEFT approaches can achieve performance comparable to traditional fine-tuning methods. Additionally, (Fu, Yuan, et al., 2024) provides an empirical study of adapters in MoRec. Although many current methods in the field continue to utilise the conventional EPEFT, they often overlook aspects of practical efficiency. The adoption of DPEFT in RS remains limited, with only a few exceptions, such as a concurrent preprint (Peng et al., 2023). Furthermore, these methods mainly concentrate on single-modality analysis, missing the opportunity to exploit the rich multimodal information available in RS. In contrast, VIP5 (Geng et al., 2023) explores the multimodal recommendation using adapters within the P5 backbone. However, its implementation primarily utilises adapters for text encoders, leaving the image encoder unchanged and dedicated solely to feature extraction. This approach differs from our focus, as adapting the image encoder incurs higher costs compared to the text encoder.

To the best of our knowledge, little research investigated practical efficiency issues of using DPEFT approaches for multimodal representation adaptation for recommendation tasks. Utilising multimodal information can certainly boost recommendation performance (Y. Li et al., 2023), while further increasing the practical efficiency issues. Therefore, the research of studying DPEFT approaches to address the real efficiency problems in this field is a novel and urgent direction.

3.3 Methodology

The overall framework of the proposed IISAN is shown in Figure 3.2. We depart from the mainstream multimodal representation learning models (Fu, Yuan, et al., 2024; Ge et al., 2021; J. Wang et al., 2022; C. Wu et al., 2021a; L. Wu et al., 2020; Yuan et al., 2023) to introduce a novel high-efficiency paradigm for personalised fine-tuning in sequential recommendation—referred to as decoupled parameter-efficient fine-tuning (decoupled PEFT, DPEFT). This innovative approach specifically addresses a critical aspect: the efficient adaptation of pre-trained large-scale multimodal models as item encoders. Initially, we propose to decouple the new trainable side adapted networks (SAN) and the frozen multimodal backbones. This decoupling is designed to optimise the extensive computation graph of backpropagation, effectively tackling challenges in training time and GPU memory encountered when transitioning large-scale models to downstream tasks. Leveraging the unique features of DPEFT, we introduce caching

strategies for IISAN to significantly enhance its practical efficiency. Furthermore, capitalising on the intrinsic nature of multimodal representations, the proposed IISAN architecture introduces two independent intra-modal SANs for visual and textual modality adaptation, along with an inter-modal SAN for handling interactions between the two modalities. The decoupled intra- and inter-modal SANs adeptly transfer pre-trained large-scale multimodal foundation models, such as BERT (Devlin et al., 2018), DEBERTA (P. He et al., 2021, 2020), ViT (Dosovitskiy et al., 2020), and CLIP (Radford, Kim, et al., 2021), to downstream multimodal recommendation tasks while maintaining multifaceted efficiency. Moreover, we pioneer a new efficiency metric—TPME—to evaluate the practical training efficiency of models, diverging from a reliance solely on trainable parameters.

3.3.1 IISAN

Given a recommendation dataset $\mathscr{D} = \{\mathscr{U}, \mathscr{V}\}$, where $\mathscr{U}$ and $\mathscr{V}$ denote the user and item sets, respectively, we aim to predict the next item that user u will interact with from their previous n interactions. In multimodal recommendation, each item v contains two modal representations, i.e. text (v^{text}) and corresponding image (v^{image}). We feed the texts and images of items into two pre-trained textual and visual foundation models, such as BERT (Devlin et al., 2018) for texts and ViT (Dosovitskiy et al., 2020) for images, which consist of an embedding layer (word embedding for texts and patch embedding for images) and multiple transformer blocks respectively. Through the pre-trained multimodal backbones, we can obtain the textual and visual embeddings from the embedding layers and the multiple hidden states ($\{h_i^{text}\}$ and $\{h_i^{image}\}$) from the transformer blocks ($\{TRM_i\}$), where i denotes the i-th layer of the backbone.

Our main innovation is the Intra- and Inter-modal Side Adapted Network (IISAN), which aims to maximise the use of knowledge from pretrained models for multimodal sequential recommendation under a decoupled PEFT paradigm. Specifically, we decouple the trainable parameters into three separate towers, which are the textual-modality side adapted network for text representation training, the visual-modality side adapted network for image representation training and an inter-modal side adapted network for image-text interactive representation training. Similar to Adapter (Houlsby et al., 2019), each side adapted network (SAN) consists of multiple SAN blocks (SANBs), each of which contains Upsample layers and downsample layers based on fully-connected network.[1] As shown in Figure 3.2, the structure of textual SAN, visual SAN and inter-SAN have structure symmetry, due to the consistency of the backbone models. Taking the textual SAN as an example, different from complex fusion ways (Ge, Chen, Shen, & Ji, 2019), a learnable gate mechanism is employed to fuse the information $\{h_{i-1}^{B^{intra}}\}$ from the last

[1] SANB, a network unit of SAN, utilises the network block similar to Adapter (Houlsby et al., 2019). However, unlike the Adapter's embedded approach, SANB is implemented in a decoupled way.

SANB and the current hidden state $\{h_i^{text}\}$, as the formula:

$$h_i^{B^{intra}} = SANB^{intra}\left(\mu_i^{text} * h_{i-1}^{B^{intra}} + (1-\mu_i^{text}) * h_i^{text}\right) \tag{3.1}$$

where $\mu_i^{text} \in [0,1]$. Note that the first SANB only inputs the text embeddings and the visual SAN employs similar operations. For the inter-SAN, we employ a similar gating method.

We design a fusing mechanism to fuse the hidden states of two modalities and sum the information from the last SANB, as the formula:

$$h_i^{B^{inter}} = SANB^{inter}\left(\beta_i * h_i^{image} + (1-\beta_i) * h_i^{text} + h_{i-1}^{B^{inter}}\right) \tag{3.2}$$

where $\beta_i \in [0,1]$. Note that the first inter-SANB only inputs the text embedding and the visual embeddings.

Additionally, to further enhance network efficiency and address issues of layer redundancy, we introduce a LayerDrop technique, like (A. Fan, Grave, & Joulin, 2019; Sung et al., 2022), to save the number of SAN blocks. Specifically, we group two transformer blocks together and drop the first hidden state to the SANs, which can save half the amount of SANBs. Moreover, we also explore different LayerDrop schemes in Section 3.6.4, i.e. dropping different hidden layer states, to achieve the best balance between efficiency and effect, which reflects the significance of different encoder layers for the final modality representation.

Finally, we obtain efficient new multimodal item representations from the intra- and inter-modal SANs, including the textual representations $\{e^{text}\}$, the visual representations $\{e^{image}\}$ and the textual-visual interacted representations $\{e^{inter}\}$. A linear-based fusion layer (FL) is added to ensure the consistency of the output dimensions of the item embedding and input dimensions of the sequential encoder for the final recommendation, as follows:

$$e^{item} = FL([e^{image} : e^{inter} : e^{text}]) \tag{3.3}$$

where [:] means the feature concatenation. Then, we input the $\boldsymbol{e}^{item}$ into the sequential encoders and calculate the final predicted score for user u to i-th item as $\hat{\boldsymbol{y}}_{ui}$, which is the product of the output sequential encoder and corresponding item embedding.

In terms of training details, we exploit the in-batch debiased Cross-Entropy loss function $\mathscr{L}_{CE}$ (Ji et al., 2023; Yi et al., 2019; Yuan et al., 2023) widely adopted in both academic literature and industrial systems.

$$D_{ui} = \exp(\hat{y}_{ui} - \log(p_i)) + \sum_{j\in[B], j\notin I_u} \exp(\hat{y}_{uj} - \log(p_j)) \tag{3.4}$$

$$\mathscr{L}_{CE} = -\sum_{u\in\mathscr{U}} \sum_{i\in[2,\ldots,n+1]} \log \frac{\exp(\hat{y}_{ui} - \log(p_i))}{D_{ui}} \tag{3.5}$$

where p is the popularity of the item, I_u and B stand for the item set interacted by user u and the batch. $n+1$ item denotes the predicted item for user u. D_{ui} is an auxiliary variable used to simplify Equation (3.5).

Introducing an innovative approach to enhance efficiency, we suggest a caching technique as a refinement strategy (Y. Chen, Liu, Zhao, & Zhu, 2020; Karedla, Love, & Wherry, 1994). As shown in Figure 3.3, this is possible because the decoupled PEFT mechanism separates the frozen pretrained backbone from the trainable adaptation network. This technique entails the storage and reuse of item hidden states extracted from pre-trained multimodal backbones, minimising the necessity for repeated forward passes of foundational models during training. This approach is not applicable to embedded PEFT methods because the input to each PEFT module changes after updates to preceding modules, making the hidden states unsuitable for caching.

3.3.2 New Efficiency metric: TPME

In this thesis, we propose a new composite metric[2] (termed TPME) to evaluate the practical efficiency of different PEFT methods, e.g. Adapter, LoRA, etc. It integrates the efficiencies of training time, trainable parameters, and GPU memory into a unified evaluation metric by adjustable practical factors. Specifically, assuming we evaluate the efficiency of i-th model among K models, TPME is calculated based on their training time $T = \{t_1, ..., t_K\}$, trainable parameters $P = \{p_1, ..., p_K\}$ and GPU memory $M = \{m_1, ..., m_K\}$ as follows:

$$t_i^{\text{norm}} = \frac{t_i - \min T}{\max T - \min T} \tag{3.6}$$

$$p_i^{\text{norm}} = \frac{p_i - \min P}{\max P - \min P} \tag{3.7}$$

$$m_i^{\text{norm}} = \frac{m_i - \min M}{\max M - \min M} \tag{3.8}$$

$$TPME_i = \alpha_1 t_i^{\text{norm}} + \alpha_2 p_i^{\text{norm}} + \alpha_3 m_i^{\text{norm}} \tag{3.9}$$

$$\alpha_1 + \alpha_2 + \alpha_3 = 1 \tag{3.10}$$

where α denotes the weighting assigned to each term, tailored to specific circumstances. For example, in scenarios where only a limited GPU capacity is available for model training, it's advisable to significantly augment the weight of M. Within the scope of this thesis, we've adjusted the values of α_1 and α_3 to 0.45, and α_2 to 0.1. This adjustment reflects our focus on two key practical aspects: training speed and memory efficiency. It is crucial to understand that

[2]The composite metric, frequently used in social statistics, measures levels across multiple dimensions. The well-known Human Development Index (HDI) (Sagar & Najam, 1998), employed by the United Nations, incorporates dimensions of education, longevity, and income. Our TPME follows the HDI's fundamental calculation principle, integrating various dimensions for a comprehensive evaluation.

Table 3.1: Efficiency Comparison of FFT and PEFT. In the Training-time metric, the FP/fp, BP/bp, and WU/wu represent the training time of the forward pass, backward propagation, and weight update respectively. Within the Parameter metric, TP/tp symbolises the model of trainable parameters. Concerning the GPU memory metric, MP/mp and A/a denote the model parameters and activations, respectively. Notably, in every instance, the lower case variables (e.g., fp, bp, wu) are significantly less than ($\ll$) their upper case counterparts (e.g., FP, BP, WU). We utilise "$>$" and "$=$" to denote relationships of greater and equal magnitude, respectively.

Metric	FFT		Adapter		LoRA		IISAN (Uncached)		IISAN (Cached)
Training-time	$O(FP+BP+WU)$	$>$	$O(FP+BP+wu)$	$=$	$O(FP+BP+wu)$	$>$	$O(FP+bp+wu)$	$>$	$O(fp+bp+wu)$
Parameter	$O(TP)$	$>$	$O(tp)$	$=$	$O(tp)$	$=$	$O(tp)$	$=$	$O(tp)$
GPU Memory	$O(MW+A)$	$=$	$O(MW+A)$	$=$	$O(MW+A)$	$>$	$O(MW+a)$	$>$	$O(mw+a)$

the Training-time, Parameter, and GPU Memory Efficiency (TPME) framework is specifically intended for the comparative analysis of a minimum of two methodologies. For example, this could involve a comparison between methodologies such as FFT and IISAN.

For each individual factor, it's crucial to ensure consistency in value derivation by conducting evaluations within the same experimental environment and setup, thus preserving the integrity and relevance of the results. An example of calculation is provided in Section 3.6.1.

3.4 Analysis of Efficiency

This section analyses the detailed composite efficiency of training downstream tasks, such as recommender systems and cross-modal retrieval, etc., with PEFTs based on pre-trained large-scale multimodal foundation models to compensate for filling in the absences in the existing literature[3]. While quantifying exact values in Transformer-based architectures is complex, we provide approximate values with an upper bound O, facilitating comparative analysis with various approaches in Table 3.1. In this comparison, we keep different dimensional variables inside O and diminish the smaller variables within the same dimension and constant coefficients using "$\approx$", following the algorithm's fundamental principles (Skiena, 1998).

3.4.1 Training-time Efficiency

Inspired by (Larochelle, Bengio, Louradour, & Lamblin, 2009; Safayenikoo & Akturk, 2021), we analyse the time spent on three key components, i.e. forward passes (FP), backward passes (BP) and weight updates (WU), of one training iteration. First, we define the aforementioned three components for the large-scale foundation model as $O(FP)$, $O(BP)$ and $O(WU)$, and for the smaller PEFT network as $O(fp)$, $O(bp)$, and $O(wu)$, where $FP \gg fp$, $BP \gg bp$ and $WU \gg wu$. Consequently, the total training time of one iteration in full fine-tuning (FFT) is

[3]Note that we only study from the model perspectives, other engineering approaches such as Quantisation applied in QLoRA (Dettmers et al., 2023) is not in our scope.

around $O(FP+BP+WU)$. For LoRA (E. J. Hu et al., 2021) and Adapter (Houlsby et al., 2019), their forward and backward pass involves both foundation models and PEFT components. In addition, since the foundation model is frozen, the weight update time includes only the PEFT model $O(wu)$. So the total training time for LoRA/Adapter is:

$$O(FP+fp+BP+bp+wu) \approx O(FP+BP+wu) \tag{3.11}$$

However, for IISAN, due to its decoupled structure, we can omit the backward propagation through the large foundation models. So the total training time is:

$$O(FP+fp+bp+wu) \approx O(FP+bp+wu) \tag{3.12}$$

Moreover, as discussed in Section 3.3.1, IISAN (Cached) approach can further save the forward pass time of the foundation models, i.e., training time is $O(fp+bp+wu)$.

3.4.2 Parameter Efficiency

The parameter efficiency is considered as a determinant of model efficiency in many research papers (Fu, Yuan, et al., 2024; Houlsby et al., 2019). The more trainable parameters a method has, the less parameter-efficient it becomes. Therefore, we analyse the entire backbone models with $O(TP)$ parameter sizes when FFT, while the PEFT components (including Adapter, LoRA, and IISAN, etc.) only contain the negligible $O(tp)$ trainable parameters, where $tp \ll TP$. As mentioned in (Fu, Yuan, et al., 2024; Houlsby et al., 2019), the main reason for the parameter efficiency being a concern is due to storage inefficiencies during downstream task model migration, especially on mobile. However, with the popularity of cloud-based storage, we believe that this should not be the only focus for recommendation model training, the training time and GPU memory efficiencies are more important in practice.

3.4.3 GPU Memory Efficiency

Different from existing methods, a detailed analysis of GPU Memory efficiency –on-GPU memory composite– is given to explain why traditional PEFTs cannot reduce much GPU memory, while IISAN can.

The consumption of GPU memory is primarily distributed across five aspects[4], i.e. (i) model weights, (ii) gradients, (iii) optimiser states, (iv) forward activations saved for gradient computation, and (vi) other components (temporary buffers, functionality-specific memory, etc.). The main sources of GPU memory usage are the first four components. The fifth component is usually a relatively small one that is omitted here.

[4]The Anatomy of Model's Memory Section from `https://huggingface.co/docs/transformers/v4.18.0/en/performance`

In particular, we first equate the model's gradients with its trainable weights[5], denoting them as $O(MW)$ for the backbone model and $O(mw)$ for the PEFT modules, where $MW \gg mw$. Secondly, the optimiser states take the standard Adam optimiser (Kingma & Ba, 2014) as an example. It doubles the parameter count for calculating first and second momentum orders, so the optimiser states are $O(2MW)$ for backbone models and $O(2mw)$ for PEFT modules. Additionally, we denote the activations for the backbone models and the PEFT modules by $O(A)$ and $O(a)$ respectively, where $A \gg a$. As stated in (H. Cai et al., 2020; L. Gao, Zhang, Han, & Callan, 2021; Gomez, Ren, Urtasun, & Grosse, 2017), the activations mainly depend on the computation graph's size[6]. This is because backpropagation requires storing the activations of the network in GPU memory, the memory cost that is proportional to the number of units in the network. Finally, we can represent the total GPU memory of FFT by:

$$O(MW + MW + 2MW + A) = O(4MW + A) \approx O(MW + A) \tag{3.13}$$

For the LoRA/Adapter-based approach, they reduce the trainable parameters from $O(MW)$ to $O(mw)$, but the computation graph as shown in Figure 3.1, has not been reduced. Therefore, the total GPU memory for LoRA/Adapter is:

$$O(MW + mw + mw + 2mw + A + a) = O(MW + 4mw + A + a) \approx O(MW + A) \tag{3.14}$$

Note that while the LoRA and adapter share the same theoretical complexity as the FFT, they can still save up to three times the GPU memory when the model parameters are the bottleneck[7], as highlighted in (E. J. Hu et al., 2021). On the other hand, IISAN can further save the computation graph of the backbone model. Consequently, the total GPU memory required for IISAN would be:

$$O(MW + mw + mw + 2mw + a) = O(MW + 4mw + a) \approx O(MW + a) \tag{3.15}$$

Furthermore, applying the caching strategies, the IISAN (cached) alleviates memory constraints for storing foundation model weights. Consequently, the GPU memory allocation for IISAN (Cached) is as follows:

$$O(mw + mw + 2mw + a) = O(4mw + a) \approx O(mw + a) \tag{3.16}$$

[5]Gradients normally equal to 4 bytes * number of trainable model weights, we denote them as equal value for simplicity

[6]Activations are also influenced by sequence length, hidden size and batch size, etc

[7]Typically, GPU memory usage is influenced by either model size or activations. In the context of Multimodality-based sequential recommendation scenarios, where sequences tend to be lengthy (item token length $*$ sequence length $*$ batch size $*$ types of modalities), activations often become the bottleneck. Consequently, IISAN demonstrates significant efficiency in the multimodal recommendation, particularly in scenarios characterised by extensive activations.

Table 3.2: Dataset Description

Dataset	Users	Items	Interaction	Content
Scientific	12,076	20,314	81,711	Text&Image
Instrument	10,000	19,246	75,022	Text&Image
Office	10,000	22,785	71,276	Text&Image

Table 3.3: Comparisons of efficiency-performance balance. Two key categories of evaluation metrics are used, i.e. “Performance" and “Efficiency". All results of HR@10 and NDCG@10 are denoted in the percentage (%). The ’Training’, ’Parameter’, and ’GPU Memory’ denote training time (seconds for each epoch), trainable parameters and max GPU memory usage, respectively. “Relative Improvement" means the result differential between IISAN and FFT. Larger values (↑) of the Performance metric indicate better performance, while smaller values (↓) of the Efficiency metric reflect better efficiency. “*” denotes that the improvements of IISAN compared with FFT are significant at the level of 0.05 with paired T-test. To clearly represent the difference between uncached IISAN and cached IISAN (in blue), we divided them into two separate columns. The best results are in bold.

Dataset	**Category**	**Metric**	**FFT**	**Adapter**	**LoRA**	**BitFit**	**IISAN (ours)**		**Relative Improvement**	
							Uncached	Cached	Uncached	Cached
Scientific	Performance(↑)	HR@10	6.62	6.61	6.62	6.37	**6.83***		+3.07%	
		NDCG@10	4.06	3.91	4.09	3.76	**4.14***		+1.93%	
	Efficiency(↓)	Training-time	443s	354s	378s	403s	**179s**	**22s**	-59.59%	-95.03%
		Parameter	195M	5M	0.8M	**0.4M**	4M	4M	-97.89%	-97.89%
		GPU Memory	46.76G	37.82G	39.07G	36.97G	**8.32G**	**3.11G**	-82.19%	-93.35%
		TPME	100%	71.50%	75.14%	70.82%	**22.34%**	**0.19%**	-78.10%	-99.80%
Instrument	Performance(↑)	HR@10	8.76	8.82	8.43	8.65	**9.06***		+3.31%	
		NDCG@10	6.76	6.83	6.64	6.71	**7.01***		+3.57%	
	Efficiency(↓)	Training-time	369s	295s	308s	287s	**142s**	**18s**	-61.52%	-95.12%
		Parameter	195M	5M	0.8M	**0.4M**	4M	4M	-97.89%	-97.89%
		GPU Memory	46.76G	37.82G	39.07G	36.97G	**8.32G**	**3.11G**	-82.19%	-93.35%
		TPME	100%	71.55%	74.28%	69.40%	**21.46%**	**0.19%**	-78.54%	-99.81%
Office	Performance(↑)	HR@10	6.30	6.65	6.55	6.78	**6.80***		+7.35%	
		NDCG@10	4.58	4.85	4.78	4.87	**4.92***		+6.91%	
	Efficiency(↓)	Training-time	355s	296s	308s	288s	**140s**	**19s**	-60.56%	-94.65%
		Parameter	195M	5M	0.8M	**0.4M**	4M	4M	-97.89%	-97.89%
		GPU Memory	46.76G	37.82G	39.07G	36.97G	**8.32G**	**3.11G**	-82.19%	-93.35%
		TPME	100%	73.12%	75.80%	70.93%	**21.77%**	**0.19%**	-78.23%	-99.81%

3.5 Experiment Setup

Datasets. In this thesis, each item for a user sequence is represented by its raw image and text modality. We perform the preprocessing following (Fu, Yuan, et al., 2024; R. Li et al., 2025; Yuan et al., 2023). To evaluate our methods on items with both raw image and text modality, we adopt the Amazon review datasets (J. Ni et al., 2019). We adopt three widely used datasets from it, "Industrial and Scientific", "Musical Instruments", and "Office Products".[8] Due to the constraint of GPU memory, we follow (Fu, Yuan, et al., 2024; Yuan et al., 2023) and set the sequence length to 10. After the preprocessing, the details of the datasets are shown in Table 3.2.
Performance Evaluations. Drawing on prior studies (X. He et al., 2017; Yuan et al., 2023), our approach involves dividing the datasets using a leave-one-out method. This involves using the final item in the interaction sequence for testing, the item before the last is used for validation, and the remaining items for training purposes. For assessment, we use HR@10 (Hit Ratio) and NDCG@10 (Normalised Discounted Cumulative Gain) (Y. Hou, He, et al., 2022; Yuan et al., 2023) as the primary metrics. Unless otherwise specified, all reported results pertain to the test set. Additionally, it's important to note that the predicted item is compared against the entire set of items (Krichene & Rendle, 2020).
Implementation Details. The "bert-base-uncased", "vit-base-patch16-224", "clip-vit-base-patch16", and "deberta-v3-base" from the Huggingface platform (Jain, 2022) are used as the text and image encoders in this research, respectively. The dimension of hidden representations of the sequential encoder is searched in {32,64,128} and set to 64. The number of the sequential encoders' Transformer blocks and attention heads is fixed to 2. We apply Adam as the optimiser without weight decay throughout the experiments and extensively search the learning rate from 1e-5 to 1e-3 while keeping the dropout probability at 0.1.

We search the batch size in {8, 16, 32, 64, 128} and obtain the largest optimal batch size to maximise GPU memory. The hidden dimension of the adapter networks and LoRA rank are carefully searched in {8, 16, 32, 64, 128}, respectively. All hyper-parameters are determined according to the performance in the validation data. All results are reported in the test set. We perform all the experiments on an A6000 GPU.

3.6 Experiment

- **RQ3.1:** How does the performance of the proposed IISAN compare to FFT and existing PEFT methods? Can IISAN achieve significant efficiency without sacrificing performance?

[8] We only keep the items with both images and text information. Due to the massive amount of data for "Office" and "Instrument", we randomly sampled 10,000 users from the remaining datasets to align with "Scientific" following (Fu, Yuan, et al., 2024).

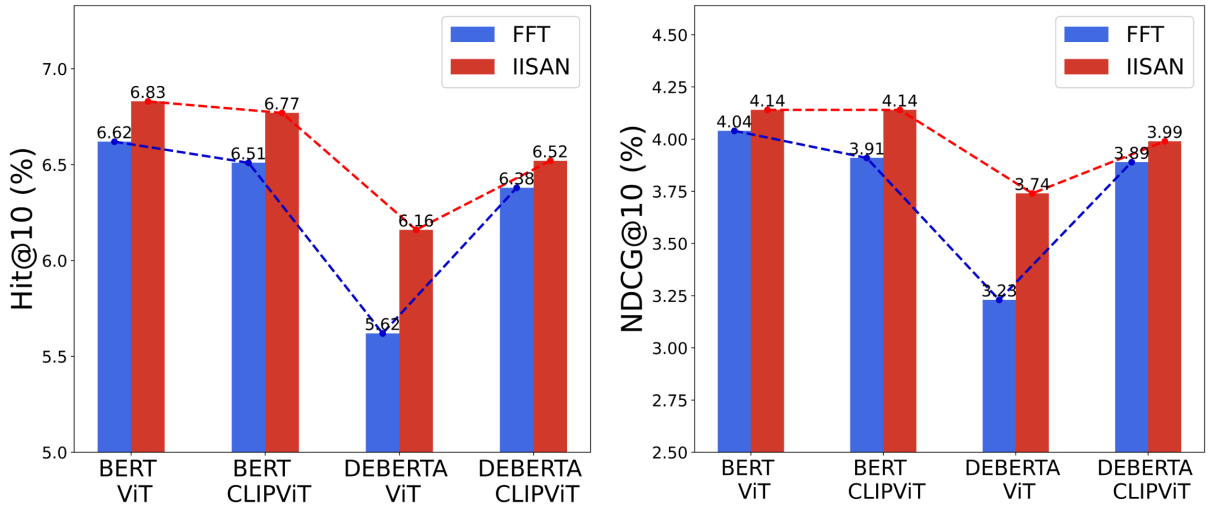


Figure 3.4: Performance comparisons between FFT and IISAN with different multimodal backbones on Scientific dataset.

- **RQ3.2:** How robust is IISAN on different multimodal backbones?
- **RQ3.3:** How do the components of the proposed IISAN affect the recommendation performance and efficiency, including the LayerDrop, modality selection, gated fusion, and SANB implementation?
- **RQ3.4:** This proposed IISAN explores multimodal scenarios; does it have any advantages over unimodal methods?

3.6.1 Efficiency-Performance Balance (RQ3.1)

We perform extensive experiments on three multimodal sequential recommendation datasets and compare IISAN with full fine-tuning (FFT) based on (Fu, Yuan, et al., 2024) in terms of model efficiency and performance. Furthermore, we also include the state-of-the-art PEFT approaches, such as Adapter (Houlsby et al., 2019), LoRA (E. J. Hu et al., 2021), and BitFit (Zaken et al., 2021). The Adapter (Houlsby et al., 2019) adopts the classic Houlsby architecture which has been proven to be the most effective PEFT approach for unimodality-based recommendation in (Fu, Yuan, et al., 2024). The LoRA (E. J. Hu et al., 2021) is widely used due to its smaller trainable parameters. BitFit, a common baseline for PEFT-based recommendation research (E. J. Hu et al., 2021; Karimi Mahabadi et al., 2021; Sung et al., 2022), updates only the bias layers within the entire network to achieve the minimum trainable parameters. Notable, the current mainstream PEFT method in the recommendation is embedded PEFT (EPEFT) (Houlsby et al., 2019; E. J. Hu et al., 2021; Zaken et al., 2021), and the proposed IISAN belongs to the decoupled PEFT (DPEFT).

Table 3.3 indicates that the proposed IISAN not only surpasses the performance of the FFT model on all datasets but only uses 22%

of the relative costs in terms of TPME. Furthermore, IISAN consistently outperforms all PEFT methods in both performance and practical efficiency under the same batch size (32). In particular, compared to the FFT model, using IISAN significantly improves the recommendation performances, e.g. by 3.07%, 3.31%, and 7.35% of HR@10 and by 1.93%, 3.57%, and 6.91% of NDCG@10 on Scientific, Instrument, and Office datasets, respectively. For model practical efficiency comparisons, IISAN achieves the best efficiency, whether compared with FFT or any EPEFT models. Without the caching strategy, IISAN can reduce the Training time and GPU memory by around 60% and 82%. Equipped with the caching strategy, the numbers even rise to 94% and 93%. While the EPEFT methods significantly reduce the number of trainable parameters, they do not substantially decrease the crucial aspects of model training, i.e. training time and GPU memory costs. According to the new balanced efficiency metric (TPME), IISAN only uses about 22% TPME cost of FFT, but the Adapter, LoRA and BitFit require around 71%, 75% and 70% respectively. Furthermore, IISAN with a caching strategy further reduces training costs and only needs 0.2% TPME.

(Answer to RQ3.1) IISAN can achieve the most competitive performance with the highest efficiency and the TPME effectively uncovers the practical efficiency levels of each model. These experiments suggest that IISAN is a promising and efficient choice for representation learning tasks that involve multimodal item representation for recommendation.

3.6.2 Robustness Evaluation (RQ3.2)

To better understand the robustness of the proposed IISAN method, we evaluate it with four different combinations of state-of-the-art multimodal encoders based on (Fu, Yuan, et al., 2024), which includes BERT+ViT, BERT+CLIPViT, DEBERTA+ViT, and DEBERTA+CLIPViT in Figure 3.4. We observe that on the Scientific dataset, although the contribution of different backbones to multimodal recommendation is inconsistent, the proposed IISAN can always remain ahead of the FFT model. For example, when changing the text encoder and image encoder to DEBERTA and CLIPViT respectively, IISAN still exceeds the FFT model by 0.12 in H@10 and 0.10 in NDCG@10.

(Answer to RQ3.2) These results validate that IISAN maintains excellent robustness on different fundamental models. This has reference implications for the model migration of IISAN.

3.6.3 Ablation Study (RQ3.3)

To comprehensively answer the RQ3, we perform extensive ablation studies. We demonstrate how each component affects the overall performance and efficiency of IISAN. Note that the decoupled structure and caching strategy are the main contributors to the efficiency gains as described in Section 3.4. Table 3.4 presents component ablation studies focusing on the various

Table 3.4: Ablation study for IISAN on Scientific Dataset. IISAN mainly contains four key components: LayerDrop, Modality Gate, Intra- and Inter-modal towers. "-" represents TPME is not applicable for the frozen Backbone.

Method	HR@10	NDCG@10	TPME
-w/o LayerDrop	6.73	4.04	22.81%
-w/o Modality Gate	6.58	3.89	22.02%
Frozen Backbone	6.00	3.53	-
-w/o Inter-modality	6.38	3.89	21.31%
-w/o Intra-modality	6.41	3.83	20.67%
IISAN	6.83	4.14	22.34%

Table 3.5: LayerDrop in IISAN on Scientific Dataset. The number of blocks in the Method column represents that keep this number of blocks and drop the others.

Method	Number of blocks	HR@10	NDCG@10	TPME
IISAN	2 blocks	6.46	3.88	22.05%
	3 blocks	6.79	4.12	22.06%
	4 blocks	6.57	3.98	22.10%
	6 blocks	**6.83**	**4.14**	22.34%
	12 blocks	6.73	4.04	22.81%

modules within IISAN, including (1) Modality selection, (2) LayerDrop strategies, (3) Gated fusion, and (4) Implementation of SANB.

(1) Modality selection. Table 3.4 highlights that although employing separate intra-modal SAN (Line 4) or inter-modal SAN (Line 5) can slightly improve efficiency, it will clearly reduce the recommendation effects. However, compared with the frozen pretrained backbones in Line 3, both the intra-modal SAN and the inter-modal SAN significantly improve performance, and their combination performs best. It reflects that intra- or inter-modal SAN can improve feature adaptation within each modality and inter-modal feature interaction adaptation.

(2) LayerDrop strategies. IISAN's layers exhibit a degree of redundancy, allowing the utilisation of LayerDrop to enhance performance effectively. First, when we remove the LayerDrop, both HR@10 and NDCG@10 decrease absolutely by 0.1% and the cost slightly rises by 0.5% on TPME, which demonstrates the effectiveness of LayerDrop. Second, we study the effect of different LayerDrop strategies as described in Table 3.5. We adopt the even number $(2, 4, 6, ...12)$ transformer-blocks (6 blocks), which skip the odd number layers. It achieves the best performance-efficiency balance.

(3) Gated fusion. To deepen our understanding of the differences in information usage between textual and visual modalities in our research, we analysed the gate weights at the optimal checkpoint. These gate values, ranging from 0 to 1, imply that values above 0.5 denote a stronger focus on a specific modality as discussed in Section 3.3.1. The data indicate that in the realm of multimodal recommendations, the weights assigned to visual modality gates consistently ranged from 0.2 to 0.4. This trend suggests a predominant reliance on textual modality within our multimodal approach.

Table 3.6: Implementation of SANB (Side Adapted Network Block) on Scientific Dataset. We get the TPME of uncached approach.

Model	Implementation	HR@10	NDCG@10	TPME
SANB	Adapter Block	6.73	4.04	22.81%
	PHM	5.95	3.59	22.34%
	LowRank	4.79	2.83	22.47%

(4) Implementation of SANB. Table 3.6 demonstrates the notable superiority of the classic adapter block in terms of performance compared with other more recent PHM (Parameterised Hypercomplex Multiplication) block (Karimi Mahabadi et al., 2021) and Low-Rank adapter blocks (Yin et al., 2023). Their performance, as indicated by HR@10, is significantly lower than 6.62 (FFT), suggesting that their adaptation is unsuccessful. This phenomenon is corroborated by findings in (Fu, Yuan, et al., 2024) where PHM shows a performance drop compared to traditional adapters. Notably, both PHM and LowRank models, despite having merely half the trainable parameters of an adapter block, exhibit comparable training time and GPU memory, resulting in similar TPME.

(Answer to RQ3.3) In this section, we conclude with three key findings of the IISAN's components: (1) Optimal performance is achieved through the placement of both intra- and inter-modal Side Adapted Networks (SANs). (2) The best efficiency and performance balance is achieved by dropping half of the SANBs in the Intra- and inter-modal SANs. (3) Text-image interaction is effective, but text modality plays a more crucial role in recommendation, where inter-modal SAN can effectively maintain the dominance of text modality and integrate image information.

3.6.4 Multimodality vs. Unimodality (RQ3.4)

The concept of PEFT in modality-based recommendation research is relatively nascent, with limited literature directly comparing multimodal and unimodal scenarios. Driven by this curiosity, we conduct experiments on the performance of FFT and various PEFTs across different modality scenarios.

Table 3.7 presents comparisons between unimodal and multimodal scenarios. First, the PEFT models (such as Adapter, LoRA, and BitFit) exhibit performance comparable to FFT in text-based and multimodal scenarios, but poor image-based performance. Additionally, the text-based adapter significantly outperforms FFT, indicating its high suitability for text-based recommendation. This may be caused by the inherent bias in the modal transfer from the foundation models in multimodal recommendation tasks where images are more difficult. This observation is corroborated by our analysis of information usage in the gated fusion (in Section 3.6.3), where text modality is dominant in the inter-modal interaction. Second, each adaptation approach based on multimodal representations demonstrates superior performance, underscor-

Table 3.7: Multimodality vs. Unimodality.

Modality	Method	HR@10	NDCG@10
Image	FFT	5.81	3.47
	Adapter	5.47	3.18
	LoRA	5.49	3.34
	BitFit	4.97	3.03
Text	FFT	5.80	3.44
	Adapter	6.55	3.88
	LoRA	6.14	3.61
	BitFit	5.92	3.56
Multimodality	FFT	6.62	4.06
	Adapter	6.61	3.91
	LoRA	6.62	4.09
	BitFit	6.37	3.76
	IISAN (ours)	**6.83**	**4.14**

ing the importance of integrating multiple modalities. In particular, IISAN achieves the best results due to the novel intra- and inter-modal adapted networks.

(Answer to RQ3.4) The findings suggest that multimodality is more advantageous than relying solely on unimodality.

3.7 Conclusion and Future Work

In this study, we present a novel Decoupled-PEFT architecture, named IISAN, for recommendation task adaptation of pre-trained large-scale multimodal foundation models. IISAN leverages advantages inherent in DPEFT to separate the trainable intra- and inter-modal adaption networks from the multimodal backbones, thus minimising the computational graph and allowing for caching strategies. These allow IISAN to optimise in terms of model practical efficiency. Besides, the new intra- and inter-modal SANs in IISAN achieve comparable performance to the full fine-tuning model by combining both intra- and inter-modal information adaptive interactions. In addition, we introduce a balanced efficiency metric, –TPME–, to evaluate the multi-faceted practical model efficiency between different methods. Finally, experimental results on three recommendation datasets have shown the superiority of both efficiency and effectiveness of IISAN. And the efficiency analysis also proves its high efficiency.

Future work includes the exploration of more potential applications, such as multimodal retrieval(Ge, Chen, Xu, Tao, & Jose, 2023; Ge, Xu, et al., 2024) and visual question answering (VQA) (Y. Zhou et al., 2021), etc., by the IISAN paradigm. Moreover, many more modality representations can be applied by the novel intra- and inter-modal SANs, e.g. image, text audio, etc., to further adapt the multimodal real-world scenarios.

Chapter 4

IISAN-Versa: Scalable Adaptation of Asymmetric Multimodal Encoders

Publication Note

This chapter is adapted from: Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Kaiwen Zheng, Yongxin Ni, and Joemon M. Jose. Efficient and Effective Adaptation of Multimodal Foundation Models in Sequential Recommendation. TKDE 2025.

Chapter Positioning

Building on IISAN, this chapter studies what happens when the foundation encoders are larger, asymmetric, and more varied in scale. IISAN-Versa extends the thesis from efficient adaptation of a fixed multimodal pair toward scalable adaptation of modern text and vision encoders whose depths, dimensions, and representation strengths differ.

4.1 Introduction

Recent advancements in recommendation algorithms have demonstrated that utilising powerful large language models (LLMs) and vision encoders like GPT, Llama[1], and CLIP (Radford, Kim, et al., 2021) can achieve state-of-the-art recommendation performance (Y. Hou, He, et al., 2022; Y. Li et al., 2024, 2023; Q. Liu, Chen, et al., 2024; Yuan et al., 2023). These types of models typically combine a recommender model as the user encoder with multimodal foundation models (MFMs)[2] to encode the items. Many studies (Y. Li et al., 2024, 2023; Yuan et al.,

[1] `https://www.llama.com/`, `https://openai.com/index/hello-gpt-4o/`

[2] The definition of multimodal foundation models (MFMs) may vary. In this thesis, we focus on separately pre-trained text and image encoders, following the approach in (Fei et al., 2022; Radford, Kim, et al., 2021), where the representations of image and text are learned in an end-to-end manner. Other MFM paradigms, such as (Alayrac et

2023) have shown that fine-tuning the item encoder can lead to optimal performance. However, these approaches are highly inefficient due to the immense computational resources required to fine-tune such large encoders.

A popular paradigm (Fu, Yuan, et al., 2024) for resolving the efficiency problem includes methods such as Adapter (Houlsby et al., 2019) and LoRA (E. J. Hu et al., 2021), BitFit (Zaken et al., 2021). These approaches integrate tunable neural network modules into the backbone multimodal foundation models, collectively referred to as embedded parameter-efficient fine-tuning (EPEFT) (Fu, Ge, et al., 2024). While EPEFT methods have gained popularity for addressing the efficiency of trainable parameters, they still face practical efficiency challenges, such as training time and GPU memory consumption. The introduction of the IISAN (Fu, Ge, et al., 2024), with its decoupled structure and caching strategy for multimodal encoders, addresses these practical efficiency issues, outperforming both traditional full fine-tuning and EPEFT methods in terms of both performance and efficiency. We refer to this decoupled, parameter-efficient fine-tuning approach as DPEFT. It significantly reduces GPU memory usage by up to 15 times compared to FFT and 12 times compared to EPEFT (Adapter/LoRA). Additionally, it accelerates training time per epoch by up to 20 times compared to FFT and 16 times compared to EPEFT.

IISAN leverages both intra- and inter-modal information from multimodal foundation models by exploiting the hidden states within the layers of backbone models to achieve fine-grained information merging. Despite these advancements, merging multimodal hidden states from two different encoders presents two limitations: **(1) it can only be conducted for symmetrical multimodal transformer encoders; (2) due to the first limitation, it is hard to explore whether scaling the text encoder in IISAN with recent state-of-the-art LLMs could yield performance improvements.** These limitations do not align with the current research trend, where text encoders are often larger and more complex than visual transformers, as shown in Figure 4.1, which compares the state-of-the-art text and visual transformers. While vision transformers like ViT-22B (Dehghani et al., 2023) have billions of parameters, they are all closed-source models. Therefore, the most commonly used vision transformers currently have fewer than 1 billion parameters (Dosovitskiy et al., 2020). Larger pre-trained transformers generally offer better performance according to the scaling effect (Rosenfeld, 2021). To maximise the potential of pre-trained models, it is advantageous to use larger models. However, finding a vision transformer that matches the size of a text encoder (LLM) is nearly impossible due to the disparity in model sizes. Therefore, addressing the issue of asymmetrical merging is of paramount importance.

In this thesis, we present an extension of the IISAN framework (Fu, Ge, et al., 2024) called the Intra- and Inter-model Side Adapted Network for Versatile Multimodal Representation (IISAN-Versa). We specifically categorise IISAN-Versa into two variants: the symmetrical

al., 2022; Geng et al., 2023), which use a large language model (LLM) as the backbone and input image encoder features into the LLM, are beyond the scope of this thesis.

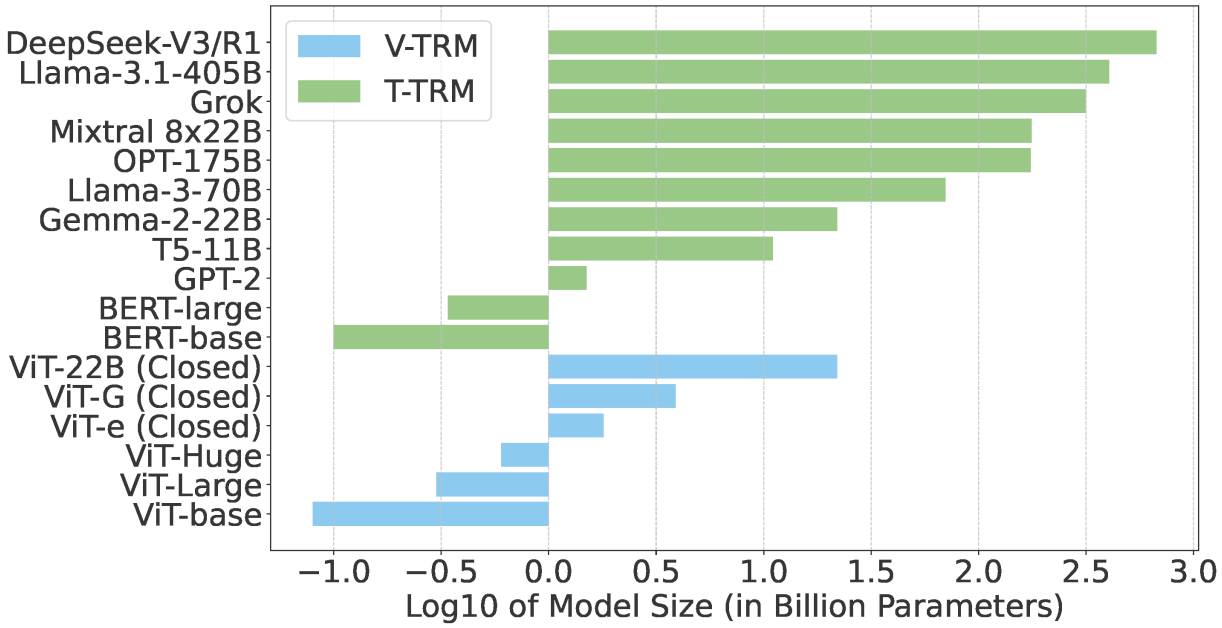


Figure 4.1: Comparisons of model sizes for various popular text transformers (T-TRM) and visual transformers (V-TRM) on a logarithmic scale. Closed-source models are indicated by "(Closed)" in the figure; all other models are open-source.

IISAN-Versa (IISAN-VS), which follows the standard IISAN configuration, and the newly proposed asymmetrical IISAN-Versa (IISAN-VA), which is capable of incorporating a larger text encoder to address asymmetry between text and vision pre-trained models. For IISAN-VA, we address the challenge of asymmetric multimodal backbone networks by proposing a simple yet effective strategy that combines group layer-dropping with dimension transformation alignment. The IISAN-VA variant, with a larger text encoder, significantly outperforms the standard symmetrical IISAN-VS, which uses BERT-base text encoders, while maintaining high efficiency. This demonstrates IISAN-Versa's effectiveness in accommodating a larger and more powerful model. Moreover, we extend and validate the IISAN-Versa framework to a wider range of multimodal scenarios, particularly focusing on multimodal text, thereby validating its scalability. This extension involves the integration of diverse text forms (named multimodal text), such as titles, text captions from image covers, and video content, utilising an open-source video recommendation dataset MicroLens (Y. Ni et al., 2023).

We summarise the main changes made in this study compared to our published conference version in SIGIR2024 (Fu, Ge, et al., 2024) as follows:

- We revise the Introduction and Related Work sections to emphasise the motivation of achieving the versatility of the new IISAN-Versa, rather than focusing only on the efficiency of IISAN.

- We expand the methodology section by categorising IISAN (Fu, Ge, et al., 2024) as the symmetrical IISAN-Versa (IISAN-VS) and introduce the asymmetrical IISAN-Versa

(IISAN-VA) to incorporate LLMs as encoders.

- We validate the effectiveness of the IISAN-VA, which incorporates state-of-the-art LLMs as text encoders, demonstrating significant performance improvements. Furthermore, we fully explore the effect of scaling laws on text encoders, observing that larger pre-trained text encoders typically lead to better performance.
- To gain a clearer understanding of efficiency improvement during the training process, this thesis analyses it from the perspectives of forward and backward propagation.
- We further validate IISAN-Versa's performance on MicroLens, a public multimodal recommendation benchmark, achieving state-of-the-art performance.
- We also utilise pre-trained captioning models to generate multimodal texts from raw images and videos from MicroLens and validate IISAN-Versa's versatility in this multimodal text scenario.

Our main contributions are the following:

- We propose a versatile paradigm, IISAN-Versa, which adapts existing mainstream symmetric and asymmetric multimodal foundation models to enable efficient and effective multimodal sequential recommendation. It ensures the flexibility of multi-structure model adaptation and the efficiency of PEFT through minimal modifications of dimension transformation layers with LayerDrop.
- We identify a scaling effect in our novel asymmetric IISAN-Versa by scaling the text encoder from a smaller language model to a larger language model. This provides a new inspiration that scaling LLMs with appropriate approaches, e.g. our IISAN-Versa, can effectively and efficiently improve multimodal recommendation.
- We construct a new multimodal text recommendation scenario by extending the open-source MicroLens dataset, i.e., generating captions from its original videos and images using a pre-trained video captioning generator, to validate the multi-scenario adaptation capabilities of the proposed IISAN-Versa. The new reconstructed dataset will be released to facilitate future research for the multimodal recommendation community.

4.2 Related Work

Multimodal Foundation Models. Recent work in multimodal learning focuses on leveraging pre-trained models for downstream tasks to enhance performance and reduce pre-training costs (Z. Chen et al., 2024). BERT (Devlin et al., 2018) pioneered the pretraining and fine-tuning paradigm in NLP, using Masked Language Modelling and the Next Sentence Prediction

to understand context and sentence relationships. Following this, the Vision Transformer (ViT) (Dosovitskiy et al., 2020) adapted transformer architecture for image classification, demonstrating competitive performance by treating image patches as tokens. CLIP (Radford, Kim, et al., 2021) further bridged vision and language, training a multimodal foundation model to match images with their textual descriptions through contrastive learning, enabling powerful zero-shot learning capabilities (Radford, Kim, et al., 2021). The evolution continued with large-scale models like GPT-4, and multimodal variants such as Flamingo (Alayrac et al., 2022). These models, characterised by their vast scale and diverse training data, perform a wide array of tasks, pushing the boundaries of AI's multimodal understanding and processing capabilities.

Modality-based Sequential Recommendation (MoRec). The Recommender System community has shown a growing interest in incorporating various modality information into recommendation systems (Y. Cheng et al., 2023; Q. Liu, Zhu, et al., 2024; Y. Ni et al., 2023; J. Wang et al., 2022; Yuan et al., 2023). These systems utilise large-scale multimodal foundation models (Devlin et al., 2018; Radford, Kim, et al., 2021; J. Wang et al., 2022) from NLP and CV (Dosovitskiy et al., 2020; Z. Liu et al., 2021) to encode text and image data. The sequential encoder remains consistent with traditional architectures like SASRec (Kang & McAuley, 2018), GRU4Rec (Hidasi et al., 2015), and BERT4Rec (F. Sun et al., 2019). Furthermore, models such as IDA-SR (Mu et al., 2022), UniSRec (Y. Hou, Mu, et al., 2022), and VQ-Rec (Y. Hou, He, et al., 2022) have advanced MoRec by developing item representations using NLP foundation models. For image representation in recommender systems, studies such as (Y. Wei et al., 2019) and (L. Meng et al., 2020) have integrated image features extracted from ResNet-50 (K. He et al., 2016) into the models.

Furthermore, research (Y. Li et al., 2023; Y. Ni et al., 2023; Yuan et al., 2023) has shown that the MoRec framework, applied through end-to-end learning, significantly outperforms previous methods that rely on offline feature extraction. Specifically, (Y. Li et al., 2024, 2023) emphasised that end-to-end training, which integrates both image and text modalities, significantly surpasses the methods that rely solely on a single modality. Although directly learning the raw content showcases tremendous performance, a significant limitation of these end-to-end studies is their ongoing reliance on fully fine-tuning large-scale multimodal encoders, which often leads to inefficiency.

Parameter-efficient Fine-tuning (PEFT). In the fields of Natural Language Processing (NLP) and Computer Vision (CV), substantial research efforts are focused on investigating Parameter-Efficient Fine-Tuning (PEFT) methods. These methods aim to tackle the challenge posed by the vast number of trainable parameters in large-scale pre-trained models. Foundational works such as Adapter (Houlsby et al., 2019), LoRA (E. J. Hu et al., 2021), and BitFit (Zaken et al., 2021) have laid the groundwork in this area. Building on these foundational works, a variety of alternative strategies have emerged, as seen in studies like (R. Wang et al., 2020). Despite the advancements, these approaches primarily utilise Embedded PEFT (EPEFT) and focus largely

on parameter efficiency, often overlooking practical efficiency concerns such as training speed and GPU memory usage.

To overcome these limitations, (H. Cai et al., 2020) proposed the concept of “Reduce Memory, Not Parameters," advocating for a reduction in activations. Additionally, LST (Sung et al., 2022) offers a memory-efficient strategy aimed at decoupling PEFT modules. A notable shift in recent NLP and CV research is the transition from an emphasis on Embedded PEFT (EPEFT) to Decoupled PEFT (DPEFT), as illustrated by recent works (Y.-B. Lin et al., 2023; M. Xu et al., 2023a, 2023b). This shift reflects an evolving understanding that focuses more on practical efficiency instead of trainable parameters.

In modality-based sequential recommender systems (RS), PEFT methods have also shown significant progress, as demonstrated by M6-Rec (Z. Cui et al., 2022), Tallrec (Bao et al., 2023), and SSNA (Peng et al., 2023). These studies indicate that PEFT approaches can achieve performance levels comparable to traditional fine-tuning methods. However, most such methods still rely on conventional EPEFT, often ignoring practical efficiency aspects. The adoption of DPEFT in recommender systems remains limited. Moreover, these methods generally focus on single-modality analysis, missing opportunities to leverage the abundant multimodal information in recommendation scenarios. VIP5 (Geng et al., 2023) investigates multimodal recommendations using adapters within the P5 framework, though it primarily utilises adapters for text encoders and leaves image encoders unchanged for feature extraction. This contrasts with our approach, as adapting multiple modality encoders incurs greater costs than only using the text encoder. We initially introduced IISAN (Fu, Ge, et al., 2024), an efficient Decoupled PEFT approach designed for symmetrical multimodal foundational models in sequential recommendation. However, it has limitations in effectively handling asymmetrical MFMs, preventing it from leveraging the capabilities of recent state-of-the-art large language models.

Overall, little research has delved into the practical efficiency challenges of using DPEFT approaches for versatile multimodal representation adaptation in recommendation tasks. While multimodal information can significantly enhance recommendation performance (Y. Li et al., 2023), it also intensifies practical efficiency challenges. Consequently, investigating DPEFT strategies to address real efficiency issues in this domain represents a novel and pressing research direction. Although IISAN partially resolves this problem, it still suffers from the limitation of incorporating asymmetrical MFMs. In this thesis, we extend IISAN into a more versatile framework that improves its ability to incorporate large and asymmetrical MFMs.

4.3 Methodology

4.3.1 Overview

The proposed IISAN-Versa framework, shown in Figure 4.2, is designed to adapt both symmetrical and asymmetrical structures of multimodal foundation models, enabling versatile representation generation. By separating the new trainable side-adapted networks (SAN) from multimodal backbones, we optimise the backpropagation computation graph, addressing practical efficiency in adapting large-scale multimodal foundation models to downstream tasks, which will be further explained in Section 4.4. Leveraging the decoupling features of IISAN-Versa, the hidden states of the multimodal foundation models can be cached to enhance IISAN-Versa's efficiency. The architecture introduces independent intra-modal SANs for visual and textual modalities and an inter-modal SAN for managing their interactions. This framework allows the efficient adaptation of pre-trained MFMs like BERT, Llama, and ViT to downstream multimodal recommendation tasks.

In this thesis, IISAN-Versa's innovation lies in proposing versatile intra- and inter-modal side-adapted networks to efficiently and effectively adapt both symmetrical and asymmetrical multimodal foundation models. For symmetrical MFMs, IISAN-Versa adopts balanced adaptation following IISAN (Fu, Ge, et al., 2024). The main challenge tackled in this thesis is the adaptation of asymmetrical structures of MFMs, which present two primary issues: (1) inconsistent embedding dimensions and (2) imbalanced hidden layers. This thesis is dedicated to proposing simple yet effective solutions to resolve them. To address the first issue, we adopt a dimension transformation layer to align the differences between the large language model (LLM) and the smaller image encoder. For the second issue, we employ a group LayerDrop strategy to resolve the imbalance in hidden layers.

4.3.2 Problem Formulation

Given a recommendation dataset $\mathscr{D} = \{\mathscr{U}, \mathscr{V}\}$, where $\mathscr{U}$ and $\mathscr{V}$ denote the set of users and the set of items, respectively, we aim to predict the next item interacted with by a user u in a multimodal sequential recommendation task by exploiting their n past behaviours. In multimodal recommendation, each item v contains two modal representations, i.e., text (v^{t}) and corresponding image (v^{v}). These texts and images are fed into two pre-trained multimodal foundation models, such as BERT (Devlin et al., 2018) or Llama (Touvron et al., 2023) for texts and ViT (Dosovitskiy et al., 2020) for images, which consist of an embedding layer (word embedding for texts and patch embedding for images) and multiple transformer blocks, respectively. Through these pre-trained multimodal backbones, we obtain textual and visual embeddings from the embedding layers and multiple hidden states ($\{h_i^{\mathrm{t}}\}$ and $\{h_i^{\mathrm{v}}\}$) from the transformer blocks ($\{TRM_i\}$), where i denotes the i-th layer of the backbone. Specifically, for IISAN-Versa, we decouple the trainable parameters into three separate towers: the textual-modality side-adapted network for text representation, obtaining the final text embedding e^{t}; the visual-modality side-adapted network

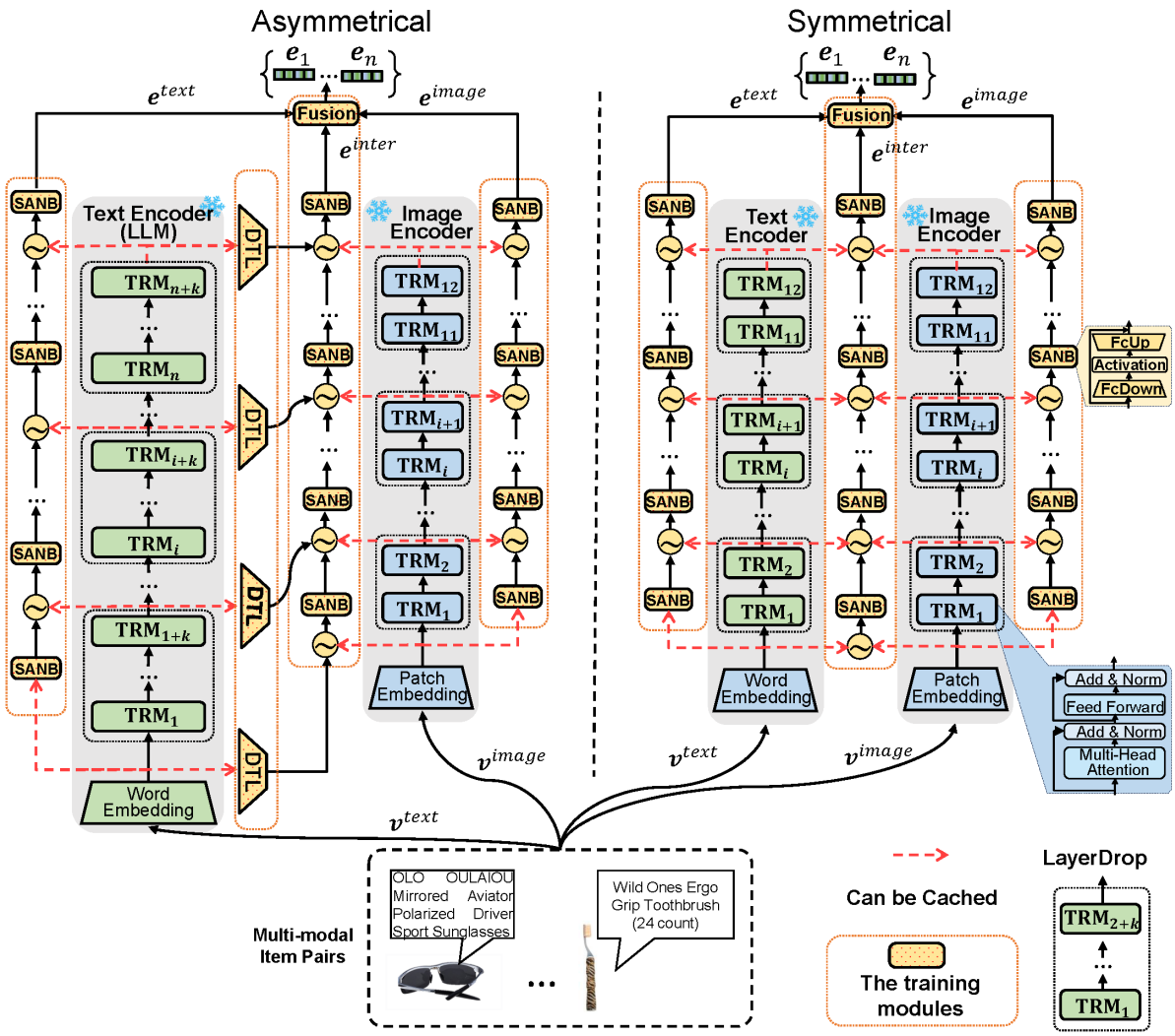


Figure 4.2: An Overview of the IISAN-Versa for Sequential Recommendation. The IISAN-Versa model introduces versatile intra- and inter-modal side-adapted networks (SANs) designed for adapting symmetrical and asymmetrical multimodal foundation models in sequential recommendation. These networks consist of multiple SAN blocks (SANBs) and learnable fusion gates, merging two asymmetrical structures. IISAN-Versa employs dimension transformation layers (DTLs) to align the embedding dimensions. Notably, we leverage LayerDrop to further eliminate redundancy. The asymmetrical framework utilises an even-numbered group LayerDrop technique, which drops K blocks per group. This approach ensures that the large text encoder aligns seamlessly with the image encoder.

for image representation, obtaining the image embedding e^{v}; and an inter-modal side-adapted network for image-text interactive representation, obtaining the inter-modality embedding e^{inter}. A linear-based fusion layer (FL) is added to ensure the consistency of the output dimensions of the item embedding and input dimensions of the sequential encoder following (Y. Hou, Mu, et al., 2022; Yuan et al., 2023) for the final recommendation, as follows:

$$e^{item} = FL([e^{v} : e^{inter} : e^{t}]) \tag{4.1}$$

where [:] means the feature concatenation. Then, we input the $\boldsymbol{e}^{item}$ into the sequential encoders and calculate the final predicted score for user u to i-th item as $\hat{\boldsymbol{y}}_{ui}$, which is the product of the output sequential encoder and corresponding item embedding.

4.3.3 Intra-model Side Adapted Network

Each side-adapted network (SAN) consists of several SAN blocks (SANBs), which include Upsample and Downsample layers built on a fully connected network. Each SANB, a unit within the SAN, serves as a network block. In this research, we adopted the adapter block as the basic unit of the SAN, termed SANB. Unlike the insertion from Adapter Tuning (Houlsby et al., 2019), SANB is implemented in a decoupled manner. For instance, in the textual SAN, a learnable gate mechanism is utilised to fuse the information $h_{i-1}^{B^{intra}}$ from the previous SANB with the current hidden state h_i^t, as demonstrated in the following formula:

$$h_i^{B^{intra}} = SANB^{intra}\left(\mu_i^t * h_{i-1}^{B^{intra}} + (1-\mu_i^t) * h_i^t\right) \tag{4.2}$$

where $\mu_i^t \in [0,1]$. Note that the first SANB will only take the text embeddings and the visual SAN employs similar operations. This Intra-SAN is applicable to both symmetrical and asymmetrical frameworks, the only difference we should make is to enlarge the dimension of input and output dimensions of SANB to fit the larger encoders.

4.3.4 Inter-model Side Adapted Network

When adapting IISAN-Versa to asymmetrical multimodal foundation models, the key difference is in handling the fusion of inter-modality. By performing group layer dropping, we transform the asymmetrical number of hidden states into a symmetrical form, as explained in Section 4.3.5. Following prior work (Fu, Yuan, et al., 2024; Yuan et al., 2023), we adopt a *Dimension Transformation Layer (DTL)* to align the high-dimensional embeddings from the LLM with the lower-dimensional outputs of vision encoders. The DTL is computed as:

$$\mathbf{z} = \mathbf{W}\mathbf{x} + \mathbf{b}, \quad \mathbf{W} \in \mathbf{R}^{d_{\text{out}} \times d_{\text{in}}},\ \mathbf{b} \in \mathbf{R}^{d_{\text{out}}}$$

where $\mathbf{x} \in \mathbf{R}^{d_{\text{in}}}$ is the input feature, $\mathbf{z} \in \mathbf{R}^{d_{\text{out}}}$ is the transformed output, d_{in} and d_{out} denote the input and output dimensions respectively, and $d_{\text{in}} > d_{\text{out}}$.

We then apply a fusion mechanism to merge the hidden states of the two modalities and sum the information from the last SANB using a gating mechanism similar to Intra-SAN, as described by the following formula:

$$h_i^{B^{inter}} = SANB^{inter}\left(\beta_i h_i^v + (1-\beta_i)X_i + h_{i-1}^{B^{inter}}\right) \tag{4.3}$$

where

$$X_i = \begin{cases} h_i^t, & \text{for IISAN-VS} \\ DTL(h_i^t), & \text{for IISAN-VA} \end{cases} \tag{4.4}$$

Algorithm 1 Group LayerDrop for layer selection

```
function SELECTLAYERS(L, L^v)
    step ← ⌊(L−1)/(L^v−1)⌋
    S ← []
    for i = L^v − 1 to 0 do
        append L − step × i to S
    end for
    return S
end function
```

Table 4.1: Selected layers S for various total depths L (with $L^v = 6$).

Number of Layers	Selected layers
12	$\{2,4,6,8,10,12\}$
24	$\{4,8,12,16,20,24\}$
28	$\{3,8,13,18,23,28\}$
32	$\{2,8,14,20,26,32\}$
48	$\{3,12,21,30,39,48\}$
80	$\{5,20,35,50,65,80\}$

where $\beta_i \in [0,1]$. Note that the first inter-SANB only inputs the text embedding and the visual embeddings. We define the symmetrical and asymmetrical IISAN-Versa as IISAN-VS and IISAN-VA.

4.3.5 LayerDrop

Additionally, to further enhance network efficiency and address issues of layer redundancy, we introduce LayerDrop techniques, similar to those in (Fu, Ge, et al., 2024; Sung et al., 2022), to reduce the number of SAN blocks and align asymmetrical MFM. Specifically, for symmetrical MFM, we group two transformer blocks together and drop the first hidden state to the SANs,

which can save half the number of SANBs. In the case of asymmetrical MFM, to address the imbalance between the layers of LLM and vision encoders, we apply a simple yet effective group LayerDrop technique that groups k layers evenly from the hidden states of the LLM. Suppose there are L^{t} and L^{v} layers in the text encoder and image encoder, respectively. In scenarios of asymmetrical adaptation, the text encoder is an LLM, thus $L^{\mathrm{t}} \gg L^{\mathrm{v}}$. For image encoders, we continue using the same LayerDrop technique as with symmetrical structures, where we drop half the number of layers. In this thesis, we design to select the largest k to ensure the maximum range of cross-layer interaction within the entire LLM. The number k can be represented as follows:

$$\max_{k} \left(L^{\mathrm{t}} - k \cdot \frac{L^{\mathrm{v}}}{2} \geq 1 \right) \tag{4.5}$$

We compute

$$\text{step} = \left\lfloor \frac{L-1}{L^{v}-1} \right\rfloor$$

and select layers as shown in Algorithm 1. The specific selections for various L (with $L^{v} = 6$) are presented in Table 4.1. In Section 4.6.5, we further investigate various LayerDrop schemes, showcasing the performance impact of dropping different numbers of blocks in both symmetrical and asymmetrical structures.

4.3.6 Caching Strategies

We adopt a caching technique as a refinement strategy for efficiency (Y. Chen et al., 2020; Karedla et al., 1994). As shown in Figure 4.2, the decoupled PEFT mechanism, consisting of the separable pre-trained backbone model and the new trainable model, enables this approach. This technique involves the storage and reuse of item hidden states extracted from pre-trained multimodal backbones, thereby minimising the need for repeated forward passes of foundational models during training. This method can significantly enhance efficiency for asymmetrical IISAN-Versa. The primary reason is that it fundamentally avoids the necessity of moving the LLM into the GPUs, thus making it feasible to involve the LLM in the adaptation process even with very limited resources. Note that all EPEFT methods are not able to be cached due to the nature of embedding trainable parameters inside the foundation models, which will be further explained in Section 4.4.

4.3.7 Training Objectives

In terms of training details, we exploit the in-batch debiased Cross-Entropy loss function $\mathscr{L}_{CE}$ (Y. Ni et al., 2023; Yuan et al., 2023) widely adopted in both academic literature and industrial systems.

$$D_{ui} = \exp(\hat{y}_{ui} - \log(p_i)) + \sum_{j \in [B], j \notin I_u} \exp(\hat{y}_{uj} - \log(p_j)) \tag{4.6}$$

$$\mathscr{L}_{CE} = -\sum_{u\in\mathscr{U}}\sum_{i\in[2,\ldots,n+1]} \log\frac{\exp(\hat{y}_{ui} - \log(p_i))}{D_{ui}} \tag{4.7}$$

where p is the popularity of the item, I_u and B stand for the item set interacted by user u and the batch. $n+1$ item denotes the predicted item for user u. D_{ui} is an auxiliary variable used to simplify Equation (4.7).

4.4 Efficiency Analysis

During the training of a neural network, the computation involved in backward propagation and the forward pass often represents a significant bottleneck in terms of both training time and GPU memory usage. In this section, we provide an efficiency analysis to understand the differences in the forward and backward propagation processes of full fine-tuning (FFT), Embedded Parameter-Efficient Fine-Tuning (EPEFT) methods such as Adapter and LoRA, and the Decoupled PEFT, IISAN-Versa, in both its Uncached and Cached approach. We argue that traditional embedded PEFT does not substantially improve training efficiency because it does not significantly reduce the computational load of backward propagation. In contrast, the Uncached IISAN-Versa approach can save computation in the backward pass, while the Cached IISAN-Versa can also save time during the forward pass of large foundation models.

To simplify the theoretical analysis, we consider a scenario involving large foundation models comprising n transformer blocks, where the i-th transformer block is denoted by TRM_i. For each transformer block, all PEFT methods, including Adapter, LoRA, and IISAN, will place a trainable PEFT module denoted by $peft$. We assume that $TRM \gg peft$ according to the definition of parameter efficiency.

In the forward pass of full fine-tuning, we denote the final output as Z_n. This process can be represented as follows:

$$Z_n = TRM_n(\cdots TRM_2(TRM_1(x))) \tag{4.8}$$

To update each neural network module, it is necessary to calculate the gradients, starting from the initial block TRM_1. This leads to the longest propagation chain of the loss L with respect to the parameters of the first block:

$$\frac{\partial L}{\partial TRM_1} = \frac{\partial L}{\partial Z_n}\cdot\frac{\partial Z_n}{\partial Z_{n-1}}\cdot\frac{\partial Z_{n-1}}{\partial Z_{n-2}}\cdot\cdots\cdot\frac{\partial Z_2}{\partial Z_1}\cdot\frac{\partial Z_1}{\partial TRM_1} \tag{4.9}$$

For embedded PEFT, the modules are incorporated inside each transformer block. This can be expressed as:

$$p_n = peft_n(TRM_n(\cdots peft_2(TRM_2(peft_1(TRM_1(x)))))) \tag{4.10}$$

We denote the output of the n-th PEFT module as p_i and the output of the n-th transformer block

as Z_n^*:

$$Z_n^* = TRM_n(\cdots peft_2(TRM_2(peft_1(TRM_1(x)))) \tag{4.11}$$

Thus, the overall forward and backward propagation through the entire foundation model is expressed as:

$$\frac{\partial L}{\partial peft_1} = \frac{\partial L}{\partial p_n} \cdot \frac{\partial p_n}{\partial Z_n^*} \cdot \frac{\partial Z_n^*}{\partial p_{n-1}} \cdot \cdots \cdot \frac{\partial Z_2^*}{\partial p_1} \cdot \frac{\partial p_1}{\partial peft_1} \tag{4.12}$$

The additional computational cost introduced by PEFT modules is generally negligible because $peft \ll TRM$. However, in practice, embedded PEFT can still offer efficiency improvements by reducing the gradients stored on the GPU and minimising training time during weight updates, as demonstrated in (Fu, Ge, et al., 2024). Nonetheless, the primary computational challenges associated with both forward and backward propagation in large foundation models remain.

In contrast, for decoupled PEFT approaches like IISAN-Versa, the training process differs significantly. The decoupled framework splits the forward pass into two stages. The first stage involves a forward pass through the backbone model, as shown in Equation 4.8. However, the key difference is that the outputs (hidden states) of each transformer block layer are cached, allowing these outputs to serve as constant inputs to the standalone PEFT modules. The forward pass of decoupled PEFT modules is then defined as:

$$p_n^* = peft_n(\cdots peft_2(peft_1(x))) \tag{4.13}$$

Thus, the backward pass does not need to traverse the large foundation model:

$$\frac{\partial L}{\partial peft_1} = \frac{\partial L}{\partial p_n^*} \cdot \frac{\partial p_n^*}{\partial p_{n-1}^*} \cdot \frac{\partial p_{n-1}^*}{\partial p_{n-2}^*} \cdot \cdots \cdot \frac{\partial p_2^*}{\partial p_1^*} \cdot \frac{\partial p_1^*}{\partial peft_1} \tag{4.14}$$

For IISAN-Versa (Cached), hidden states are stored in on-disk memory rather than recalculated each time during training, further optimising the forward pass and achieving a high level of efficiency. Therefore, we can further give the definition of DPEFT, which is the PEFT method whose forward pass can be written as the form of Equation 4.8 and Equation 4.13 with the backward pass only by Equation 4.14, where the step of Equation 4.8 can be optimised by caching.

In this analysis, we compared Full Fine-Tuning (FFT), Embedded PEFT, and Decoupled DPEFT in terms of their efficiency and resource utilisation when training large foundation models. We found that EPEFT does not significantly boost efficiency because it does not effectively reduce computational demands in both forward and backward propagation. This approach still requires traversing the entire model during training, which limits its potential to save computational resources. In contrast, DPEFT (IISAN-Versa) reduces computational demands by decoupling the fine-tuning process, significantly minimising backward propagation through the foundation models. Moreover, when equipped with a caching strategy, DPEFT can handle for-

ward propagation more efficiently by performing it once and caching the results for subsequent use. This combination makes DPEFT a highly efficient and resource-effective approach for adapting large foundation models.

4.5 Experiment Setup

In this section, we illustrate the used datasets, caption generation, evaluation methods, and implementation details of the IISAN-Versa.

4.5.1 Dataset

In this study, each item in a user sequence is represented by its raw modality information. To assess our methods on items that include both image and text modalities, we utilise the Amazon review datasets (J. Ni et al., 2019). Specifically, we employ three widely used datasets from this collection: “Industrial and Scientific," “Musical Instruments," and “Office Products." These datasets were adopted following the preprocessing steps outlined in prior works (Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024; R. Li et al., 2025; Yuan et al., 2023). Furthermore, to explore the performance of IISAN-Versa compared with other recommendation approaches with a fair comparison, we also incorporate MicroLens (Y. Ni et al., 2023), a public multimodal video recommendation benchmark that includes raw text, images, and video content. Since MicroLens is a dataset rich in modality information, we also explore multimodal text scenarios within this public benchmark. To improve the evaluation scale, we additionally sampled one million interactions from the publicly available NINERec dataset (J. Zhang et al., 2023).

4.5.2 Multimodal Text Generation

To explore the potential of IISAN-Versa in adapting multiple scenarios, such as multimodal text scenarios containing cover text and corresponding video texts, we utilise the pretrained LLaVA model to generate captions for image and video content from the public benchmark MicroLens datasets, as illustrated in Figure 4.3. For this process, we input the cover image and extract three evenly spaced frames from each video. In terms of caption models, we use “llava-hf/llava-v1.6-mistral-7b-hf" for cover image captioning. For video captioning, we apply

Table 4.2: Dataset Description

Dataset	Users	Items	Interactions	Content
Scientific	12,076	20,314	81,711	Text&Image
Instrument	10,000	19,246	75,022	Text&Image
Office	10,000	22,785	71,276	Text&Image
MicroLens	100,000	19,738	719,405	Text&Image&Video
NINERec-1M	151,495	104,605	1,035,822	Text&Image

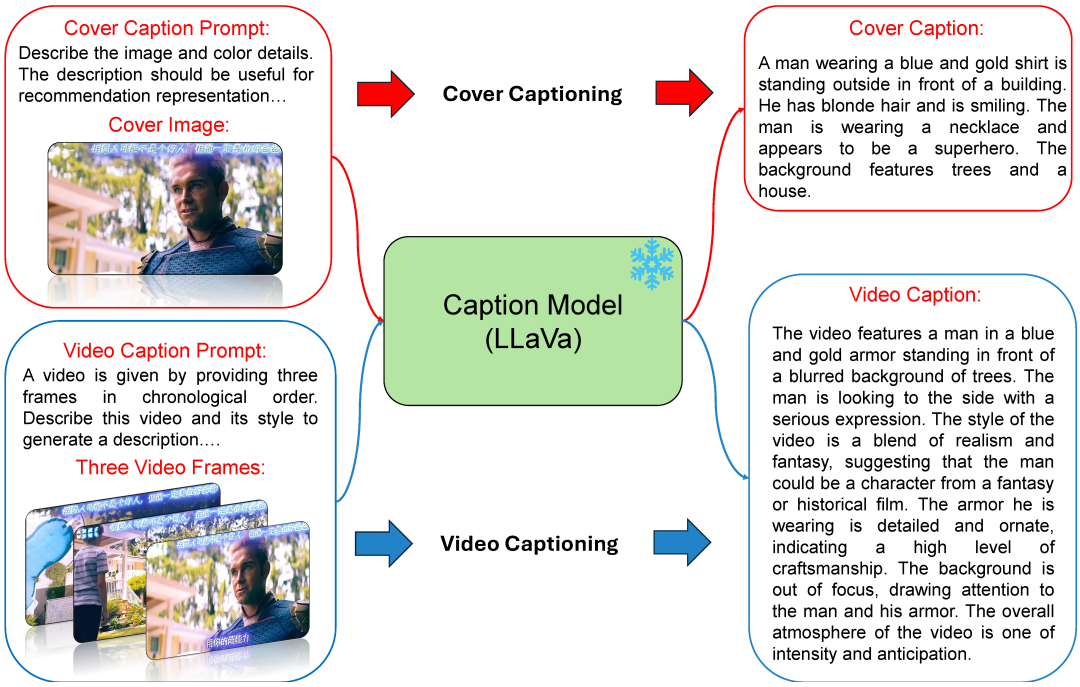


Figure 4.3: An Overview of Caption Generation. The image or video frames, along with the corresponding caption prompt, are input into the pre-trained LLaVA caption model. The model, kept frozen during this process, generates both the cover and the caption simultaneously.

"liuhaotian/llava-v1.6-34b" due to the higher complexity of videos; a larger caption model provides a more accurate representation of the video content. Our video captioning strategies follow the preprocessing methods used in Video generation OpenSora[3]. To enhance LLaVA's ability to produce effective representations for postprocessing, we incorporate in-context learning by including concrete examples in the prompt. We adopt the prompt from the Sora project[4] for video prompts and use the prompt from DALLE2[5] for cover image prompts.

Finally, these multiple generated captions as well as the original texts will be used as the multimodal text input to IISAN-Versa to evaluate multi-scenario generalisation ability.

4.5.3 Performance Evaluations.

Based on previous research (Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024; X. He et al., 2017; Yuan et al., 2023), our methodology employs a leave-one-out evaluation strategy. In this approach, the final item in the interaction sequence is reserved for testing, the penultimate item is used for validation, and all preceding items are utilised for training. We evaluate our model using HR@10 (Hit Ratio) and NDCG@10 (Normalised Discounted Cumulative Gain), consistent with the evaluation methodologies in (Fu, Yuan, et al., 2024; X. He et al., 2017; Yuan et al., 2023), as the key performance metrics. Unless stated otherwise, all results reported refer to the test set.

[3] https://github.com/hpcaitech/Open-Sora
[4] https://openai.com/sora
[5] https://openai.com/index/dall-e-2

Furthermore, it is crucial to mention that the predicted item is evaluated against the complete set of items.

4.5.4 Implementation Details.

The pretrained models from the Huggingface platform[6], detailed in Table 4.3, are employed as text and image encoders in this research. Note that due to the limited computing resources, we adopt the Llama-3-70b with GPTQ 4-bit quantisation. We experiment with hidden representation dimensions for the sequential encoder from {32, 64, 128} and set it to 64. We fixed the sequential encoder to include 2 Transformer blocks and 2 attention heads following (Fu, Ge, et al., 2024; Yuan et al., 2023). Using the Adam optimiser without weight decay, we tested

Table 4.3: Description of the pre-trained models used in this research. L denotes the number of Transformer blocks, H represents the hidden dimension of the transformer layer, B indicates the number of parameters in billions, and M indicates the number of parameters in millions.

Modality	Pre-trained model	Architecture	#Param
Text	BERT-tiny	L=2, H=128	4.4M
	BERT-base	L=12, H=768	110M
	BERT-large	L=24, H=1024	340M
	Gemma-7b	L=28, H=3072	7B
	Gemma-7b-it	L=28, H=3072	7B
	Mistral-7b	L=32, H=4096	7B
	Llama2-7b	L=32, H=4096	7B
	Llama2-7b-chat	L=32, H=4096	7B
	Llama3-8b	L=32, H=4096	8B
	Llama3-8b-it	L=32, H=4096	8B
	Mixtral-8x7b-it	L=32, H=4096	47B
	Mixtral-8x7b	L=32, H=4096	47B
	Llama3-70b-GPTQ-it	L=80, H=8192	70B
	Llama3-70b-GPTQ	L=80, H=8192	70B
Image	ViT-tiny	L=12, H=192	21M
	ViT-base	L=12, H=768	86M
	ViT-large	L=24, H=1024	307M
	ViT-huge	L=32, H=1280	632M
	EVA-ViT	L=48, H=5120	11B

learning rates from 1e-5 to 1e-3 while maintaining a dropout probability of 0.1. We explored batch sizes from {8, 16, 32, 64, 512}, selecting the largest viable option to optimise GPU memory, and searched for SANB hidden dimensions from {8, 16, 32, 64, 128, 512, 1024, 2048}. Hyperparameters were searched separately for the MicroLens and Amazon datasets. All hyperparameters were finalised based on validation performance, and results were reported using the test set. The experiments were conducted on one A6000 GPU.

4.6 Experiment

In this section, we explore IISAN-Versa by addressing the following research questions:

[6]https://huggingface.co/

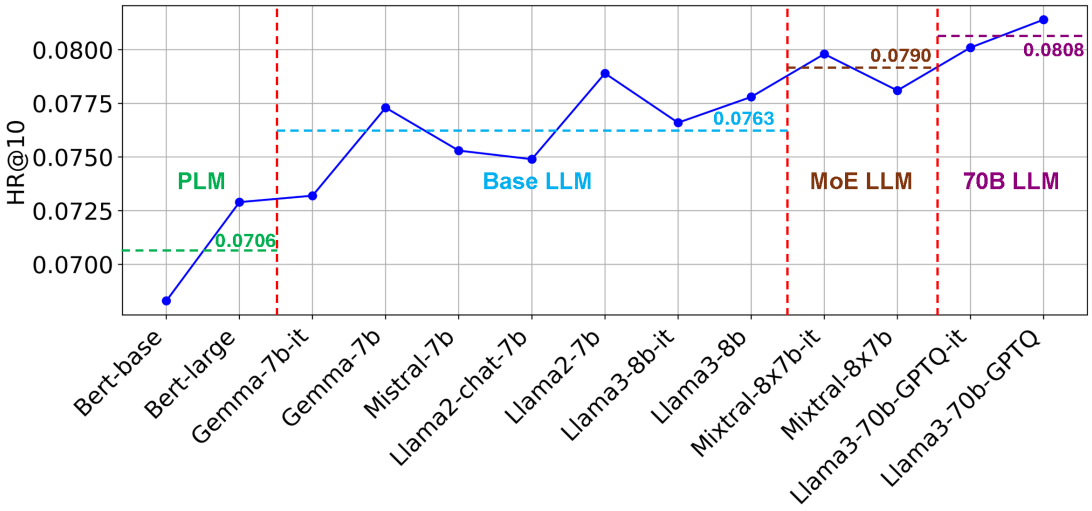


Figure 4.4: Scaling effect of IISAN-Versa's text encoder on Scientific dataset with Hit@10 metric. We group the pre-trained text encoders based on their scale, with each group increasing in size from left to right. We adopt the vertical red dotted line to separate each group. The horizontal line represents the average performance of each group.

- **RQ4.1:** Does Asymmetrical IISAN-Versa's performance improve when equipped with a larger text encoder compared to its standard symmetrical IISAN-Versa?
- **RQ4.2:** If the answer to RQ1 is "Yes", does IISAN-Versa exhibit a scaling effect with the integration of progressively larger LLMs?
- **RQ4.3:** How does IISAN-Versa compare in performance against established public benchmarks?
- **RQ4.4:** How is the efficiency of IISAN-Versa compared with popular EPEFT approaches?
- **RQ4.5:** What are the key findings about the components affecting the performance of IISAN-Versa?
- **RQ4.6:** Would IISAN-Versa show advantages in processing other multimodal scenarios such as multimodal text using both raw text (title) and the text generated (cover and video captions) from images or video?
- **RQ4.7:** How does IISAN-Versa perform under different hyperparameter settings?

4.6.1 IISAN-Versa Performance with larger text encoders (RQ4.1)

To validate RQ1, which is whether incorporating a larger text encoder can enhance the performance of IISAN-Versa, this section evaluates the performance of an asymmetrical IISAN-Versa (IISAN-VA) using the BERT-large as text encoder compared to a symmetrical IISAN-Versa (IISAN-VS) with a BERT-base encoder. The specifics of the models are detailed in Table 4.3. In

Table 4.4: Performance (%) comparison across methods on three datasets. Best results in bold, second-best underlined. "*" denotes significant improvement over the second-best (IISAN-VS) at 0.05 level with paired T-test.

Method	Scientific		Instrument		Office	
	H@10	N@10	H@10	N@10	H@10	N@10
FFT	6.62	4.06	8.76	6.76	6.30	4.58
Adapter	6.61	3.91	8.82	6.83	6.65	4.85
LoRA	6.62	4.09	8.43	6.64	6.55	4.78
BitFit	6.37	3.76	8.65	6.71	6.78	4.87
IISAN-VS	<u>6.83</u>	<u>4.14</u>	<u>9.06</u>	<u>7.01</u>	<u>6.80</u>	<u>4.92</u>
IISAN-VA	**7.29***	**4.41***	**9.52***	**7.17***	**7.02***	**5.11***

this section, we fix the image encoder as a ViT-base for a fair comparison, more image encoders will be further discussed in Section 4.6.2. We also use FFT and popular EPEFT approaches (Adapter, LoRA, and BitFit) as baselines. FFT and EPEFT methods were applied only to the symmetrical model due to the high computational cost of EPEFT and FFT. From Table 4.4, IISAN-VS demonstrates superior performance over traditional full fine-tuning and embedded PEFT approaches (Adapter, LoRA, and BitFit), as evidenced by its effectiveness across the three Amazon datasets.

The asymmetrical IISAN-Versa (IISAN-VA) further enhances performance with the larger text encoder. It achieves a gain of +6.31% in HR@10 and +6.12% in NDCG@10 for the Scientific dataset, +4.83% and +2.23% for the Instrument dataset, and +3.13% and +3.72% for the Office dataset. These improvements underscore IISAN-VA's ability to effectively adapt to larger text encoders, leading to increased recommendation performance. This also demonstrates that compared to a BERT-base model, asymmetrical IISAN-Versa with a larger text encoder can generate a more powerful text representation for sequential recommendation tasks.

(Answer to RQ4.1) IISAN-Versa can improve performance when equipped with a larger text encoder. Furthermore, we conclude the overall findings as the following: (1) IISAN-VS demonstrates its effectiveness compared to traditional fine-tuning approaches, including FFT and EPEFT, as shown by its superior performance. (2) The IISAN-VA further enhances the performance of IISAN-VS, affirming the advantages of using a larger and more powerful text encoder over symmetrical IISAN-VS for generating powerful text representations in sequential recommendation tasks. Such a finding further highlights the IISAN-Versa's versatility in adapting both symmetrical and asymmetrical structures and their effectiveness.

4.6.2 Scaling Effect (RQ4.2)

In this section, we explore whether the IISAN-Versa exhibits a scaling effect. We perform experiments on the Scientific dataset, scaling up the text encoder across 14 models ranging from the basic BERT-base to the gigantic Llama-3-70b model. Since the available ViT models

Table 4.5: IISAN-Versa Encoders Scaling.

Model structure	HR@10	NDCG@10
BERT-base + ViT-base	0.0683	0.0414
BERT-large + ViT-base	0.0729	0.0441
Gemma-7b-it + ViT-base	0.0732	0.0446
Gemma-7b + ViT-base	0.0773	0.0475
Mistral-7b + ViT-base	0.0753	0.0465
Llama2-7b + ViT-base	0.0789	0.0481
Llama2-chat-7b + ViT-base	0.0749	0.0453
Llama3-8b + ViT-base	0.0778	0.0463
Llama3-8b-it + ViT-base	0.0766	0.0474
Mixtral-8×7b-it + ViT-base	0.0798	0.0480
Mixtral-8×7b + ViT-base	0.0781	0.0473
Llama3-70B-GPTQ-it + ViT-base	0.0801	0.0478
Llama3-70B-GPTQ + ViT-base	**0.0814**	**0.0494**
BERT-base + ViT-large	0.0688	0.0425
BERT-base + ViT-huge	0.0682	0.0415
BERT-base + EVA-ViT	0.0746	0.0457
Llama3-70B-GPTQ + ViT-large	0.0756	0.0455
Llama3-70B-GPTQ + ViT-huge	0.0776	0.0464
Llama3-70B-GPTQ + EVA-ViT	0.0762	0.0469
Llama3-70B-GPTQ + ViT-tiny	0.0780	0.0478
BERT-tiny + EVA-ViT	0.0744	0.0449

Table 4.6: Performance comparison of IISAN-Versa on a large dataset over 1M interactions. "*" denotes that the improvements are significant at the level of 0.05 with paired T-test.

Class	Model	HR@10	NDCG@10
MM VideoRec	Full finetuning	0.0098	0.0048
	Adapter	0.0097	0.0048
IISAN-Versa	IISAN-VS (ours)	<u>0.0105</u>	<u>0.0052</u>
	IISAN-VA (ours)	**0.0111***	**0.0054***

are ViT-base, ViT-large, and ViT-huge, we also performed scaling experiments on these image encoders. To better visualise the recommendation performance alongside the scale of model size, we categorise the models into four groups: PLM for the BERT model, Base LLM for models with 7 to 8 billion parameters, MoE LLM for the 8x7b Mixtral LLM, and 70b LLM.

As depicted in Table 4.5, IISAN-Versa equipped with the largest scale of Llama3-70b-GPTQ model demonstrated the most superior performance. The trend of these results is visualised in Figure 4.4, which shows the overall trajectory is generally upward. Moreover, the average performance within each group significantly improves with each scale increase, supporting the hypothesis that IISAN-Versa follows a scaling effect for text encoders. This observation validates the IISAN-Versa framework's capability to leverage advancements from pre-trained foundation models in the NLP domain. We can observe that, with the exception of the Mixtral 8x7b model, most LLMs with instruction tuning typically do not outperform those without it. This reflects

that the LLM without instruction tuning may generate a more powerful representation.

Table 4.7: Performance comparison of IISAN-Versa on a multimodal public benchmark MicroLens. In this experiment, IISAN-VS and IISAN-VA adopt BERT-base and BERT-large respectively. "*" denotes that the improvements of the best models compared with previous state-of-the-art GRU4Rec$_v$ are significant at the level of 0.05 with paired T-test.

Class	Model	HR@10	NDCG@10
IDRec (CF)	DSSM	0.0394	0.0193
	LightGCN	0.0372	0.0177
	NFM	0.0313	0.0159
	DeepFM	0.0350	0.0170
IDRec (SR)	NextItNet	0.0805	0.0442
	GRU4Rec	0.0782	0.0423
	SASRec	0.0909	0.0517
VIDRec (Frozen)	$\text{YouTube}_{\text{ID}}$	0.0461	0.0229
	$\text{YouTube}_{\text{ID+V}}$	0.0392	0.0188
	MMGCN_{ID}	0.0141	0.0065
	$\text{MMGCN}_{\text{ID+V}}$	0.0214	0.0103
	GRCN_{ID}	0.0282	0.0131
	$\text{GRCN}_{\text{ID+V}}$	0.0306	0.0144
	$\text{DSSM}_{\text{ID+V}}$	0.0279	0.0137
	$\text{SASRec}_{\text{ID+V}}$	0.0799	0.0415
VideoRec	$\text{NexItNet}_{\text{v}}$	0.0862	0.0466
	$\text{GRU4Rec}_{\text{v}}$	0.0954	0.0517
	SASRec_{v}	0.0948	0.0515
MM VideoRec	Full finetuning	0.0926	0.0493
IISAN-Versa	IISAN-VS (ours)	<u>0.0960</u>	<u>0.0530</u>
	IISAN-VA (ours)	**0.0964***	**0.0533***

As shown in Table 4.5, replacing ViT-base with ViT-large or ViT-huge yields only minor improvements, and in some cases even reduces performance—especially when using strong text encoders like Llama3-70B-GPTQ. To explore further, we include EVA-CLIP-18B which contains one of the largest publicly available vision transformers. While it significantly boosts performance when paired with lightweight text models, its impact is limited when the text encoder is already strong. This suggests that once strong textual representations are in place, the value of visual features diminishes—a trend aligned with prior findings (Fu, Yuan, et al., 2024; Yuan et al., 2023) and supported by our ablation studies showing the dominant role of text in multimodal recommendation. To further strengthen our analysis, we include extreme architectural asymmetry—where the text encoder is significantly larger or smaller than the vision encoder—demonstrating the robustness of our IISAN-Versa.

Beyond model-wise scaling, we evaluate IISAN-Versa on the large-scale NINERec dataset (over 1M interactions) as shown in Table 4.6. IISAN-VS and IISAN-VA achieve competitive performance over FFT and Adapter, highlighting the practical effectiveness of our approach.

(Answer to RQ4.2) IISAN-Versa complies with the scaling effect of text encoders, confirming its potential to inherit the benefits of advanced pre-trained text transformer models. This also indicates that researchers can explore LLMs not only as powerful text generators

but also as effective text encoders in future works, thus reaching more comprehensive representations for recommender systems. Further analysis shows that a large vision encoder, such as EVA-ViT, can effectively compensate when the text encoder is relatively weak, leading to noticeable performance gains. However, its impact diminishes once the text encoder is already strong, suggesting that visual enhancements offer limited benefit in such cases. Therefore, scaling the vision encoder is most beneficial in asymmetric setups with lightweight text models, but less effective when paired with high-capacity text encoders.

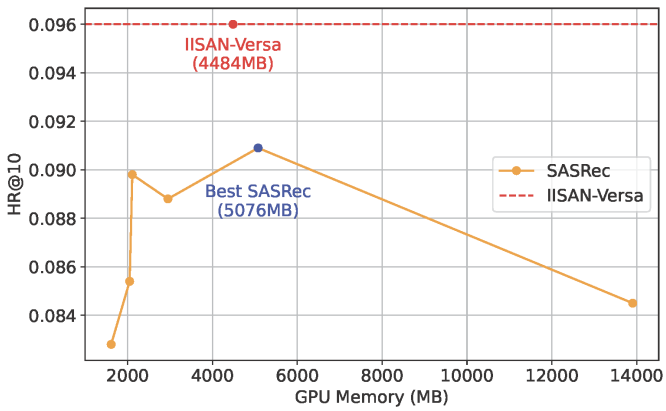


Figure 4.5: Comparison of GPU Memory Usage and Performance Between SASRec and IISAN-Versa.

Table 4.8: Efficiency comparison with optimal SASRec and IISAN-Versa. GM represents GPU memory (GB), TT represents Training time (seconds) per epoch.

Model	GM	TT
IISAN-Versa	4.38	51
SASRec	4.96	9

4.6.3 Performance on Public Benchmark (RQ4.3)

To further evaluate IISAN-Versa, we compare it with the public benchmark MicroLens, as detailed in Table 4.7. This public benchmark (Y. Ni et al., 2023) includes four main types of baselines: sequential recommendation (IDRec (SR)), collaborative filtering (IDRec (CF)), and ID-based recommendation with features generated from frozen pre-trained video encoder (VIDRec (Frozen Encoder)) or end-to-end (E2E) learning encoders (VideoRec (E2E learning)). The previous state-of-the-art approach, GRU4Rec$_V$ (Y. Ni et al., 2023), utilises GRU4Rec as the sequential encoder with a tunable video encoder VideoMAE to encode video content in an E2E manner directly. To ensure a fair comparison, we directly use the results reported in the original benchmark (Y. Ni et al., 2023). We also perform a comprehensive comparison by including a multimodal full fine-tuning of the multimodal VideoRec framework, as applied in (Fu, Ge, et al., 2024).

Figure 4.5 and Table 4.8 present a comparison between our proposed model and the widely adopted ID-based baseline SASRec, focusing on both effectiveness and computational efficiency. Although SASRec demonstrates strong performance in sequential recommendation, our model consistently achieves higher accuracy across all metrics while requiring less GPU memory. While multimodal processing inherently introduces additional training time, the com-

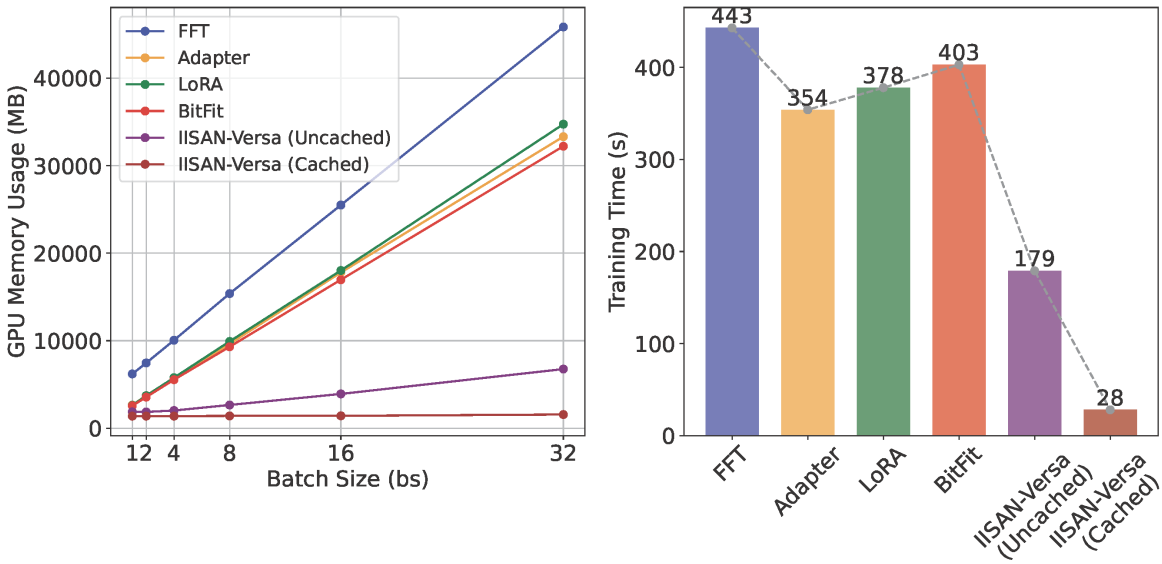


Figure 4.6: GPU Memory and Training time comparison. The left figure shows GPU memory usage under different batch sizes, while the right figure shows training time per epoch.

bination of reduced memory consumption and improved accuracy underscores the practical potential of our approach. These results affirm the real-world viability of our method, particularly in deployment scenarios where both performance and GPU memory are essential.

As shown in Table 4.7, both IISAN-Versa approaches outperform existing approaches on this dataset. Notably, IISAN-Versa can significantly reduce training costs compared to VideoRec, which requires over four times more GPU resources than even full fine-tuning of the multimodal paradigm. This is due to the high computational cost of video encoders that process multiple frames during training. Our approach, with its caching strategy, results in IISAN-Versa using vastly smaller GPU memory. To further validate IISAN-Versa on MicroLens, we also incorporate the standard full fine-tuning method, which underperforms the IISAN-Versa. Within the IISAN-Versa paradigm, IISAN-VA demonstrates better performance compared to IISAN-VS, which is consistent with our previous findings.

(Answer to RQ4.3) IISAN-Versa surpasses state-of-the-art performance on the public benchmark MicroLens dataset.

4.6.4 Efficiency Analysis (RQ4.4)

In Section 4.4, we provided a comprehensive analysis of various adaptation techniques. This section presents a detailed evaluation of experimental efficiency, focusing on two key aspects: GPU memory usage and training speed across full fine-tuning, EPEFT, and DPEFT methods. We provide details on the time taken per epoch and the GPU memory consumption for each method. Additionally, we report the on-disk memory usage for the cached hidden states of each pretrained model.

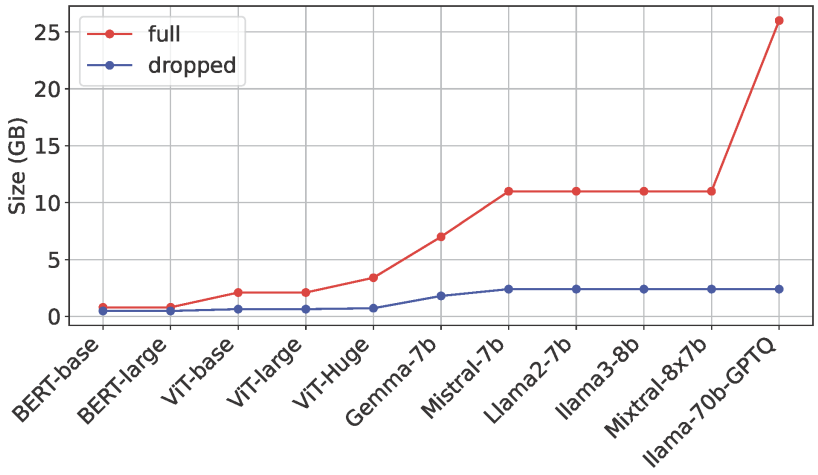


Figure 4.7: On-Disk Storage Requirements for Caching Strategy, by Model Size. The x-axis represents the increasing model size, with the Scientific dataset, comprising 20,314 embedding files to be cached.

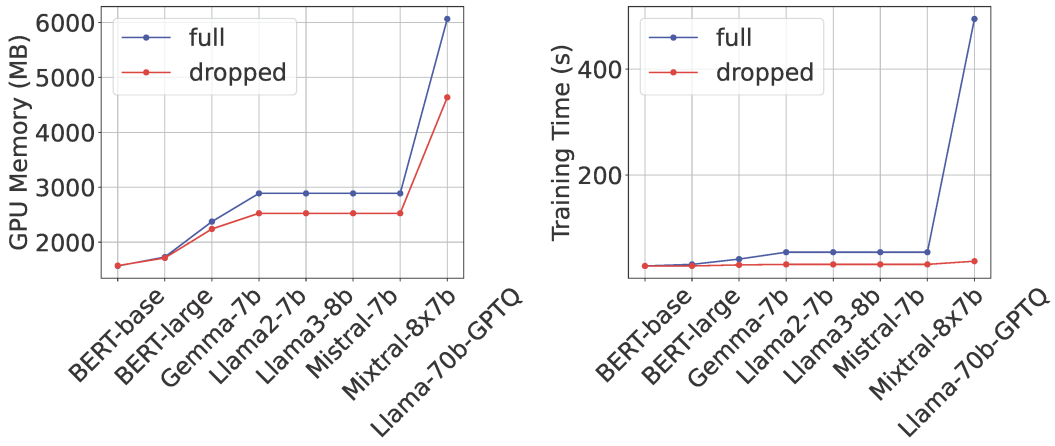


Figure 4.8: GPU memory and Training time usage for IISAN-Versa with different text encoders. 'Full' represents using the entire cached embedding, whereas 'dropped' indicates retaining only the useful cached embeddings.

Figure 4.6 illustrates the variation in GPU memory usage as the batch size increases from {1, 2, 4, 8, 16, 32}. We utilise the BERT-base and ViT-base as the fixed text and vision encoders, respectively. Further studies on the efficiency of various encoder sizes will be discussed in subsequent sections. The results show that IISAN-Versa significantly reduces GPU memory usage as the batch size grows, indicating that the decoupled PEFT approach employed by IISAN-Versa effectively minimises activations during training. Since activations are directly proportional to batch size, this reduction is particularly noteworthy. The EPEFT methods experience a substantial increase as the batch size scales up, making their practical memory savings less impactful compared to the IISAN-Versa. Additionally, IISAN-Versa also exhibits a significant improvement in training speed.

With the caching strategy, we essentially trade on-disk memory (cached for hidden states) for enhanced training efficiency in terms of both time and GPU memory. To demonstrate practical storage requirements, we employ the Scientific dataset as an example and exclude the model with instruction tuning, given that its hidden state shapes are identical to those of the model without instruction tuning. As shown in Figure 4.7, the on-disk storage requirement increases with the model size. The maximum storage required for IISAN-Versa (Cached) is for the Llama-70b-GPTQ model, which needs 26 GB of memory for the entire Scientific dataset, which is not excessively large for a cluster machine. If the cluster does not have sufficient on-disk memory, the storage can be reduced by the number of layers of the model. For example, after applying our k-group layerdrop strategies mentioned in Section 4.3.5, the Llama-70b-GPTQ model requires only six layers and 2.4 GB of storage.[7]

Furthermore, the efficiency of IISAN-Versa (Cached) with different sizes of text encoder is demonstrated in Figure 4.8. The GPU memory usage increases with the model size, but it can be slightly reduced by reducing the hidden states of unnecessary layers. Furthermore, it is worth noting that Llama-70b-GPTQ significantly increases the training cost when we directly use its full embeddings. Given their substantial size, we have observed that loading and processing these models on the CPU becomes the primary bottleneck rather than the GPU computation. Consequently, by reducing the embeddings to only six essential layers, we have achieved a significant reduction in training time.

(Answer to RQ4.4) IISAN-Versa demonstrates a marked efficiency improvement over EPEFT approaches in both training time and GPU memory. Furthermore, while the caching strategy incurs additional on-disk memory costs, these are generally manageable in most cases. Moreover, these costs can be further reduced by selectively caching only the useful layers after layer drop. Utilising the useful layers can also significantly enhance efficiency when dealing with large language model (LLM) encoders.

[7]The storage of hidden states from Llama-70b-GPTQ is identical to that of Mistral, Llama2, etc., because it uses float16 as the datatype, whereas other base LLMs use float32. Additionally, the base LLMs have half the dimension size of the 70B LLM and the same number of layers after removing the unnecessary layers. Consequently, their storage requirements are the same.

Table 4.9: Ablation study for IISAN-Versa and IISAN-VA on Scientific Dataset. IISAN-Versa mainly contains four key components: LayerDrop, Modality Gate, and Intra- and Inter-modal towers. “Frozen Backbone" refers to keeping the modality backbone frozen, without applying any adaptation methods.

Method	IISAN-VS		IISAN-VA	
	HR@10	NDCG@10	HR@10	NDCG@10
-w/o LayerDrop	6.73	4.04	6.85	4.11
-w/o Modality Gate	6.58	3.89	7.08	4.30
Frozen Backbone	6.00	3.53	5.81	3.37
-w/o Inter-modality	6.38	3.89	6.96	4.21
-w/o Intra-modality	6.41	3.83	7.12	4.43
IISAN-Versa	6.83	4.14	7.29	4.41

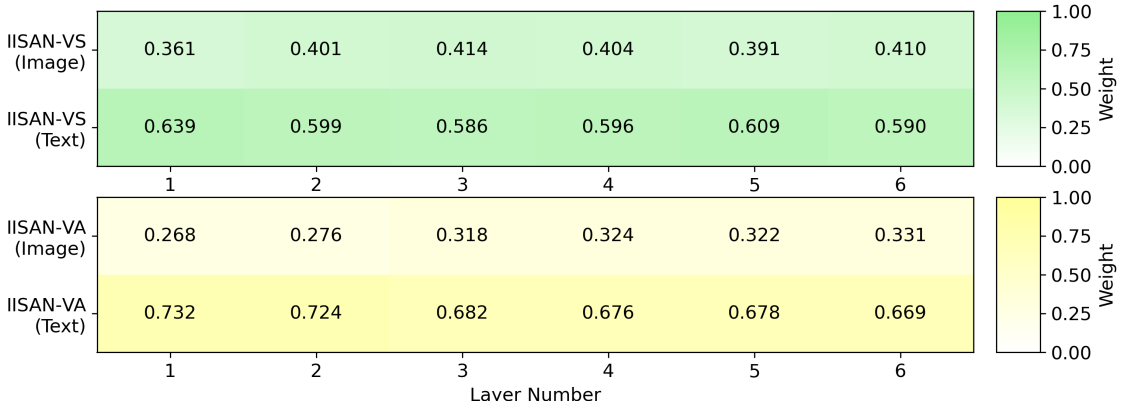


Figure 4.9: Visualisation of inter-modality gates in IISAN-Versa. Darker colours indicate higher gate weights.

Table 4.10: LayerDrop in symmetrical IISAN-Versa on Scientific Dataset. The number of blocks in the Method column represents that keep this number of blocks and drop the others.

Method	Number of blocks	HR@10	NDCG@10
IISAN-VS	2 blocks	6.46	3.88
	3 blocks	6.79	4.12
	4 blocks	6.57	3.98
	6 blocks	**6.83**	**4.14**
	12 blocks	6.73	4.04

Table 4.11: Ablation Study for blocks of asymmetrical IISAN-Versa. n(uniform_all) refers to selecting n blocks evenly distributed across all blocks. "top" and "bottom" indicate that the blocks are drawn from either the upper half or the lower half, respectively.

Method	Configuration	HR@10	NDCG@10
24 + 12	Full	6.85	4.11
12+12	12(uniform_all)+12(all)	6.96	4.29
	12(top)+12(all)	6.96	4.27
	12(down)+12(all)	6.82	4.21
	Mean (12+12)	6.91	4.26
12+6	12(uniform_all)+6(uniform_all)	7.08	4.44
	12(uniform_all)+6(top)	6.80	4.19
	12(uniform_all)+6(down)	7.15	4.29
	12(top)+6(top)	6.94	4.22
	12(top)+6(uniform_all)	7.25	4.37
	12(top)+6(down)	6.67	4.04
	12(down)+6(top)	7.14	4.30
	12(down)+6(uniform_all)	6.84	4.24
	12(down)+6(down)	7.26	4.43
	Mean (12+6)	7.01	4.28
6+6	6(uniform_all)+6(uniform_all)	7.29	4.41
	6(uniform_all)+6(down)	7.21	4.37
	6(uniform_all)+6(top)	6.80	4.19
	6(uniform_top12)+6(uniform_all)	6.96	4.24
	6(uniform_top12)+6(top)	6.95	4.27
	6(uniform_top12)+6(down)	6.97	4.25
	6(uniform_down12)+6(uniform_all)	7.29	4.44
	6(uniform_down12)+6(top)	6.87	4.26
	6(uniform_down12)+6(down)	7.35	4.51
	Mean (6+6)	**7.08**	**4.33**

4.6.5 Ablation Studies (RQ4.5)

In this section, we first demonstrate the importance of the module's layer dropping for both symmetrical and asymmetrical IISAN-Versa. The BERT-large is applied as the text encoder for IISAN-VA's ablation study. We extensively explore the blocks required for asymmetrical IISAN-Versa. Specifically, we experiment with the combinations of blocks, ranging from {24+12, 12+12, 12+6, 6+6}, with the former value in each combination indicating the blocks for 24-layer BERT-large and the latter representing the blocks for 12-layer ViT-base.

(1) Modality Selection. Table 4.9 highlights that using separate intra-modal SAN (Line 4) or inter-modal SAN (Line 5) clearly reduces recommendation effectiveness. However, compared

Table 4.12: Performance on Multimodal Text.

Category	Method	HR@10	NDCG@10
FFT	Title	0.0916	0.0490
	Video Caption	0.0878	0.0470
	Cover Caption	0.0915	0.0493
Same input	Title x2	0.0917	0.0507
	Video Caption x2	0.0866	0.0476
	Cover Caption x2	0.0892	0.0491
Multimodal Text	Title-Video Caption	<u>0.0951</u>	<u>0.0517</u>
	Title-Cover Caption	0.0934	0.0511
	Video-Cover Caption	0.0899	0.0497
Multimodal	IISAN-Versa(Text-Image)	**0.0960**	**0.0530**

to the pre-trained backbones (Line 3) with frozen layers (where only the sequential encoder is trained), both intra-modal SAN and inter-modal SAN can significantly improve performance, performing best when used simultaneously. This reflects that intra- and inter-modal SAN can enhance feature adaptation within each modality and improve inter-modal feature interaction, this argument holds for both IISAN-VS and IISAN-VA.

(2) Gated Fusion. To deepen our understanding of the differences in information usage between textual and visual modalities, we analysed the gate weights at the optimal checkpoint for both symmetrical and asymmetrical IISAN-Versa. We visualise the gate values in Figure 4.9. Since text gate values are consistently above 0.5, all models primarily rely on text. Visual gate values stay between 0.26 and 0.42, with IISAN-VA depending on text even more than IISAN-VS. This trend suggests a predominant reliance on textual modality within our multimodal approach.

(3) LayerDrop Strategies for IISAN-Versa. IISAN-VS's layers exhibit a degree of redundancy, allowing LayerDrop to enhance performance effectively. First, when we remove LayerDrop, both HR@10 and NDCG@10 decrease by 0.1%, demonstrating LayerDrop's effectiveness. Second, we study the effect of different LayerDrop strategies, as described in Table 4.10. We adopt the even-numbered $(2, 4, 6, \ldots, 12)$ transformer blocks (6 blocks), skipping the odd-numbered layers, which achieves the best performance-efficiency balance. In terms of IISAN-VA, the performance drop is even more significant when LayerDrop is completely removed, indicating that the redundancy is more prominent than in IISAN-VS. To explore how many blocks should be used for IISAN-VA, we perform extensive experiments on the combination of the layers and compare their mean values (see Table 4.11). Due to the complexity of the combination, we study the even selection for layer group of {top, down, all} for the 6+6 combination. When the numbers are asymmetrical such as "24+12" and "12+6", we adopt the top hidden states to align with the vision tower for Inter-SAN. The intra-modality stays the same.

Extensive experiments show that the "6+6" structure has the best overall mean performance among all structures. This further supports the point that IISAN-VA has more layer redundancy compared to IISAN-VS, due to its greater number of incorporated layers.[8]

(Answer to RQ4.5) In this section, we conclude with three key findings regarding IISAN-Versa's components: (1) Optimal performance is achieved through the integration of both intra- and inter-modal SANs for both symmetrical and asymmetrical structures. (2) The best performance is achieved by dropping half of the SANBs in the intra- and inter-modal SANs for IISAN-VS. For IISAN-VA, the best performance comes from using the least number of blocks, transforming the number of layers to be symmetrical to the vision encoders.

(3) Text-image interaction is effective, but the text modality plays a more crucial role in the recommendation, where inter-modal SAN can effectively maintain the dominance of text modality while integrating image information.

4.6.6 Performance on Multimodal Text (RQ4.6)

Multimodal text encompasses different types of text, such as captions extracted from raw images and videos. These settings are useful for multimodal recommender systems that lack the computational resources to process raw images or videos (L. Zhang, Zhou, & Shen, 2023), or for datasets in which these modalities are unavailable. Therefore, we aim to explore how IISAN-Versa performs in this scenario. We use the MicroLens dataset (Y. Ni et al., 2023) because its rich modality information allows us to extract captions from both cover images and video content. In the multimodal text experiments, we use IISAN-Versa to encode two combinations of text: raw text (title) and generated caption text (video/cover caption), with each tower encoding a specific text modality. To isolate the impact of the modality itself and minimise the influence of the text encoder's power, we use an IISAN-Versa model with two BERT-base models to evaluate its performance.

The results, shown in Table 4.12, demonstrate that IISAN-Versa effectively adapts to multimodal text by outperforming both title-based and caption-based approaches. This superior performance over these three baselines highlights IISAN-Versa's versatility. Additionally, the combination of raw text—title and video caption—proves to be the most informative. By learning from this combination of these texts, we achieve the best overall performance in multimodal text scenarios. However, it is important to note that no combination of multimodal text outperforms the use of titles with raw cover images[9], indicating that raw modality remains crucial in the multimodal representation of sequential recommendations.

(Answer to RQ4.6) (1) IISAN-Versa is a viable option for multimodal text and effec-

[8]Exhaustively testing all layer combinations, even for smaller models like IISAN-VA (BERT-large + ViT-base), would require about 68 billion evaluations. Instead, we aim to find a general strategy for larger models. We adopt the 6(uniform_all) + 6(uniform_all) method as the default for IISAN-VA, which, while not the best, performs above average among 6+6 combinations in Table 4.11, offering convenience and generality for asymmetrical structures.

[9]Note that using raw video and audio is beyond the scope.

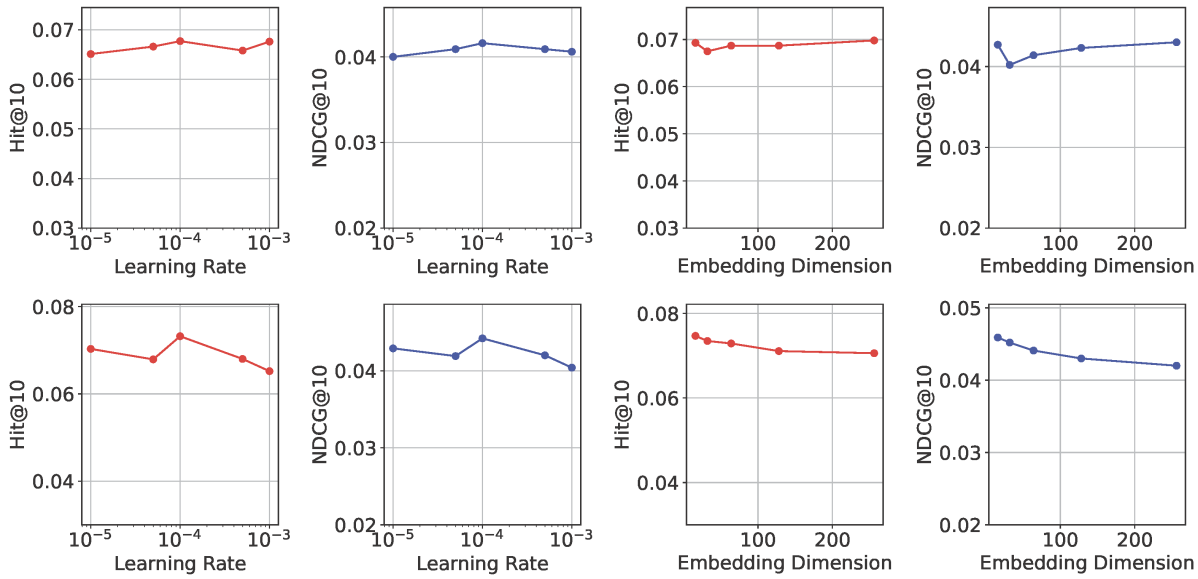


Figure 4.10: Learning rate and Embedding Dimension of IISAN-Versa. The first row represents the performance of IISAN-VS on the Scientific dataset whereas the second row is for IISAN-VA.

tively incorporates multiple types of text information. (2) According to the experiment, titles and video captions are the most important sources for IISAN-Versa. (3) Despite the effectiveness of incorporating multimodal text compared to using only one type of text, we find that IISAN-Versa with raw multimodality achieves the best performance.

4.6.7 Hyper-parameter Sensitivity Analysis (RQ4.7)

In this section, we address RQ7 by conducting a sensitivity analysis on two key hyperparameters, learning rate and embedding dimension, following the setup in (Fu, Yuan, et al., 2024), to provide a more comprehensive empirical evaluation of PEFT-based modality recommendation.

As shown in Figure 4.10, both IISAN-Versa variants demonstrate robust performance across a range of hyperparameter settings. IISAN-VS maintains stable performance regardless of the learning rate or embedding dimension. IISAN-VA also remains stable overall, but its performance slightly degrades when either the embedding dimension or the learning rate is set too high. The best performance is achieved when both hyperparameters are set to relatively small values; notably, smaller embedding dimensions can enhance efficiency.

(Answer to RQ4.7) Both IISAN-VS and IISAN-VA exhibit robustness to hyperparameter variations. However, IISAN-VA requires more conservative tuning, with smaller embedding dimensions and moderate learning rates yielding improved performance and efficiency.

4.7 Conclusion and Future Work

This thesis introduces the new IISAN-Versa, an extension of the IISAN framework, which offers a versatile and efficient architecture for multimodal sequential recommendation tasks. IISAN-Versa overcomes the previous limitation of being applicable only to symmetrical MFMs, demonstrating a clear scaling effect where larger text models consistently enhance recommendation performance. Additionally, IISAN-Versa proves effective in handling diverse multimodal text scenarios, significantly broadening its applicability.

Future research should aim to overcome current limitations by incorporating additional modalities such as audio, raw video, or graph information. Furthermore, exploring the potential of adapting the IISAN-Versa framework to tasks such as Visual Question Answering (VQA) and video retrieval is a promising direction, as these tasks commonly adopt dual-tower architectures where one tower encodes the visual modality and the other the textual input. IISAN-Versa naturally fits into this paradigm, with its modality-specific intra-SANs and Inter-SAN facilitating effective interaction between different modalities. Extending to additional modalities like audio or video can be easily achieved by adding more intra-SANs for each modality and including them in the Inter-SAN computation, enabling flexible multimodal fusion while maintaining adaptation efficiency. Another promising research direction involves optimising training efficiency by reducing the number of trainable parameters, particularly in large text encoders like Llama-70B, given their extensive embedding dimensions.

Chapter 5

CROSSAN: Cross-Modal Adaptation of Multiple Foundation Models

Publication Note

This chapter is adapted from: Junchen Fu, Yongxin Ni, Joemon M. Jose, Ioannis Arapakis, Kaiwen Zheng, Youhua Li, and Xuri Ge. CROSSAN: Towards Efficient and Effective Adaptation of Multiple Multimodal Foundation Models for Sequential Recommendation. arXiv 2025; under review by TKDE.

Chapter Positioning

The first two technical chapters focus on image-text settings. CROSSAN moves the thesis to richer raw-content environments where recommendation may depend on text, image, audio, and video foundation encoders at the same time. Its central question is how to adapt multiple modality-specific foundation models while preserving useful cross-modal interaction.

5.1 Introduction

Multimodal Foundation Models (MFMs) have advanced rapidly, with models like ViT (Dosovitskiy et al., 2020), BERT (Devlin et al., 2018), GPT (Brown et al., 2020), VideoMAE (Tong et al., 2022), and AST (Gong et al., 2021) demonstrating exceptional performance in representing a wide range of raw modalities. At the same time, the increasing availability of recommendation datasets (Y. Ni et al., 2023) containing raw multimodal data (e.g., images, text, video, audio, etc.) provides a natural avenue for exploring how these powerful models can be effectively adapted for sequential recommendation tasks.

One intuitive approach that preserves sufficient information is to adapt these models to the

raw modalities of recommendation datasets (Y. Li et al., 2023; C. Wu et al., 2021a; Yuan et al., 2023). Adaptation methods for MFMs that leverage raw modality information, such as fine-tuning and parameter-efficient fine-tuning (PEFT), are generally recognised for their ability to achieve better performance compared to traditional feature-based approaches (Y. Li et al., 2023; Q. Liu et al., 2023; Y. Ni et al., 2023; Yuan et al., 2023). However, these approaches have been largely sidelined due to the central challenge of the significant computational costs associated with existing adaptation methods for multiple MFMs. Both full fine-tuning and PEFT techniques, such as Adapters (Fu, Yuan, et al., 2024; Houlsby et al., 2019) and LoRA (E. J. Hu et al., 2021), become increasingly expensive as the number of modality encoders grows[1]. This computational bottleneck makes it more challenging to investigate the adaptation of additional MFMs, leaving the potential of these models for sequential recommendation largely unexplored.

Recently, the side adapter paradigm (Fu, Ge, et al., 2024; Sung et al., 2022; M. Xu et al., 2023b) has garnered significant attention for its superior efficiency in adaptation compared to traditional adapter-based or LoRA-based approaches. This efficiency stems from its fully decoupled design, which eliminates the need for gigantic computational graphs. However, existing studies on side adapters have primarily focused on single or dual-modal scenarios, leaving their potential for scalability to additional modalities and the integration of multiple multimodal foundation models (MFMs) unexplored. This gap highlights the need for further research to extend their applicability to more complex and diverse multimodal setups.

Motivated by the aforementioned challenges, this study aims to explore the under-explored area of adapting multiple ($>$2) multimodal foundation models (MFMs) for sequential recommendation and to propose a solution that enables efficient and effective adaptation. To this end, we introduce the Cross-modal Side Adapter Network (CROSSAN), a novel plug-and-play approach that addresses the computational inefficiencies of existing adaptation methods for multiple MFMs in item representation. To enhance multimodal interactions and improve overall effectiveness, we propose a cross-modal side adapter network, building on key insights from our preliminary study (Section 5.3). In our analysis, we demonstrate that cross-modal interaction enhances mutual information compared with independent side adapters. Moreover, in contrast to existing methods that fuse item representations through a fully connected layer (Fu, Ge, et al., 2024), we adopt the Mixture of Modality Expert Fusion (MOMEF). This approach incorporates a fine-grained gating mechanism that enables the adaptive integration of input modalities, providing a straightforward yet highly effective solution for capturing intricate multimodal interactions and enhancing representation fidelity. CROSSAN offers a scalable, efficient, and effective solution, achieving superior performance on public datasets. It outperforms existing efficient adaptation approaches, with its performance improvement becoming even more significant as more MFMs are integrated. Our contributions are listed below:

[1]Recommender systems are typically retrained on a daily or weekly basis (Y. Zhang et al., 2020), making costly training paradigms impractical in practice.

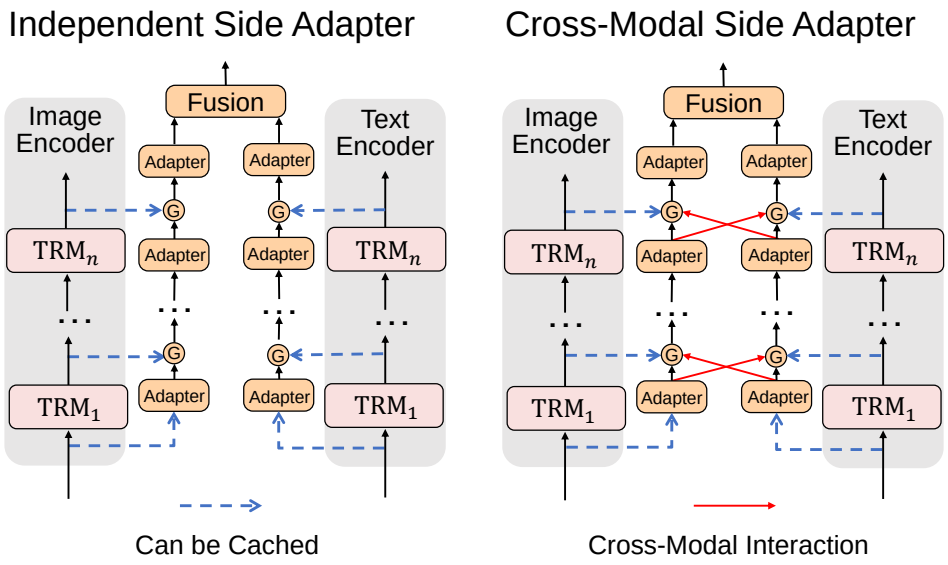


Figure 5.1: Independent vs. Cross-modal Side Adapter. During training, only the adapters and fusion layer are updated, while the rest of the foundation models remain frozen.

- To achieve enhanced multimodal representation learning while maintaining high efficiency, we introduce CROSSAN, a simple yet effective, cross-modality side adapter method.
- Building upon traditional concatenation-based fusion for multimodal item representation, we explore different fusion strategies and show that the Mixture of Modality Expert Fusion (MOMEF) mechanism effectively integrates item representations across modalities, leading to improved recommendation performance.
- Through extensive experiments, we demonstrate that CROSSAN delivers superior performance and efficiency on the public dataset. These results highlight the potential of leveraging multiple MFMs with raw modalities for the multimodal sequential recommendation, paving the way for future research in this direction.

5.2 Related Work

Multimodal Foundation Models (MFMs). Recent advances in multimodal learning leverage pre-trained models to enhance downstream task performance while reducing pre-training costs (Z. Chen et al., 2024; Ge, Fu, et al., 2024; Y. He et al., 2025; Q. Sun et al., 2023; Z. Zhang, Cao, & Wang, 2024). BERT (Devlin et al., 2018) pioneered the pretraining and fine-tuning paradigm in NLP, while Vision Transformer (ViT) (Dosovitskiy et al., 2020) adapted this approach for image classification. CLIP (Radford, Kim, et al., 2021) bridged vision and language through contrastive learning, enabling robust zero-shot capabilities. Large-scale models like GPT-4 (Achiam et al., 2023), T5 (Raffel et al., 2020), and multimodal variants like DALL-E

(Ramesh et al., 2021) and Flamingo (Alayrac et al., 2022) expanded the ability to process diverse modalities. In video representation learning, models such as SlowFast-R50 (Feichtenhofer et al., 2019), MViT-b (H. Fan et al., 2021), and VideoMAE (Tong et al., 2022) effectively capture both temporal and spatial features. Meanwhile, audio models like wav2vec 2.0 (Baevski et al., 2020) and the Audio Spectrogram Transformer (AST) (Gong et al., 2021) enhance audio classification by operating on spectrograms. Together, these models highlight the increasing strength of multimodal learning across a variety of tasks.

Efficient Adaptation of MFMs in RS. The RS field has progressively investigated the incorporation of diverse modalities to improve the effectiveness of recommendations (Bao et al., 2023; Y. Cheng et al., 2023; Fu, Ge, Xin, Yu, et al., 2025; H. Hu et al., 2023, 2024; Huan et al., 2024; Khan & Sisodia, 2025; R. Li et al., 2025; J. Lin et al., 2023; Q. Liu, Chen, et al., 2024; Q. Liu, Zhu, et al., 2024; Y. Liu et al., 2023; Y. Qu, Qu, Chen, Nguyen, & Yin, 2025; Z. Qu et al., 2023; R. Sun et al., 2020; Z. Tang et al., 2023; J. Wang et al., 2023; X. Wang et al., 2025; W. Wei et al., 2023; Y. Wei, Xu, Zhu, Ma, & Huang, 2024; Y. Wei et al., 2024; C. Wu et al., 2021b; J. Wu, Yu, He, Gao, & Gong, 2024; Ye et al., 2025; Yuan et al., 2023; L. Zhang, Zhou, Zeng, & Shen, 2024; X. Zhang et al., 2023; P. Zhou et al., 2025; X. Zhou, 2023). Recent studies (Elsayed et al., 2022; Y. Li et al., 2023; Y. Ni et al., 2023; Yuan et al., 2023) have demonstrated the superiority of the MoRec framework using end-to-end learning, showing it significantly outperforms traditional approaches that rely on offline feature extraction. For instance, (Y. Li et al., 2023) and (Fu, Ge, et al., 2024) highlighted the advantages of end-to-end training, which jointly leverages both image and text modalities, compared to methods that employ a single modality. Despite the strong performance of raw content learning, a major drawback of these approaches is the continued dependence on full fine-tuning of large multimodal encoders, leading to computational inefficiency.

Parameter-efficient fine-tuning (PEFT) methods have made strides in addressing this issue, as shown in works like M6-Rec (Z. Cui et al., 2022), Tallrec (Bao et al., 2023), and Adapter-Rec (Fu, Yuan, et al., 2024), which demonstrate that PEFT techniques can achieve competitive performance with significantly reduced overhead. However, many PEFT methods still rely on established approaches, often overlooking practical efficiency concerns.

Much of the existing research predominantly focuses on traditional Adapter or LoRA-based solutions and is limited to single-modality adaptation due to the inefficiency of these adaptation approaches. A recent study, IISAN (Fu, Ge, et al., 2024), introduced a structure that utilises independent adapters within each tower, complemented by a single inter-adapter for image and text adaptation. However, the inter-tower has a fixed input and lacks sufficient cross-modal interaction. In contrast, CROSSAN emphasises cross-modal side adapters with joint learning and updates. Additionally, IISAN considers only image and text scenarios, leaving many other existing modalities unexplored. Upon attempting to expand this method, we concluded that it lacked the desirable scalability for additional modalities. To the best of our knowledge, cross-

modality side adapters are largely underrepresented in the existing literature on recommender systems. Furthermore, CROSSAN investigates the novel area of scalable and efficient adaptation for more (>2) MFMs with raw modalities, which will facilitate future research.

5.3 Preliminary study: Cross or Independent?

To maximise efficiency, we adopt a recently advanced, fully decoupled, side adapter paradigm with a caching strategy (Fu, Ge, et al., 2024; Sung et al., 2022; M. Xu et al., 2023b), where the adapters are positioned outside the transformer models. This paradigm, while extensively studied for single-modality adaptation, remains underexplored in the context of multiple multimodal foundation models (MFMs). Therefore, we investigate two potential approaches: implementing the adapters either in a cross-modal configuration or independently, as illustrated in Figure 5.1. Preliminary experiments conducted on two commonly used combined modalities (text and image), using the MicroLens-100K dataset, indicate that the cross-modal approach outperforms the independent method across four evaluation metrics of Hit ratio and NDCG (Table 5.1). To understand this performance improvement, we further provide a theoretical analysis in Section 5.4.

Table 5.1: Performance comparison between INDSAN (Independent Side Adapter) and CROSSAN (Cross-Modal Side Adapter). "*" denotes that the improvements are significant at the level of 0.05 with a paired T-test.

Metric	INDSAN	CROSSAN
HR@10	0.0957	**0.0970***
HR@20	0.1373	**0.1393***
NDCG@10	0.0521	**0.0537***
NDCG@20	0.0626	**0.0644***
Mutual Information	0.0129	**0.2001***

5.4 Theoretical Analysis

In this section, we provide a theoretical analysis demonstrating why the proposed Cross-Modal Side Adapter (CROSSAN) outperforms the Independent Side Adapter (INDSAN). For clarity, we focus on a representative scenario involving text and image modalities.

5.4.1 Mutual Information as an Evaluation Metric

Let $Z^{(v)}$ and $Z^{(t)}$ denote the learned visual and textual representations at a given network layer. The mutual information (MI) (Paninski, 2003) between these representations is defined based

on entropy as:

$$I(Z^{(v)};Z^{(t)}) = H(Z^{(v)}) - H(Z^{(v)} \mid Z^{(t)}), \tag{5.1}$$

where $H(\cdot)$ denotes the entropy, which measures the uncertainty or information content of a random variable. Specifically, $H(Z^{(v)})$ quantifies the total uncertainty of the visual representations, while $H(Z^{(v)} \mid Z^{(t)})$ captures the remaining uncertainty in $Z^{(v)}$ after observing the textual representations $Z^{(t)}$.

A higher mutual information $I(Z^{(v)};Z^{(t)})$ implies that the textual representations $Z^{(t)}$ provide more information about the visual representations $Z^{(v)}$, thus reducing their uncertainty. This reflects stronger dependency and more effective cross-modal integration, indicating that the learned representations capture complementary and aligned information across modalities, which is desirable for downstream tasks. Many studies show that maximising mutual information enhances representation quality (W. Guo, Huang, Kong, & He, 2019; Liao et al., 2021; K. Zhou et al., 2020). Thus, we use it as the evaluation metric for multimodal learning.

5.4.2 Independent vs. Cross-Modal Adapters for Item Representation

Based on the definition of mutual information in terms of entropy (Paninski, 2003), where $H(\cdot)$ denotes entropy, the **INDSAN** model produces approximately independent modality-specific representations, denoted as $Z^{(v)}$ and $Z^{(t)}$. Specifically, we assume that the joint distribution of these representations satisfies $p\big(Z^{(v)},Z^{(t)}\big) \approx p\big(Z^{(v)}\big)p\big(Z^{(t)}\big)$, indicating near-independence between modalities. This near-independence implies that one modality provides minimal information about the other, leading to $H\big(Z^{(v)} \mid Z^{(t)}\big) \approx H\big(Z^{(v)}\big)$ and, consequently, negligible mutual information:

$$I_{\text{INDSAN}}\big(Z^{(v)},Z^{(t)}\big) = H\big(Z^{(v)}\big) - H\big(Z^{(v)} \mid Z^{(t)}\big) \approx 0.$$

In contrast, **CROSSAN** establishes dependency through cross-modal interactions at each layer, where

$$Z_l^{(v)} = f_l^{(v)}\big(X, Z_{l-1}^{(t)}\big), \quad Z_l^{(t)} = g_l^{(t)}\big(Y, Z_{l-1}^{(v)}\big).$$

This ensures that $p\big(Z_l^{(v)},Z_l^{(t)}\big) \neq p\big(Z_l^{(v)}\big)p\big(Z_l^{(t)}\big)$, meaning the uncertainty of the final representation $Z_L^{(v)}$ is reduced when $Z_L^{(t)}$ is known, i.e., $H\big(Z_L^{(v)} \mid Z_L^{(t)}\big) < H\big(Z_L^{(v)}\big)$. Consequently, the mutual information in CROSSAN is inherently positive:

$$I_{\text{CROSSAN}}\big(Z_L^{(v)},Z_L^{(t)}\big) = H\big(Z_L^{(v)}\big) - H\big(Z_L^{(v)} \mid Z_L^{(t)}\big) \geq 0.$$

This yields higher mutual information than INDSAN. This continuous cross-layer interaction aligns with prior findings, indicating that deeper, multi-level interactions enhance shared information across modalities (Gu et al., 2022; H. Wu et al., 2021; J. Yang et al., 2022).

To empirically validate our theoretical analysis, we evaluate the final outputs of the textual and visual adapters using the optimal checkpoints for both INDSAN and CROSSAN, pro-

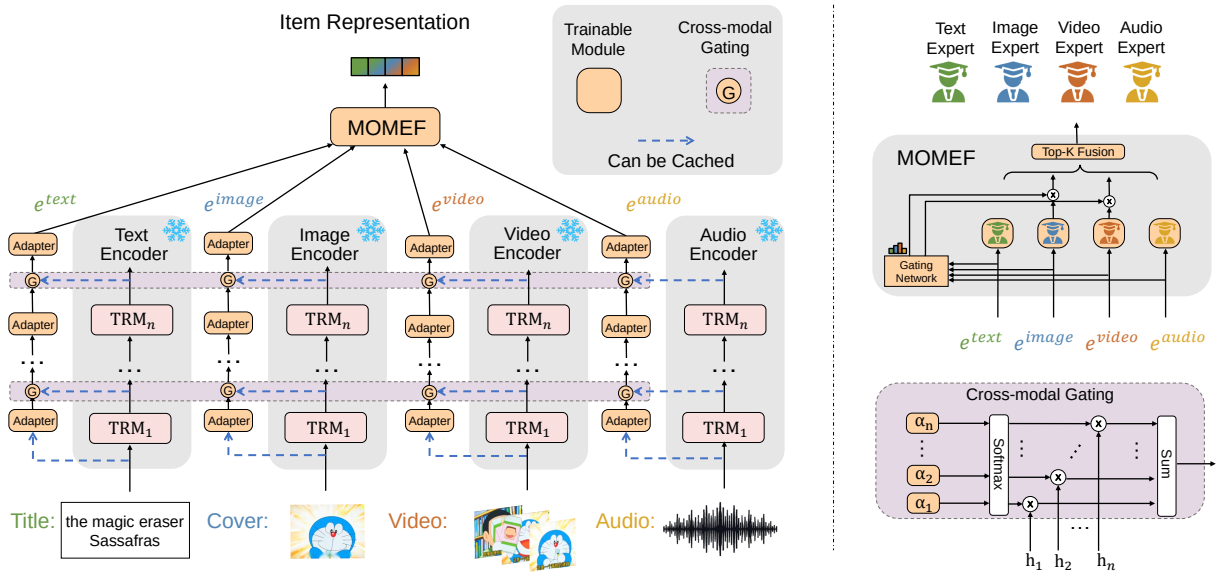


Figure 5.2: CROSSAN Overview. The example utilises four MFMs, each paired with its respective side adapters for adaptation. A cross-modal gating mechanism is applied, combining the outputs of each modality's side adapter at the same layer-level with gated fusion, ensuring effective interaction across modalities. MOMEF (Mixture of Modality Expert Fusion) treats each input modality as an expert. The gating network assigns probabilities to each expert, selecting the top-k experts based on these probabilities for further processing.

cessing all items from the MicroLens-100K dataset. The results corroborate our conclusion, demonstrating that CROSSAN achieves significantly higher mutual information than INDSAN. Specifically, as illustrated in Table 5.1, CROSSAN yields mutual information values more than 15 times greater than those observed with INDSAN. This notable difference supports the previous analysis and provides insights into the underlying reasons for CROSSAN's performance improvement.[2]

5.5 Methodology

In this work, we introduce CROSSAN, a plug-and-play approach designed to adapt multiple MFMs. CROSSAN is designed to provide effective multimodal representations in a scalable and efficient manner, offering a general solution for the adaptation of multiple MFMs. The overview of CROSSAN is shown in Figure 5.2. Our key innovation lies in designing a novel fully cross-modal gating between each modality's side adapters, dedicated to learning rich mutual information between modalities. Furthermore, to enable more effective multimodal fusion, we

[2]In the following section, we primarily adopt the state-of-the-art IISAN as the main comparison counterpart. In terms of mutual information, IISAN is theoretically equivalent to INDSAN due to the absence of dependencies between side adapters. The measured mutual information of IISAN is 0.0127, which is approximately the same as that of INDSAN. Compared to INDSAN, IISAN introduces an additional set of independent inter-modal side adapters that capture static multimodal information. However, dependencies among the side adapters are still absent.

propose a Mixture of Modality Expert Fusion (MOMEF) network, which dynamically combines the multimodal outputs of all towers to achieve fine-grained fusion for each item.

Problem Formulation. Given a recommendation dataset $\mathscr{D} = \{\mathscr{U}, \mathscr{V}\}$, where $\mathscr{U}$ and $\mathscr{V}$ represent the set of users and items respectively, our objective in a multimodal sequential recommendation task is to predict the next item that a user u will interact with, based on their past n interactions. For multimodal recommendation, each item v can have representations from M different modalities, such as text (v^{text}), image (v^{image}), video (v^{video}), and audio (v^{audio}). Although additional modalities can be incorporated, depending on the application, in this thesis, we mainly focus on these four standard modalities. Following (Fu, Ge, et al., 2024; Yuan et al., 2023), we process each of the modalities using their corresponding pre-trained foundational models, such as BERT (Devlin et al., 2018) for text, ViT (Dosovitskiy et al., 2020) for images, VideoMAE (Tong et al., 2022) for videos, and Audio Spectrogram Transformer (AST) (Gong et al., 2021) for audio. By leveraging these pretrained MFM backbones, we obtain the hidden states for each modality (e.g., b_i^{text}, b_i^{image}, b_i^{video}, and b_i^{audio}) from their transformer layers (TRM_i).

Cross-modal Gating. We employ a simple yet effective cross-modal gating mechanism, which differs from the traditional cross-attention mechanism proposed in (Huang et al., 2019). Unlike cross-attention, which relies on computationally intensive attention mechanisms, our approach is based on straightforward tunable weights, offering a lightweight and efficient alternative. Specifically, for each modality at the i-th layer, the input to the adapter combines the output of the adapter from the $(i-1)$-th layer with the hidden states from the corresponding MFM. For instance, taking the side adapter's text modality at the i-th layer as an example, we define its input as follows:

$$h_i^t = Adapter_i^t \left(\alpha_i^H b_i^{\text{text}} + \sum_{m \in M} \alpha_i^m h_{i-1}^m \right) \tag{5.2}$$

where $\alpha_i^H + \sum_{m \in M} \alpha_i^m = 1$. The gate α_i^H weights the current text-backbone hidden state b_i^{text}, while α_i^m weights the previous-layer side-adapter output for modality m. Here, b_i^{text} denotes the frozen text encoder's hidden state at the current backbone layer i, while h_{i-1}^m denotes the output of modality m's side adapter at the previous layer. The i-th adapter therefore combines current-layer backbone information with previous-layer adapter outputs.

Here, M represents the set of modalities considered in this work, specifically

$$M = \{\text{text}, \text{image}, \text{video}, \text{audio}\},$$

and t denotes the text modality. The parameter α controls the learnable weight for each layer. We utilise the adapter block design proposed by (Houlsby et al., 2019), as it has proven to be highly effective in sequential recommendation tasks (Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024).

Mixture-of-Modality Expert Fusion (MOMEF). Furthermore, to enhance multimodal repre-

sentation fusion while maintaining efficiency, we adopt a MOMEF method, drawing inspiration from the mixture-of-experts paradigm (D. Dai et al., 2024; Y. Zhou et al., 2022), which dynamically combines the multimodal outputs of all towers to achieve a fine-grained fusion for each item. Specifically, each modality $m \in M$ has its own expert network producing an output vector $f_m(i)$ for item i. We compute importance scores $w_m(i)$ using a gating network based on fully connected layers and select the top-k modalities $\text{TopK}(i)$ with the highest scores for each item. The final multimodal item representation e^{final} is then obtained by dynamically weighting and combining the outputs of these top-k modality experts:

$$e^{final} = \sum_{m \in \text{TopK}(i)} w_m(i) \cdot f_m(i) \tag{5.3}$$

where $\sum_{m \in \text{TopK}(i)} w_m(i) = 1$. This approach allows MOMEF to focus on the outputs of the most relevant modalities for each item, leading to a more effective and fine-grained multimodal fusion.

Subsequently, the vector e^{final} is fed into the sequential encoders to compute the final predicted score $\hat{y}_{ui}$ for user u and item i, which is calculated as the product of the sequential encoder output and the corresponding item embedding. Note that our entire framework maintains high efficiency, as all trainable modules are primarily composed of linear layers and gating mechanisms, without relying on attention mechanisms.

Loss Function. Regarding training, we adopt the commonly used in-batch debiased Cross-Entropy loss function $\mathscr{L}_{CE}$ (Fu, Ge, et al., 2024; Ji et al., 2023; Y. Ni et al., 2023; Yi et al., 2019; Yuan et al., 2023), which is defined as:

$$D_{ui} = \exp(\hat{y}_{ui} - \log(p_i)) + \sum_{j \in [B], j \notin I_u} \exp(\hat{y}_{uj} - \log(p_j)) \tag{5.4}$$

$$\mathscr{L}_{CE} = -\sum_{u \in \mathscr{U}} \sum_{i \in [2,\ldots,n+1]} \log \frac{\exp(\hat{y}_{ui} - \log(p_i))}{D_{ui}} \tag{5.5}$$

where p_i represents the popularity of item i, I_u denotes the set of items interacted by user u, and B is the batch size. The item $n+1$ refers to the predicted item for user u.

5.6 Experiment Setup

Dataset. To assess the effectiveness of adapting multiple MFMs for recommendation tasks, we consider datasets that contain more than two raw modalities. Specifically, we use the publicly available MicroLens-100K and the MicroLens-50K datasets provided in (Y. Ni et al., 2023).[3]

[3]To the best of our knowledge, MicroLens is the only publicly available dataset that includes three or more raw modalities (e.g., text, images, video, and audio). We leave the exploration of additional datasets for future work as and when more modality-rich datasets become available.

Table 5.2: Dataset Description.

Dataset	Users	Items	Interaction	Raw Content
MicroLens-100K	100,000	19,738	719,405	Text&Image&Video&Audio
MicroLens-50K	50,000	19,099	339,511	Text&Image&Video&Audio

The dataset statistics are presented in Table 5.2. Further evaluations on datasets with only two modalities are provided in Section 5.8.

Evaluation. Based on previous studies (Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024; Y. Ni et al., 2023; Yuan et al., 2023), our approach implements a leave-one-out evaluation strategy. Specifically, the final item in the interaction sequence is set aside for testing, the second-to-last item is used for validation, and the rest of the sequence is employed for training. To evaluate our model's performance, we consider the HR (Hit Ratio) and NDCG (Normalised Discounted Cumulative Gain) metrics, which are aligned with previous studies (Fu, Yuan, et al., 2024; Y. Ni et al., 2023; Yuan et al., 2023). Unless otherwise specified, all reported results correspond to the test set. We also note that the predicted item is evaluated against the entire set of items (Krichene & Rendle, 2020).

Implementation Details. We employ "bert-base-uncased", "vit-base-patch16-224", "MIT/ast-finetuned-audioset-10-10-0.4593", and "MCG-NJU/videomae-base" from Huggingface[4] as the text, image, audio, and video encoders, respectively.[5] Our choice is informed by previous research in the field (Dong, Liu, Chen, Polak, & Zhang, 2024; Fu, Ge, et al., 2024; Jain, 2022; Y. Ni et al., 2023; Yuan et al., 2023). For video processing, we use the first three seconds of footage and extract 16 frames for VideoMAE, following its original setup (Tong et al., 2022), with corresponding audio processed by AST. For side adapters, we employ LayerDrop, dropping half the layers of each foundation model for efficiency (Fu, Ge, et al., 2024; Sung et al., 2022). We utilise a transformer-based sequential encoder to model user sequences, following the approach outlined in (Y. Hou, He, et al., 2022; Yuan et al., 2023). The hidden dimension of the sequential encoder is set to 64 after a search in {32, 64, 128}, with two Transformer blocks and attention heads following (Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024). The learning rate is optimised between 1e-5 and 1e-3, keeping dropout at 0.1 (Y. Ni et al., 2023). We search batch sizes from 32 to 1024, selecting the largest based on GPU memory limits. Adapter hidden dimensions and LoRA ranks are tuned between 32 and 8192. Hyperparameters are determined by tuning on validation data, and all results are reported on the test set. All experiments are completed on an A6000 GPU.

[4] `https://huggingface.co/`

[5] The exploration of using LLMs as encoders is beyond the scope of this thesis, primarily due to the challenge of managing asymmetry across multiple MFMs. However, we provide a preliminary investigation of advanced LLM and LVM models in Section 5.8.

5.7 Experiment

Our evaluation addresses the following research questions:

- **RQ5.1:** How effective is CROSSAN compared with existing adaptation approaches, and does adapting more MFMs improve its performance compared to state-of-the-art efficient adaptation approaches?
- **RQ5.2:** How does CROSSAN's efficiency compare to state-of-the-art adaptation approaches?
- **RQ5.3**: How does each component affect the overall performance?
- **RQ5.4**: How do the key hyperparameters affect CROSSAN?

Table 5.3: Performance comparison of CROSSAN on MicroLens-100K and MicroLens-50K with four types of raw modalities (Text, Image, Video and Audio, denoted as T, I, V, and A, respectively). "*" indicates that the improvements of the best models compared with previous state-of-the-art methods are significant at the level of 0.05 with paired T-test. 'Relative Improvement' is computed in comparison to the corresponding IISAN and its extension.

	MicroLens-100K				MicroLens-50K			
Model	HR@10	NDCG@10	HR@20	NDCG@20	HR@10	NDCG@10	HR@20	NDCG@20
Full Finetuning (T+I)	0.0934	0.0499	0.1363	0.0607	0.0772	0.0408	0.1129	0.0498
Adapter (T+I)	0.0962	0.0514	0.1376	0.0618	0.0765	0.0407	0.1132	0.0500
LoRA (T+I)	0.0866	0.0462	0.1298	0.0571	0.0644	0.0331	0.0975	0.0415
IISAN (T+I)	0.0960	0.0526	0.1366	0.0628	0.0771	0.0421	0.1121	0.0509
IISAN-E (T+I+A)	0.0953	0.0523	0.1353	0.0623	0.0777	0.0422	0.1137	0.0513
IISAN-E (T+I+V)	0.0939	0.0517	0.1341	0.0619	0.0775	0.0428	0.1150	0.0522
IISAN-E (T+I+V+A)	0.0949	0.0524	0.1350	0.0625	0.0790	0.0430	0.1135	0.0517
CROSSAN (T+I) **(ours)**	0.0999*	0.0553*	0.1428*	0.0661*	0.0806*	0.0431*	0.1177*	0.0524*
CROSSAN (T+I+A) **(ours)**	0.1012*	0.0557*	0.1445*	0.0666*	0.0811*	0.0444*	0.1188*	0.0539*
CROSSAN (T+I+V) **(ours)**	0.1006*	0.0557*	0.1428*	0.0663*	0.0808*	0.0437*	0.1183*	0.0531*
CROSSAN (T+I+V+A) **(ours)**	**0.1033***	**0.0568***	**0.1452***	**0.0673***	**0.0847***	**0.0462***	**0.1222***	**0.0557***
Relative Improvement								
Text+Image	+3.86%	+4.99%	+4.43%	+5.03%	+4.25%	+2.24%	+4.81%	+2.99%
Text+Image+Audio	+5.87%	+6.17%	+6.39%	+6.43%	+4.17%	+4.95%	+4.34%	+4.88%
Text+Image+Video	+6.70%	+7.18%	+6.04%	+6.74%	+4.06%	+2.04%	+2.74%	+1.75%
Text+Image+Video+Audio	+8.10%	+7.70%	+7.00%	+7.15%	+6.73%	+6.97%	+7.13%	+7.21%

5.7.1 Effectiveness Evaluation (RQ5.1)

We compare CROSSAN with Full finetuning, Adapter (Houlsby et al., 2019), LoRA (E. J. Hu et al., 2021), and the state-of-the-art IISAN (Fu, Ge, et al., 2024), as well as its intuitive extension to support additional modalities. IISAN and IISAN-E serve as our primary multimodal adaptation baselines for two reasons: (1) with two modalities, IISAN achieves competitive performance compared to other methods, and (2) our limited GPU memory (48GB) prevents us from using full fine-tuning or adapter-based approaches on more than two modalities, given their already

high requirements for just two (see Table 5.5). Given that the original IISAN implementation is limited to two modalities, we extend its structure by incorporating additional intra-SAN layers to handle more multimodal foundation models (MFMs). In this expanded IISAN-E, we use a gated sum to combine the hidden states of all MFMs into the inter-SAN.

As shown in Table 5.3, CROSSAN achieves the best performance on the MicroLens-100K, outperforming all other adaptation approaches. Furthermore, we observe a progressive improvement in performance as more modalities from MFMs are incorporated. This trend becomes especially evident when compared to IISAN and its extension, IISAN-E, where the relative improvement increases with the introduction of additional modalities, as adding more MFMs to IISAN-E does not consistently result in better performance. This highlights the scalability and efficacy of CROSSAN over existing state-of-the-art efficient adaptation methods. To further validate these findings, we evaluate CROSSAN on the MicroLens-50K dataset (Table 5.3). The relative improvements remain consistent, with one exception: the T+I+V configuration shows comparable improvement to T+I. However, the overall trend of increasing performance with more MFMs and raw modalities persists. These results reaffirm the superiority of CROSSAN.

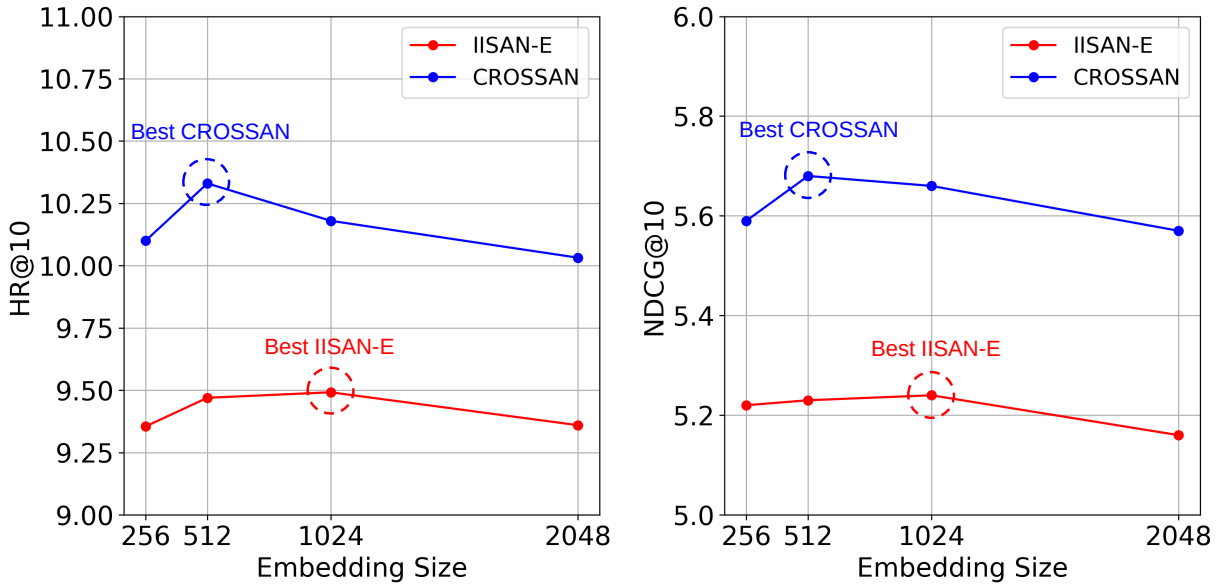


Figure 5.3: Optimal Embedding dimension for IISAN-E and CROSSAN

To further compare CROSSAN with additional state-of-the-art multimodal sequential recommendation models (MMSRs) beyond IISAN, we also include a recent state-of-the-art HM4SR model in our evaluation, as summarised in Table 5.4. Across all evaluation metrics, CROSSAN consistently outperforms these strong baselines, demonstrating its superior effectiveness. These results further highlight the advantages of adapting raw modalities through multimodal foundation models (MFMs).

(Answer to RQ5.1): After extensive evaluation, we conclude that CROSSAN demonstrates superior scalability compared to the state-of-the-art efficient adaptation approaches. This is

Table 5.4: Performance comparison for state-of-the-art multimodal sequential recommendation approaches on MicroLens datasets with respect to Hit Ratio (HR@K) and NDCG@K. The best results are in bold.

Method	HR@10	HR@20	NDCG@10	NDCG@20
MicroLens-100K				
HM4SR (WWW'25)	0.0807	0.1207	0.0437	0.0538
IISAN (SIGIR'24)	0.0960	0.1366	0.0526	0.0628
CROSSAN (T+I)	0.0999	0.1428	0.0553	0.0661
CROSSAN (T+I+V+A)	**0.1033**	**0.1452**	**0.0568**	**0.0673**
MicroLens-50K				
HM4SR (WWW'25)	0.0661	0.0976	0.0367	0.0446
IISAN (SIGIR'24)	0.0771	0.1121	0.0421	0.0509
CROSSAN (T+I)	0.0806	0.1177	0.0431	0.0524
CROSSAN (T+I+V+A)	**0.0847**	**0.1222**	**0.0462**	**0.0557**

Table 5.5: Efficiency Comparison. TT, GM, and TP stand for Training Time (seconds/epoch), GPU Memory (MB), and Trainable Parameters, respectively. A lower value for each metric indicates an improvement in efficiency. We demonstrate the improvement over IISAN-E.

Method	TT (↓)	GM (↓)	TP (↓)
Full Finetuning (T+I)	3,278	45,886	194,897,216
Adapter(T+I)	2,856	35,652	38,017,088
LoRA(T+I)	3,110	36,902	37,992,512
IISAN-E$_{best}$ (T+I+V+A)	213	5,556	58,432,824
IISAN-E$_{same}$ (T+I+V+A)	207	4,476	30,889,784
CROSSAN (T+I+V+A) **(ours)**	**144**	**4,272**	**24,741,456**
Improvement$_{best}$	+32.39%	+23.11%	+57.66%
Improvement$_{same}$	+30.43%	+4.56%	+19.90%

evidenced by its more substantial performance improvement when additional raw modalities are incorporated.

5.7.2 Efficiency Evaluation (RQ5.2)

In this section, we explore the efficiency of CROSSAN in terms of three dimensions: Training time, GPU memory, and Trainable Parameters, following the work by (Fu, Ge, et al., 2024). We primarily focus on the **training-time** and **GPU Memory**, since they are the most important aspects of efficiency in practical settings. Due to computational limitations, we were only able to evaluate the efficiency of traditional adaptation approaches on image and text modal-

ities[6]. Therefore, we primarily compare CROSSAN with IISAN-E (the intuitive extension of IISAN (Fu, Ge, et al., 2024)), as other methods are too computationally intensive and not suitable for direct comparison with CROSSAN. In the following, we default to reporting efficiency based on the best performance.

Table 5.6: Ablation Study on Fusion Method. D-Gated and S-Gated Fusion represents Dynamic Gated and Static Gated Fusion. Concat Fusion refers to the direct concatenation of all modalities.

Dataset	Method	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-100K	MOMEF	**0.1033**	**0.0568**	**0.1452**	**0.0673**
	D-Gated Fusion	<u>0.0994</u>	<u>0.0551</u>	<u>0.1416</u>	<u>0.0657</u>
	S-Gated Fusion	0.0972	0.0532	0.1394	0.0638
	Concat Fusion	0.0965	0.0531	0.1380	0.0636
MicroLens-50K	MOMEF	**0.0847**	**0.0462**	**0.1222**	**0.0557**
	D-Gated Fusion	0.0796	0.0430	<u>0.1196</u>	0.0530
	S-Gated Fusion	0.0807	0.0437	0.1171	0.0529
	Concat Fusion	<u>0.0816</u>	<u>0.0442</u>	0.1181	<u>0.0534</u>

Table 5.7: Ablation Study on Cross- vs. Independent-Modal. H and N represent the Hit Ratio and NDCG.

Dataset	Modality	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-100K	Cross-modal	**0.1033**	**0.0568**	**0.1452**	**0.0673**
	Independent	0.0993	0.0546	0.1418	0.0654
MicroLens-50K	Cross-modal	**0.0847**	**0.0462**	**0.1222**	**0.0557**
	Independent	0.0815	0.0439	0.1169	0.0528

As shown in Table 5.5, traditional adaptation methods with only two modalities reach the maximum of our available GPU memory. While IISAN-E reduces GPU memory, our proposed method, CROSSAN, is even more efficient and achieves better performance (see Table 5.3). We attribute this observation to two key factors: (1) CROSSAN reduces the computational burden by using only one adapter tower per modality in each Multimodal Foundation Model (MFM), whereas IISAN-E requires an additional inter-modal adapter tower; (2) Hyperparameter tuning revealed that IISAN-E achieves its best performance with an embedding size of 1024, while CROSSAN reaches optimal performance with a hidden dimension of only 512, as shown in

[6]Adopting full fine-tuning or PEFT for the four encoders is nearly impossible given our GPU resources (we have access to only one A6000 GPU). Therefore, we leave this investigation for future work or for institutions with more extensive computational resources.

Figure 5.3. This results in IISAN-E having more trainable parameters, which contributes to its reduced efficiency. Even when IISAN is configured with the same embedding dimension as CROSSAN, it remains less efficient due to the additional inter-modal adapter tower. **(Answer to RQ5.2):** Although CROSSAN achieves superior performance, it also exceeds the efficiency of the state-of-the-art IISAN model.

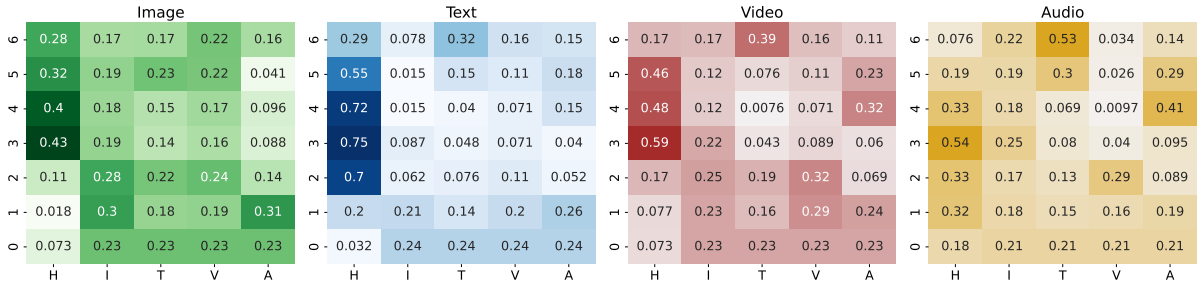


Figure 5.4: The heatmap visualisation of CROSSAN's gates, where deeper colours indicate higher values. The y-axis corresponds to the layers of the side adapter, ranging from bottom (0) to top (6). "H" denotes the hidden state of the current layer's modality, while "I", "T", "V", and "A" represent the gate values for the previous side adapter layer's outputs corresponding to their respective modality.

5.7.3 Ablation Study (RQ5.3)

In this section, we present an ablation study focusing on two key components of CROSSAN: the fusion mechanism, MOMEF, and cross-modal interaction.

For the fusion mechanism, we compare MOMEF against three commonly used fusion methods: (1) Concat Fusion, which is widely adopted in existing literature (Fu, Ge, et al., 2024; J. Hu, Liu, Zhao, & Jin, 2021; Y. Li et al., 2023); (2) Static Gated Fusion (S-Gated), where a learnable gate is assigned to each modality and remains fixed after training, applying the same weights to all items; and (3) Dynamic Gated Fusion (D-Gated), which leverages a fully connected layer to generate different weights for each item, allowing more flexibility based on task-specific inputs. The proposed method, MOMEF, utilises dynamic gating of input modalities to selectively activate the top-k modalities for each item. This approach offers a more fine-grained mechanism, allowing not only the weighting of modalities but also the precise selection of the most useful modalities.

As shown in Table 5.6, MOMEF outperforms all other methods, demonstrating its superior ability to fuse multiple modalities efficiently. While the D-gated method achieves the second-best results on the MicroLens-100K dataset, it underperforms on the MicroLens-50K dataset across three metrics when compared to direct concat. This suggests that other approaches may be less robust across different datasets.

Regarding cross-modal interaction, the results in Table 5.7 further confirm our preliminary findings (Section 5.3): incorporating cross-modal interactions significantly enhances multi-

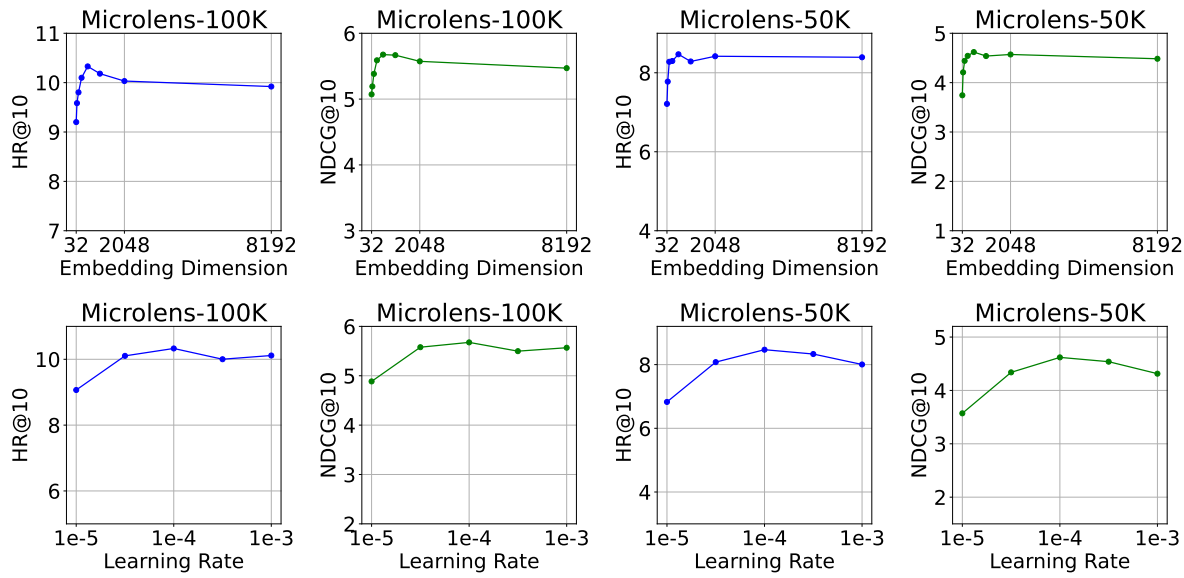


Figure 5.5: The first row of figures depicts CROSSAN's embedding dimension adaptation, where the embedding dimensions of the side adapters range from 32 to 8192. The second row of figures illustrates CROSSAN's learning rate adaptation, with the learning rate varying between 1e-5 to 1e-3.

modal learning, validating its importance in achieving better performance. Additionally, we observed that the gating mechanism is crucial; without it, the model struggles to train properly, leading to a collapse in performance. We present the heatmap in Figure 5.4, where we observe that the gate values for the hidden states in the corresponding modality tower are significantly larger in the middle layers, while the values in the lower layers are relatively smaller. This emphasises the importance of the middle layers in MFMs. **(Answer to RQ5.3):** Both MOMEF and cross-modal interaction contribute to the overall performance of CROSSAN, with each approach demonstrating its usefulness.

Table 5.8: Top-K experts for CROSSAN.

Dataset	Top-K	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-100K	4	0.0992	0.0545	0.1422	0.0653
	3	0.1002	0.0558	0.1459	0.0673
	2	**0.1033**	**0.0568**	**0.1452**	**0.0673**
	1	0.1014	0.0560	0.1444	0.0668
MicroLens-50K	4	0.0821	0.0444	0.1189	0.0536
	3	**0.0847**	**0.0462**	**0.1222**	**0.0557**
	2	0.0815	0.0438	0.1197	0.0534
	1	0.0822	0.0451	0.1178	0.0541

Table 5.9: Performance (%) comparison across methods on three datasets. Best results in bold, second-best underlined.

Method	Scientific		Instrument		Office	
	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
FFT	6.62	4.06	8.76	6.76	6.30	4.58
Adapter	6.61	3.91	8.82	6.83	6.65	4.85
LoRA	6.62	4.09	8.43	6.64	6.55	4.78
BitFit	6.37	3.76	8.65	6.71	6.78	4.87
IISAN	<u>6.83</u>	<u>4.14</u>	<u>9.06</u>	<u>7.01</u>	<u>6.80</u>	<u>4.92</u>
CROSSAN	**6.90**	**4.22**	**9.08**	**7.05**	**6.83**	**4.95**

5.7.4 Hyperparameter Analysis (RQ5.4)

In this section, we explore three key hyperparameters: (1) learning rate, (2) hidden dimension, and (3) number of experts. The former two are fundamental parameters commonly explored in adapter-based recommendation models (Fu, Yuan, et al., 2024), while the number of experts is introduced by MOMEF. To optimise the model's performance, we conduct an extensive hyperparameter search for these settings.

Embedding Dimension. As shown in Figure 5.5, the performance of CROSSAN demonstrates a clear dependency on the embedding dimension size. A relatively large embedding dimension is essential for effective adaptation. In contrast, smaller embeddings lead to noticeable drops in performance. However, due to the efficiency of CROSSAN's adapter-based architecture, increasing the embedding dimension does not result in significant computational overhead, such as extended training time or excessive GPU memory usage. Notably, when scaling the embedding size up to 8192, performance remains stable, suggesting that CROSSAN maintains its effectiveness as long as the embedding size exceeds a certain threshold.

Learning Rate. Figure 5.5 illustrates the effect of learning rate on model performance. CROSSAN's performance appears to remain stable once the learning rate exceeds 5e-5. However, using a smaller learning rate (e.g., 1e-5) results in suboptimal performance. This highlights the necessity of fine-tuning this hyperparameter within a suitable range to achieve optimal performance.

Top-K experts. The Top-K is a new hyperparameter introduced through MOMEF. As shown in Table 5.8, the optimal number of experts differs across datasets. For example, the MicroLens-100K dataset performs best with two experts, while three experts yield the best results for the MicroLens-50K dataset. These findings indicate that both very large and small numbers of experts are ineffective.

(Answer to RQ5.4): Based on the experiments, we conclude two observations from the hyperparameter analysis: (1) The embedding dimension and learning rate for CROSSAN should be set to relatively large values to ensure stable performance. (2) Top-K experts are dataset-specific and typically lie within a moderate range.

Table 5.10: Performance comparison on MicroLens-100K.

Method	H@10	H@20	N@10	N@20
CROSSAN	0.1033	0.1452	<u>0.0568</u>	0.0673
CROSSAN (Qwen-7B)	0.1033	**0.1464**	0.0564	<u>0.0674</u>
CROSSAN (Qwen-VL-7B)	0.1033	<u>0.1460</u>	**0.0570**	**0.0677**

Table 5.11: Performance Comparison of Independent vs. Cross-Modal Architectures with Cross-Attention Fusion on the MicroLens-100K Dataset.

Modality	H@10	N@10	H@20	N@20
Independent (w/ MOMEF Fusion)	0.0993	0.0546	0.1418	0.0654
Independent (w/ Cross-attn Fusion)	0.1008	0.0558	0.1423	0.0663
Cross-modal (w/ MOMEF Fusion)	0.1033	0.0568	0.1452	0.0673
Cross-modal (w/ Cross-attn Fusion)	0.1014	0.0561	0.1435	0.0668

Table 5.12: Performance comparison of Independent and Cross-modal methods.

Method	Top-K	H@10	N@10	H@20	N@20
Independent	1	0.0973	0.0535	0.1404	0.0643
	2	**0.0993**	**0.0546**	0.1418	0.0654
	3	**0.0993**	**0.0546**	**0.1432**	0.0657
	4	0.0989	0.0540	0.1411	0.0646
Cross-modal	1	0.1014	0.0560	0.1444	0.0668
	2	**0.1033**	**0.0568**	0.1452	**0.0673**
	3	0.1002	0.0558	**0.1459**	**0.0673**
	4	0.0992	0.0545	0.1422	0.0653

5.8 More evaluations

Exploration of More Datasets. To further investigate the effectiveness of CROSSAN, we conducted experiments on three Amazon datasets used in (Fu, Ge, et al., 2024). We compared the performance of CROSSAN against full fine-tuning and various PEFT methods, including Adapter, LoRA, BitFit, and IISAN. Our results show that CROSSAN achieves the best adaptation performance across all three datasets using only two modalities as described in Table 5.9. Although the improvement over IISAN on these datasets is not large, CROSSAN offers greater efficiency by requiring only two side adapter towers, compared to the three needed by IISAN. Consequently, even with just two modalities, CROSSAN proves to be an optimal solution for these datasets.

Exploration of LLM and LVM. To assess the impact of incorporating a recent large language model (LLM)[7] and large vision-language model (LVM)[8], we follow the implementation of (Fu, Ge, Xin, Karatzoglou, et al., 2025) by replacing the original text tower. This substitution yields a slight performance improvement for CROSSAN when using Qwen-VL-7B, while Qwen-7B performs marginally worse in terms of NDCG@10 and marginally improves the other metrics (Table 5.10). Nonetheless, while the implementation of LLMs can provide slight performance gains, the simple BERT-based text tower used in our main paper already delivers strong and competitive results.

More Investigation on Independent Variant. In some cases, the independent side-adapter variant still has merit, as it avoids layer-wise cross-modal operations and may be more suitable for asymmetrical architectures (Fu, Ge, Xin, Karatzoglou, et al., 2025). Understanding why it underperforms compared to the cross-modal approach is therefore meaningful. A natural hypothesis is that the independent variant underperforms because it lacks sufficient interaction during layer-wise fusion, and the sparse fusion in MOMEF may not fully realise its potential. To investigate this, we first examine the effect of varying the number of experts to mitigate the sparsity factor, as summarised in Table 5.12. The results indicate that the independent approach requires a relatively larger number of experts in MOMEF. In contrast, the cross-modal method reduces the optimal number of experts by promoting a more unified feature representation, thanks to its higher mutual information.

To further enhance the final fusion interaction, we adopt a commonly used cross-attention fusion module to the independent variant. Unlike MOMEF, this module performs dense computation (see Table 5.11). The results support our hypothesis: cross-attention yields small improvements over MOMEF fusion in the independent variant (e.g., HR@10 improves from 0.0993 to 0.1008), aligning with the intuition that cross-attention can help in this setting. However, when integrated into CROSSAN's interaction layers, performance slightly declined (e.g., Hit@10: 0.1033 $\rightarrow$ 0.1014), suggesting that cross-attention is not always advantageous when

[7] https://huggingface.co/Qwen/Qwen2-7B

[8] https://huggingface.co/Qwen/Qwen2-VL-7B

strong cross-modal interactions are already present. These findings may provide useful insights for future work that adopts the independent variant, highlighting that cross-attention can be beneficial compared with MOMEF.

5.9 Conclusion

CROSSAN provides a scalable, efficient, and effective approach for adapting multiple multimodal foundation models (MFMs) in sequential recommendation tasks. By incorporating cross-modal side adapters along with the Mixture of Modality Expert Fusion (MOMEF), CROSSAN achieves superior performance. Extensive experimental results validate the approach's ability to improve recommendation effectiveness as additional modalities are integrated, demonstrating its superiority over existing methods.

Future research can extend CROSSAN to various multimodal tasks, including multimodal classification (Gómez-Chova, Tuia, Moser, & Camps-Valls, 2015), retrieval (Rafailidis, Manolopoulou, & Daras, 2013), and generative modelling (W. Dai et al., 2022), where the integration of diverse data modalities is critical. These directions offer significant potential to improve both the efficiency and performance of multimodal learning across a broad range of applications, positioning CROSSAN as a general adaptation approach for future research.

Chapter 6

Stresa: Stream-Aware Side Adaptation for Large Multimodal Embedding Models

Publication Note

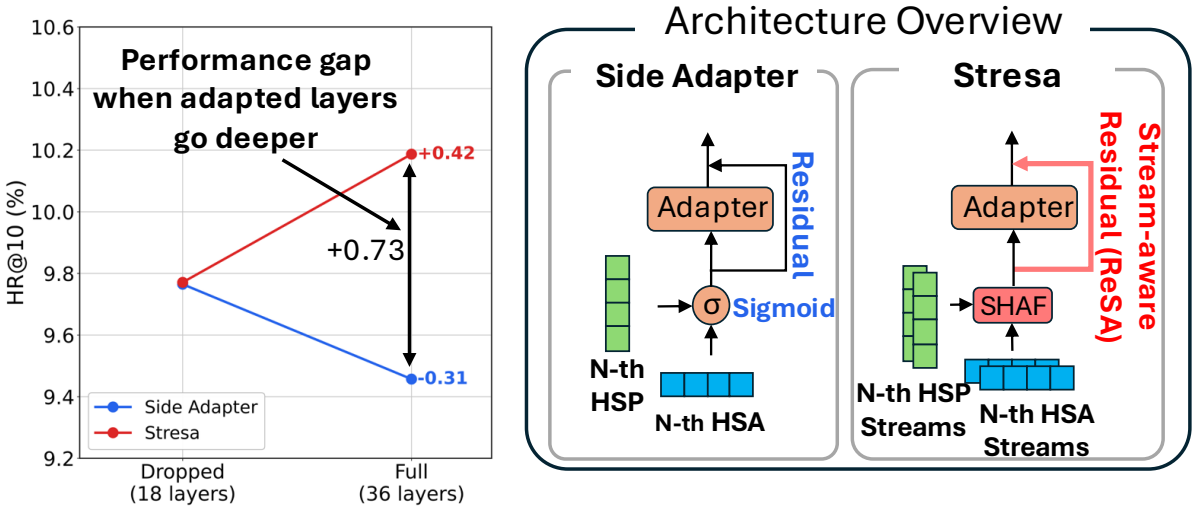
This chapter is adapted from: Junchen Fu, Kaiwen Zheng, Ioannis Arapakis, Wenhao Deng, Xin Xin, Joemon M. Jose, and Xuri Ge. Stream-aware Side Adaptation for Large Pre-trained Multimodal Embedding Models in Sequential Recommendation. Accepted by ACM MM 2026.

Chapter Positioning

Building on the multi-tower adaptation frameworks developed in IISAN-Versa and CROSSAN, this chapter shifts the focus to unitower multimodal embedding models, which have become increasingly powerful as general-purpose representation providers. While previous methods benefit from the collaborative and interactive capacity among separate modality towers, directly applying their basic side-adapter design to unitower embedding models proves insufficient, particularly when adaptation reaches deeper hidden states. Stresa addresses this limitation by studying stream-aware side adaptation for unitower multimodal embedding models, aiming to preserve, fuse, and update internal representation streams for recommendation without damaging the pretrained unified embedding space.

6.1 Introduction

Large-scale pretrained multimodal embedding models are increasingly becoming reusable infrastructure for retrieval, ranking, and recommendation, as they project heterogeneous item content into transferable semantic spaces (Y. He et al., 2025; Jia et al., 2021; J. Li et al., 2023; Radford, Kim, et al., 2021; L. Wang et al., 2022). A recent state-of-the-art example is Qwen3-



Method	Residual	Fusion
Stresa	Selective residual update (ReSA)	Earlier Representation Preserved Fusion (SHAF)
Side Adapter	Direct Addition (Residual)	Direct Sigmoid Fusion (Sigmoid)

Figure 6.1: **Conventional Side Adapter vs. Stresa.** Using the frozen Qwen3-VL-8B-Embedding backbone, *Dropped 18 layers* and *Full 36 layers* mean that the item encoder extracts hidden states from 18 or 36 backbone layers for adapter computation. Left: HR@10 under both settings. Right: Side Adapter uses residual addition and sigmoid fusion, whereas Stresa uses selective residual update (ReSA) and earlier-representation-preserved fusion (SHAF). *HSP* and *HSA* denote hidden states from the pretrained model and previous adapter layer, respectively.

VL-Embedding, which demonstrates the power of this paradigm by mapping text, images, document images, and video into a unified representation space with strong multimodal retrieval performance (M. Li et al., 2026). This is especially promising for multimodal sequential recommendation, where the quality of an item's multimodal representations directly shapes next-item prediction (J. Hu et al., 2024; Z. Li et al., 2026; Y. Wang et al., 2025; Z. Wang et al., 2025; Z. Zhou, Meng, & Zhang, 2025). When the predictor operates on semantically rich multimodal embeddings, it can exploit content continuity, semantic substitutability, and cross-modal expressions of user intent in addition to collaborative signals (Fu, Ge, et al., 2024; Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025; Y. Ni et al., 2025). Stronger multimodal embedding backbones should therefore provide a stronger foundation for sequential recommendation.

However, directly applying large pre-trained multimodal embedding models to sequential recommendation often leads to suboptimal performance due to domain misalignment (Fu, Ge, et al., 2024). These pretrained embedding models are typically learned for general semantic representation rather than recommendation-specific objectives, while sequential recommendation depends on user behaviour, transition patterns, and recommendation relevance that can

differ substantially from standard semantic similarity (Yuan et al., 2023). Although end-to-end adaptation can effectively bridge this gap, it is often impractical for deployment because of its high computational and memory cost (R. Li et al., 2025). This trade-off is especially important in sequential recommendation, where the same item encoder is reused over a large catalogue and efficient offline embedding extraction is often crucial for deployment. This challenge remains significant not only for full fine-tuning but also for conventional parameter-efficient tuning when the backbone model is large (Sung et al., 2022). In this setting, side adaptation (Fu, Ge, et al., 2024; Fu, Ni, et al., 2025) is particularly appealing because it keeps the backbone frozen while learning only a lightweight side branch, enabling efficient training, offline caching of item representations, and practical deployment at scale, while still retaining strong recommendation performance. Motivated by this promise, recent works have begun to explore side-adapter-style designs for sequential recommendation with large embedding backbones (Fu, Ge, et al., 2024; Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025; M. Xu et al., 2023b). In this work, we focus specifically on frozen, offline-cacheable item-side adaptation of a *unified single-tower* multimodal embedding backbone. The resulting item representations are user-independent, while the downstream sequential model is treated as a shared component rather than part of the proposed adaptation architecture.

Most prior methods (Fu, Ge, et al., 2024; Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025) adopt a conventional side-adapter design, in which a lightweight bottleneck adapter with a residual connection is attached to each backbone layer to adapt its hidden states. Although this paradigm is computationally efficient, we find that its performance becomes suboptimal when extended to full-depth adaptation, as shown in Figure 6.1. Existing studies (Fu, Ge, et al., 2024; Fu, Ge, Xin, Karatzoglou, et al., 2025; Sung et al., 2022) observe the same issue and therefore treat layer dropping as a default setting, since adapting more hidden layers can even degrade performance.

We argue that this limitation stems from two key challenges: (1) the lack of modelling of how fused representations are selected during direct residual addition, and (2) the inadequate preservation of earlier representations during progressive sigmoid fusion. **First**, a residual connection does not explicitly model which components should be retained or suppressed in fused representations. In side adaptation, the propagated representation at each layer is already a fused embedding of the current hidden states and the previous adapter's output (Fu, Ge, et al., 2024; Sung et al., 2022). For such representations, effective propagation requires deciding which components should be preserved and which should be suppressed. Plain residual addition does not model this process, which may limit the quality of the propagated embedding. **Second**, simple fusion (Fu, Ge, et al., 2024; Sung et al., 2022) cannot effectively preserve earlier representations during progressive aggregation. Since side adaptation proceeds by continuously incorporating newly emerging layer-wise information, the model should also retain useful representations accumulated from earlier layers. However, simple sigmoid-based fusion is often insufficient for

this purpose, causing earlier information to be gradually overwritten as depth increases. These observations motivate the central question of this thesis: *how can we design effective side adaptation for deep unified multimodal embedding models?*

To this end, we propose *Stresa*, **Stre**am-aware **S**ide **A**daptation for large pre-trained multimodal embedding models in sequential recommendation. Stresa addresses the two challenges above by redesigning the two core components of conventional side adapters: residual connection and sigmoid fusion. To introduce explicit selection into the residual connection, inspired by the empirical success of stream-aware residual designs in large language models (D. Zhu et al., 2024), we introduce *ReSA*, a residual stream adapter tailored for side-adapter blocks. To move beyond simple sigmoid fusion toward stronger fusion and better preservation of earlier-layer representations, we further propose *Stream-aware Hidden-Adapter Fusion* (SHAF), which not only produces a stream-structured fused representation for ReSA, but also preserves historical side memory from earlier adapter layers. Together, ReSA and SHAF enable stronger fusion, more stable memory propagation, and more effective deep side adaptation.

Our contributions are summarised as follows:

- We propose *Stresa*, a **stre**am-aware **s**ide **a**daptation paradigm that enables deep adaptation for unified large pre-trained multimodal embedding models.
- Stresa introduces the *Residual Stream Adapter* (ReSA) for selection modelling in residual connection via stream-aware residual adaptation, and the *Stream-aware Hidden-Adapter Fusion* (SHAF) for fine-grained stream-aware fusion with better preservation of earlier-layer representations.
- We provide both structural analysis and empirical evidence showing that Stresa better preserves earlier representations and remains effective under deep adaptation. Across multiple datasets and two Qwen3-VL-Embedding scales, Stresa consistently outperforms the standard Side Adapter and strong prior baselines.

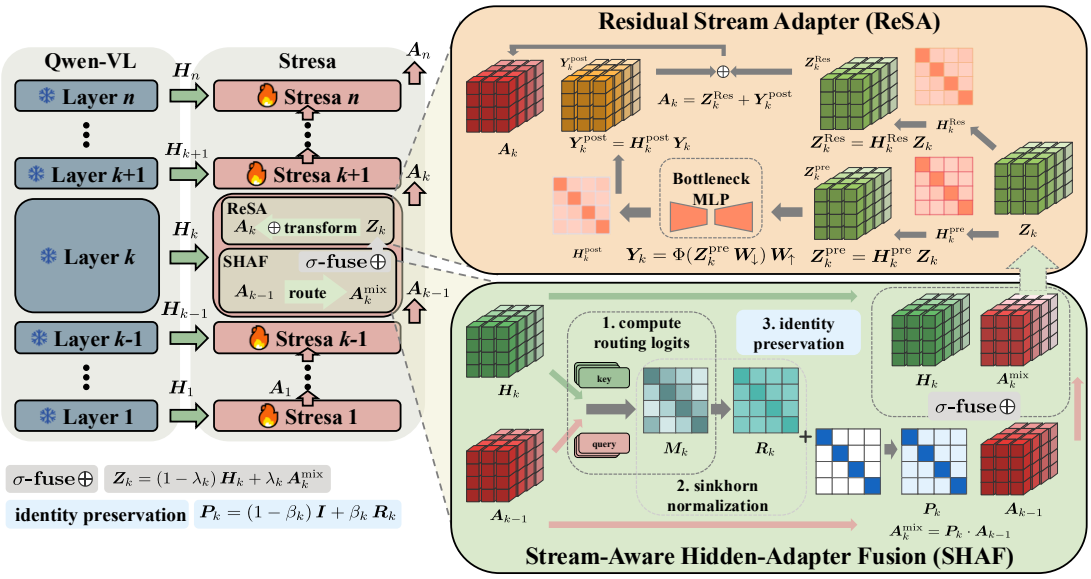


Figure 6.2: Overview of the proposed stream-aware fusion framework. (**1**) **Stream-aware Hidden-Adapter Fusion (SHAF)** at layer k first reshapes the hidden representations into multiple streams and computes routing logits via query-key interactions. An identity-preservation operation is applied to preserve information from earlier layers. (**2**) **Residual Stream Adapter (ReSA)** applies stream-wise pre-, post-, and residual mappings around a bottleneck MLP. The mapped residual and nonlinear branches are combined to produce the next side state $\mathbf{A}_k$, which is directly propagated to the following adapted layer.

6.2 Related Work

Pre-trained Multimodal Embedding Models. Pretrained encoders first established the value of reusable contextual representations for downstream matching and ranking (Devlin, Chang, Lee, & Toutanova, 2019; Y. Liu et al., 2019). Multimodal representation learning then extended this idea to image–text and vision–language settings through models such as CLIP, ALIGN, BLIP-style architectures, and more recent general-purpose embedding backbones (Y. He et al., 2025; Jia et al., 2021; J. Li et al., 2023; M. Li et al., 2026; Radford, Kim, et al., 2021; L. Wang et al., 2022). These models are increasingly used as reusable embedders, which makes them a natural foundation for recommendation when items contain rich text and visual content. Recently, the Qwen team proposed the Qwen3-VL-Embedding models (M. Li et al., 2026), which achieve state-of-the-art results in multimodal representation learning. This trend is particularly relevant to sequential recommendation. Recent multimodal recommendation benchmarks such as MicroLens (Y. Ni et al., 2025) highlight that user behaviour often depends on content-rich items, making large pre-trained multimodal embedding models a strong starting point for item representation. Recent recommendation-oriented adaptation frameworks further show that frozen or lightly adapted multimodal backbones can materially improve recommendation (Bao et al., 2023; Fu, Yuan, et al., 2024; Geng et al., 2023; S. Kim et al., 2025; Q. Liu, Zhu, et al., 2024; Y. Liu, Zhang, Ren, et al., 2024; H. Wang, Liu, et al., 2025; Ye, Fu, Song, Zheng, & Jose, 2026;

Zhuang et al., 2025). These embeddings have also been adopted in the generative retrieval paradigm for recommendation (Fu, Ge, Karatzoglou, et al., 2026; W. Sun et al., 2026).

Parameter-Efficient Adaptation and Side Adaptation. Parameter-efficient adaptation methods such as adapters, prompt tuning, and low-rank updates reduce trainable parameter counts relative to end-to-end tuning (Houlsby et al., 2019; E. J. Hu et al., 2022; Lester, Al-Rfou, & Constant, 2021). In recommendation, however, the relevant design criterion is not only parameter count. Because the same item encoder is reused across a large catalogue, adaptation methods that preserve a frozen backbone are especially attractive for caching, deployment, and offline feature extraction. Side adaptation (Sung et al., 2022; M. Xu et al., 2023b) is a natural fit for this constraint. It learns a lightweight side adapter while keeping the backbone frozen, thereby avoiding the need to store the backbone's backward computation graph. Subsequent work, including Quantised Side Tuning, further explored lightweight side branches as a way to complement frozen LLMs (Z. Zhang, Zhao, et al., 2024). In multimodal sequential recommendation, recent methods have shown that multimodal information can be used as cached embeddings to improve efficiency. For example, IISAN (Fu, Ge, et al., 2024) introduces this paradigm, IISAN-Versa (Fu, Ge, Xin, Karatzoglou, et al., 2025) extends it by adapting larger asymmetric two-tower encoders of different scales, and CROSSAN (Fu, Ni, et al., 2025) further demonstrates the effectiveness of side adaptation across four modalities. These studies mainly focus on efficiency and adaptation design in multi-tower architectures. However, most of them rely on LayerDrop (Sung et al., 2022) and do not address the stability of deep side-memory propagation in such models. Besides these, many state-of-the-art studies (X. Cui, Lu, Tong, Li, & Zhao, 2025; J. Dai et al., 2026; X. Guo, Zhang, Wang, et al., 2025; X. Guo, Zhang, Xue, et al., 2025; J. Liang et al., 2023; W. Lu & Yin, 2025; F. Meng et al., 2025; Song et al., 2025; J. Xu et al., 2025; W. Yang et al., 2025; Ye et al., 2025; L. Zhang et al., 2024; S. Zhang, Chen, Shen, Wang, & Xiong, 2025; J. Zhu et al., 2025) propose architectures based on multimodal large language models or related techniques to tackle multimodal recommendation tasks, but they do not investigate the side adaptation problem.

To the best of our knowledge, efficient and stable deep side adaptation for large pre-trained multimodal embedding models in sequential recommendation remains largely underexplored, which motivates the present study.

6.3 Method

6.3.1 Problem Formulation

Let $\mathscr{U}$ and $\mathscr{V}$ denote the user and item sets. For each user $u \in \mathscr{U}$, we observe a chronological interaction sequence

$$\mathscr{S}_u = (v_1^u, v_2^u, \ldots, v_{T_u}^u), \qquad v_t^u \in \mathscr{V}. \tag{6.1}$$

Each item v is associated with multimodal content

$$\mathbf{x}_v = \{\mathbf{x}_v^{(m)}\}_{m=1}^{M}, \tag{6.2}$$

where M is the number of modalities. Let $\mathscr{B}$ be a frozen large multimodal embedding model with L layers. Given $\mathbf{x}_v$, the backbone produces hidden states

$$\mathbf{h}_v^{(1)}, \mathbf{h}_v^{(2)}, \ldots, \mathbf{h}_v^{(L)}, \qquad \mathbf{h}_v^{(\ell)} \in \mathbb{R}^d. \tag{6.3}$$

We choose a set of adapted layers

$$\mathscr{L} = \{\ell_1 < \ell_2 < \cdots < \ell_K\} \subseteq \{1, \ldots, L\}. \tag{6.4}$$

Because the backbone is frozen, the selected hidden states can be extracted and cached offline. Stresa recursively constructs a task-specific side representation from these cached representations:

$$\mathbf{a}_v^{(0)} = \mathbf{0}, \qquad \mathbf{a}_v^{(k)} = \mathrm{Stresa}_k\left(\mathbf{h}_v^{(\ell_k)}, \mathbf{a}_v^{(k-1)}\right), \quad k = 1, \ldots, K, \tag{6.5}$$

and produces the final adapted embedding

$$\tilde{\mathbf{e}}_v = \mathbf{e}_v^0 + \mathbf{W}_o \mathbf{a}_v^{(K)}, \tag{6.6}$$

where $\mathbf{e}_v^0 \in \mathbb{R}^d$ is the frozen backbone output used by the downstream recommender.

A sequential recommender $\mathscr{F}_\psi$ maps the adapted history $(\tilde{\mathbf{e}}_{v_1^u}, \ldots, \tilde{\mathbf{e}}_{v_t^u})$ to a user state $\mathbf{q}_{u,t}$, and the score for candidate item v is

$$s(u,t,v) = \mathbf{q}_{u,t}^\top \tilde{\mathbf{e}}_v. \tag{6.7}$$

We optimise the standard next-item prediction loss

$$\mathscr{L}_{\mathrm{rec}} = -\sum_{(u,t)} \log \frac{\exp(s(u,t,v_{t+1}^u))}{\sum_{v' \in \mathscr{C}_{u,t}} \exp(s(u,t,v'))}, \tag{6.8}$$

where only the Stresa parameters, the output projection $\mathbf{W}_o$, and the sequence model are trainable. Below, we omit the item index and write $\mathbf{h}_k := \mathbf{h}^{(\ell_k)}$ and $\mathbf{a}_{k-1} := \mathbf{a}^{(k-1)}$ for a single Stresa block.

6.3.2 Overview of Stresa

A standard side adapter typically fuses the current backbone state with the previous side state and then applies a lightweight transform. This works well for shallow adaptation, but under deeper adaptation the side branch is no longer just a temporary local feature: it becomes a

memory that is propagated across layers. Once this happens, two questions become central: how should historical side memory arrive at the current layer, and how should the fused signal be transformed into the next side state while retaining useful historical information? Stresa addresses these two questions explicitly.

At adapted layer k, Stresa first aligns the propagated side memory with the current backbone representation through a stream-aware fusion module, and then converts the fused state into the next propagated side representation:

$$\mathbf{z}_k = \mathrm{SHAF}(\mathbf{h}_k, \mathbf{a}_{k-1}), \qquad \mathbf{a}_k = \mathrm{ReSA}(\mathbf{z}_k). \tag{6.9}$$

Figure 6.2 illustrates the pipeline. SHAF transports the previous side representation to the current depth and fuses it with the current backbone state. ReSA then applies an HC-style residual composition to the fused representation and produces the next side state, which is directly propagated to the following adapted layer. In this sense, Stresa separates *memory transport* from *state refinement*: SHAF determines how the previous side state enters the current layer, while ReSA determines how the fused representation is transformed into the next propagated state.

This separation is the main design principle behind Stresa. The frozen backbone state $\mathbf{h}_k$ serves as the layer-local semantic anchor at depth k, whereas the side branch carries task-specific information propagated from earlier adapted layers. Stresa therefore treats deep adaptation as *progressive state refinement*: each adapted layer first transports and aligns the previous side state with the current frozen representation, and then maps the fused signal into the next side state.

6.3.3 Stream-aware Hidden-Adapter Fusion (SHAF)

SHAF is designed to address two needs in deep side adaptation: (1) stronger fusion with the current hidden state and (2) preservation of historical memory. Plain sigmoid fusion only controls how much previous side memory enters the current layer, but does not model how it should interact with the current hidden representation in a structured way. SHAF therefore introduces stream-aware interaction before fusion, so that the propagated memory can first be aligned with the current hidden state at the stream level before entering the final fusion. This makes the fusion more selective and informative, and also organises the fused representation for the subsequent adapter. At the same time, since the side state is propagated across layers, it should remain an explicit memory path during deep adaptation. SHAF thus adopts an identity-preserving routing operator, which retains a direct carry-over route from the historical memory while allowing controlled cross-stream reorganisation when useful.

We first reshape both signals into s streams of width d_s, where $d = s\,d_s$:

$$\mathbf{H}_k = \mathrm{reshape}(\mathbf{h}_k) \in \mathbb{R}^{s \times d_s}, \qquad \mathbf{A}_{k-1} = \mathrm{reshape}(\mathbf{a}_{k-1}) \in \mathbb{R}^{s \times d_s}. \tag{6.10}$$

This stream decomposition is a lightweight structured coordinate system rather than a semantic assumption about the hidden dimension. It allows SHAF to model cross-subspace interaction through small stream-level operators, and it also gives the following ReSA block the same structured space in which to refine the fused representation.

To decide how historical side memory should interact with the current layer, SHAF computes stream-level routing logits from pooled descriptors of $\mathbf{H}_k$ and $\mathbf{A}_{k-1}$:

$$\mathbf{M}_k(i,j) = \frac{\mathbf{q}_{k,i}^{\top}\mathbf{k}_{k,j}}{\sqrt{d_r}}, \tag{6.11}$$

where $\mathbf{q}_{k,i}$ and $\mathbf{k}_{k,j}$ are learned query/key projections of the corresponding stream summaries. Both projections lie in $\mathbb{R}^{d_r}$, where d_r is the dimensionality of the routing query and key vectors; $\sqrt{d_r}$ scales their dot product. We then apply Sinkhorn normalisation (Adams & Zemel, 2011),

$$\mathbf{R}_k = \mathrm{Sinkhorn}\left(\frac{\mathbf{M}_k}{\tau}\right), \tag{6.12}$$

to obtain a balanced soft routing matrix over streams. This keeps the stream redistribution structured rather than leaving it as an unconstrained dense remapping.

Routing alone, however, would directly replace each stream by a mixture of other streams. For deep propagation, we instead use an identity-preserving routing operator:

$$\beta_k = \sigma(\hat{\beta}_k), \qquad \mathbf{P}_k = (1-\beta_k)\mathbf{I} + \beta_k\mathbf{R}_k. \tag{6.13}$$

This is the central design choice of SHAF. The gate β_k controls how far the transported memory is allowed to deviate from its original stream layout: when β_k is small, the side memory is carried forward mostly along its current stream path; when β_k is larger, related streams can exchange information through $\mathbf{R}_k$. In other words, SHAF treats cross-stream routing as a controlled reorganisation of historical memory rather than a hard replacement.

Finally, SHAF routes the historical side memory and fuses it with the current backbone state:

$$\mathbf{A}_k^{\mathrm{mix}} = \mathbf{P}_k\mathbf{A}_{k-1}, \tag{6.14}$$

$$\lambda_k = \sigma(\hat{\lambda}_k), \qquad \mathbf{Z}_k = (1-\lambda_k)\mathbf{H}_k + \lambda_k\mathbf{A}_k^{\mathrm{mix}}, \tag{6.15}$$

where λ_k controls the overall contribution of historical side memory relative to the current backbone state. The two gates β_k and λ_k therefore play different roles: β_k controls *how historical memory is redistributed across streams*, whereas λ_k controls *how strongly the transported memory enters the current layer*. This separation makes SHAF easier to interpret and also underlies the structural observations in Sec. 6.3.6: SHAF can be viewed as plain sigmoid fusion together with a controlled stream-aware correction.

6.3.4 Residual Stream Adapter (ReSA)

After SHAF, the fused representation $\mathbf{Z}_k$ already lives in stream space. ReSA is designed to refine this structured signal directly, rather than collapsing it back to a single vector and applying a standard bottleneck update. Inspired by Hyper-Connections, ReSA introduces lightweight stream interaction before, after, and along the residual branch of the bottleneck transform.

The adapter first performs *pre-connection mixing*,

$$\mathbf{Z}_k^{\text{pre}} = \mathbf{H}_k^{\text{pre}} \mathbf{Z}_k, \tag{6.16}$$

which lets each stream be transformed in the context of other related streams before entering the bottleneck. It then applies a shared bottleneck MLP,

$$\mathbf{Y}_k = \phi\left(\mathbf{Z}_k^{\text{pre}} \mathbf{W}_{\downarrow}\right) \mathbf{W}_{\uparrow}, \tag{6.17}$$

followed by *post-connection mixing*,

$$\mathbf{Y}_k^{\text{post}} = \mathbf{H}_k^{\text{post}} \mathbf{Y}_k, \tag{6.18}$$

which redistributes the transformed features back across streams. Here $\mathbf{H}_k^{\text{pre}}$ and $\mathbf{H}_k^{\text{post}}$ are learned stream-mixing operators internal to ReSA; they are distinct from the reshaped backbone state $\mathbf{H}_k$ in Eq. (6.10).

Stresa also adds *adaptive residual routing*:

$$\mathbf{Z}_k^{\text{res}} = \mathbf{H}_k^{\text{res}} \mathbf{Z}_k. \tag{6.19}$$

This residual route provides a direct stream-wise shortcut for the fused representation. It complements the nonlinear bottleneck path: the bottleneck increases expressive power, while the residual route preserves a simple path for information that does not need strong transformation.

The final ReSA output is

$$\mathbf{A}_k = \mathbf{Z}_k^{\text{res}} + \mathbf{Y}_k^{\text{post}}. \tag{6.20}$$

Here $\mathbf{A}_k$ denotes the next stream-space side state produced by ReSA. It is directly propagated to the following adapted layer and serves as the historical side state in the next SHAF block. The residual operation is internal to ReSA: $\mathbf{Z}_k^{\text{res}}$ provides the mapped shortcut branch, while $\mathbf{Y}_k^{\text{post}}$ provides the nonlinear refinement branch.

Relative to a standard side adapter, ReSA provides two advantages. First, stream interaction occurs before, after, and along the residual branch of the bottleneck, making the refinement more structured than a single transform on an unstructured vector. Second, the fixed identity shortcut of a conventional residual adapter is replaced by a learnable stream-wise residual mapping. Overall, SHAF and ReSA play complementary roles: SHAF determines how the previous side

state is transported and fused at the current depth, while ReSA maps the fused representation into the next propagated side state. These complementary roles are further clarified by the structural observations in Sec. 6.3.6.

6.3.5 Complexity and Deployment

Compared with a plain side adapter, Stresa introduces additional computation only through lightweight stream-level operators. In SHAF, the extra cost comes from computing stream-level routing logits, applying Sinkhorn normalisation, and multiplying the routed memory by the $s \times s$ operator $\mathbf{P}_k$. In ReSA, the additional structure comes from the stream-mixing operators $\mathbf{H}_k^{\text{pre}}$, $\mathbf{H}_k^{\text{post}}$, and $\mathbf{H}_k^{\text{res}}$, together with the shared bottleneck MLP.

Per adapted layer, the stream operators act on matrices of size $s \times s$ or $s \times d_s$, while the bottleneck operates on width d_s with rank r. Thus, the added overhead scales with the number of streams and the stream width, rather than with a dense full-dimensional $d \times d$ transform. In practice, this overhead is modest because typically $s \ll d$ and $r \ll d_s$.

This additional cost buys two capabilities that a plain side adapter does not model explicitly. SHAF introduces controlled transport of historical side memory before fusion, instead of injecting $\mathbf{a}_{k-1}$ as a single unstructured vector. ReSA then refines the fused representation in the same stream geometry through an explicit stream-wise residual composition within each Stresa block. The gain is therefore not just extra nonlinearity, but a more structured treatment of deep side adaptation.

Stresa also preserves the deployment pattern that matters in recommendation. Since the backbone is frozen, selected hidden states can be extracted and cached offline. Training back-propagates only through the lightweight side branch, the output projection $\mathbf{W}_o$, and the downstream sequential model. After training, the adapted item embeddings $\tilde{\mathbf{e}}_v$ remain item-specific and can be refreshed offline and served in the same way as standard item embeddings. In this sense, Stresa changes the adaptation mechanism without changing the frozen-backbone workflow preferred by large-scale recommendation systems.

6.3.6 Structural Observations on SHAF and ReSA

We provide three algebraic observations that clarify the roles of SHAF and ReSA. They characterise how SHAF differs from plain sigmoid fusion and how ReSA relates to a conventional residual bottleneck adapter. These observations describe the forward mapping of Stresa and are not optimisation or convergence guarantees.

Observation 1 (SHAF as a correction to plain fusion)**.** Let $\mathbf{Z}_k^{\text{plain}} = (1-\lambda_k)\mathbf{H}_k + \lambda_k \mathbf{A}_{k-1}$ denote plain sigmoid fusion. The SHAF representation can be written exactly as

$$\mathbf{Z}_k = \mathbf{Z}_k^{\text{plain}} + \mathbf{C}_k, \qquad \mathbf{C}_k = \lambda_k \beta_k (\mathbf{R}_k - \mathbf{I})\mathbf{A}_{k-1}. \tag{6.21}$$

Observation 1 shows that SHAF contains plain sigmoid fusion as the special case $\beta_k = 0$ or $\mathbf{R}_k = \mathbf{I}$, while otherwise adding a controlled stream-aware correction.

Observation 2 (Common-mode invariance of the SHAF correction)**.** Let $\Pi = \frac{1}{s}\mathbf{1}\mathbf{1}^\top$ be the projector onto the common stream mode. If $\mathbf{R}_k$ is doubly stochastic, then

$$\mathbf{C}_k = \lambda_k \beta_k (\mathbf{R}_k - \mathbf{I})(\mathbf{I} - \Pi)\mathbf{A}_{k-1}, \qquad \Pi \mathbf{C}_k = \mathbf{0}. \tag{6.22}$$

Thus, relative to plain sigmoid fusion, the additional SHAF routing correction leaves the common stream mode unchanged and acts only on variation across streams. With a finite number of Sinkhorn iterations, this statement holds approximately to the extent that $\mathbf{R}_k$ is approximately doubly stochastic.

Observation 3 (ReSA generalises a residual bottleneck update)**.** If $\mathbf{H}_k^{\text{pre}} = \mathbf{H}_k^{\text{post}} = \mathbf{H}_k^{\text{res}} = \mathbf{I}$, then ReSA reduces to

$$\mathbf{A}_k = \mathbf{Z}_k + \phi\left(\mathbf{Z}_k \mathbf{W}_\downarrow\right) \mathbf{W}_\uparrow, \tag{6.23}$$

which is a residual bottleneck update in stream space.

Observation 3 shows that ReSA retains the residual form while generalising its fixed identity shortcut and unstructured feature flow with learnable stream-wise pre-, post-, and residual mappings. Together, Observations 1–3 provide algebraic interpretations of memory transport in SHAF and state refinement in ReSA without requiring additional optimisation claims.

6.4 Experimental Setup

Datasets. We evaluate on three datasets as demonstrated in Table 6.1: MicroLens-50K, MicroLens-100K, and Scientific. MicroLens-50K and MicroLens-100K are public user-level subsets of the MicroLens benchmark for content-driven micro-video recommendation (Y. Ni et al., 2025). Scientific data from the Amazon dataset is included to test on other domains following (Fu, Ge, et al., 2024). For all datasets, we construct chronological user interaction sequences and pair them with the corresponding item-side multimodal content following (Fu, Ge, et al., 2024).

][t]

Table 6.1: Dataset Description.

Dataset	Users	Items	Interaction
MicroLens-100K	100,000	19,738	719,405
MicroLens-50K	50,000	19,099	339,511
Scientific	12,076	20,314	81,711

Implementation Details. Our frozen backbone family is Qwen3-VL-Embedding in two scales, 2B and 8B (M. Li et al., 2026). We strictly follow the experiment setup in (Fu, Ge, et al., 2024). Unless otherwise stated, the backbone is kept frozen, and only cached hidden states are used following (Fu, Ge, et al., 2024). Under this setting, only two key components are trained: the side adapter and the downstream sequential model. The downstream sequential recommender follows (Fu, Ge, et al., 2024; Y. Hou, Mu, et al., 2022; Yuan et al., 2023) and uses two Transformer blocks with two attention heads. Following (Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025), we optimise with AdamW with a weight decay of 0.1. We search the adapter learning rate over $\{1 \times 10^{-5}, 1 \times 10^{-4}, 1 \times 10^{-3}\}$, set it to 1×10^{-4}, and use a batch size of 512. The bottleneck adapter rank was selected from {128, 256, 512, 1024}. User sequences are truncated to length 10.

Evaluation Protocol. We follow the standard leave-one-out evaluation protocol used in prior sequential recommendation work (Fu, Ge, et al., 2024; Fu, Ni, et al., 2025; Kang & McAuley, 2018; Y. Ni et al., 2025). For each user, the last interaction is used for testing and the previous one for validation. We report ranking quality with HR@10, NDCG@10, HR@20, and NDCG@20 (Y. Ni et al., 2025). All methods share the same data splits, downstream sequential model configuration, and full-ranking evaluation over the entire item set.

6.5 Experimental results

In this section, we report our experimental results and address the following research questions:

- **RQ6.1:** Is Stresa better than a standard side adapter?
- **RQ6.2:** How does Stresa compare with state-of-the-art Side adaptation-based multimodal sequential recommendation baselines?
- **RQ6.3:** Which modules contribute to the gains?
- **RQ6.4:** What learning mechanism underlies the difference between Stresa and Side Adapter?

6.5.1 Comparison with Side Adapters (RQ6.1)

RQ1 asks whether Stresa is better than a standard Side Adapter. To answer this question, we compare Stresa against both a conventional Side Adapter (Fu, Ge, et al., 2024; Fu, Ge, Xin, Karatzoglou, et al., 2025; Fu, Ni, et al., 2025) and a Two-Stage baseline (Yuan et al., 2023) that uses only the frozen final-layer backbone embedding. The comparison is conducted on both MicroLens-50K and MicroLens-100K, with two backbone scales, Qwen3-VL-2B and Qwen3-VL-8B, and we report the best test performance under each setting. In addition, to directly verify

Table 6.2: **Main results under frozen-backbone adaptation.** Two-Stage uses only the frozen final-layer backbone embedding. Side Adapter denotes the conventional decoupled side-branch baseline. Stresa consistently improves both regimes across Qwen3-VL-2B and Qwen3-VL-8B backbones and MicroLens-50K/100K. Best results in each backbone block are bold, and second-best results are underlined. * indicates that Stresa significantly outperforms the second-best method at the 0.05 level based on a t-test over five random seeds

Backbone	Method	H@10	N@10	H@20	N@20
MicroLens-50K					
Qwen3-VL-Embedding-8B	Two-Stage	0.0618	0.0335	0.0897	0.0405
	Side Adapter	0.0813	0.0437	0.1169	0.0527
	Stresa (Ours)	**0.0841***	**0.0456***	**0.1212***	**0.0549***
Qwen3-VL-Embedding-2B	Two-Stage	0.0595	0.0323	0.0878	0.0394
	Side Adapter	0.0728	0.0397	0.1042	0.0476
	Stresa (Ours)	**0.0783***	**0.0420***	**0.1145***	**0.0511***
MicroLens-100K					
Qwen3-VL-Embedding-8B	Two-Stage	0.0706	0.0385	0.1006	0.0461
	Side Adapter	0.0977	0.0531	0.1398	0.0637
	Stresa (Ours)	**0.1019***	**0.0559***	**0.1442***	**0.0666***
Qwen3-VL-Embedding-2B	Two-Stage	0.0665	0.0365	0.0981	0.0444
	Side Adapter	0.0895	0.0489	0.1296	0.0590
	Stresa (Ours)	**0.0974***	**0.0541***	**0.1384***	**0.0644***

Table 6.3: **Main results on three datasets.** Best results in each dataset are bold, and second-best results are underlined. * indicates that Stresa significantly outperforms the second-best method according to a t-test ($p < 0.05$) over five seeds.

Method	MicroLens-50K				MicroLens-100K				Scientific			
	H@10	N@10	H@20	N@20	H@10	N@10	H@20	N@20	H@10	N@10	H@20	N@20
IISAN (SIGIR'24)	0.0771	0.0421	0.1121	0.0509	0.0960	0.0526	0.1366	0.0628	0.0683	0.0414	0.0933	0.0470
XSMoE (CIKM'25)	0.0550	0.0296	0.0807	0.0361	0.0697	0.0376	0.1037	0.0461	0.0620	0.0395	0.0840	0.0450
CROSSAN (arXiv'25)	0.0806	0.0431	0.1177	0.0524	0.0999	0.0553	0.1428	0.0661	0.0690	0.0422	0.0928	0.0482
IISAN-Versa (TKDE'25)	0.0784	0.0421	0.1126	0.0507	0.0964	0.0533	0.1341	0.0618	0.0739	0.0447	0.1014	0.0516
Stresa (Ours)	**0.0841***	**0.0456***	**0.1212***	**0.0549***	**0.1019***	**0.0559***	**0.1442***	**0.0666***	**0.0794***	**0.0483***	**0.1067***	**0.0551***

our motivation of adapting more of the backbone's potential hidden representations, we further vary the number of adapted hidden-state layers and compare the resulting trends between Side Adapter and Stresa in Fig. 6.3.

The results in Table 6.2 show that Stresa consistently outperforms both Two-Stage and the standard Side Adapter across all backbone and dataset settings. The gains over Two-Stage confirm the value of adaptation beyond frozen embeddings, while the gains over Side Adapter show that the improvement comes from a better adaptation design rather than simply using side information. More importantly, Fig. 6.3 reveals a clear difference in depth behaviour. Stresa remains stable as adaptation extends to more hidden layers and achieves the best performance when adapting all layers, while its degradation is still limited even when only a small number of layers are adapted. In contrast, the standard Side Adapter reaches its best performance around

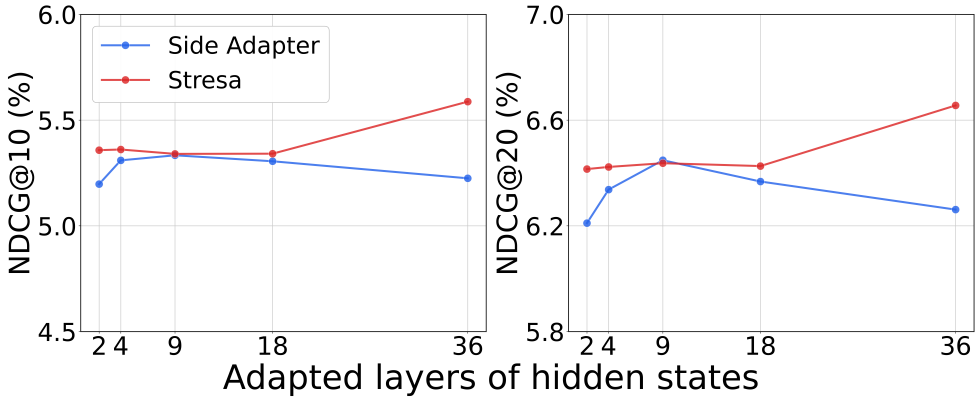


Figure 6.3: NDCG@10 and NDCG@20 on MicroLens-100K (Qwen3-VL-8B) vs. the number of adapted layers of hidden states, for Side Adapter and Stresa. Metrics are taken at the epoch with the best validation HR@10.

Table 6.4: **Ablation on Qwen3-VL-8B.** The progression from a standard side adapter to Stresa.

Method	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-50K				
Side Adapter	0.0813	0.0437	0.1169	0.0527
+ ReSA	0.0829	0.0444	0.1188	0.0534
+ ReSA + SHAF (Stresa)	**0.0841**	**0.0456**	**0.1212**	**0.0549**
MicroLens-100K				
Side Adapter	0.0977	0.0531	0.1398	0.0637
+ ReSA	0.0999	0.0545	0.1417	0.0650
+ ReSA + SHAF (Stresa)	**0.1019**	**0.0559**	**0.1442**	**0.0666**

9 adapted layers, and then declines noticeably as more layers are included. This supports our claim that standard side adaptation does not fully exploit deeper hidden states, whereas Stresa can better utilise the full adaptation space while maintaining stability.

Answer to RQ6.1: Yes. Stresa consistently outperforms the standard Side Adapter across datasets and backbone scales. Its advantage is particularly evident in deeper adaptation settings, where it can stably exploit more hidden states and better unlock the full potential of the backbone.

6.5.2 Comparison with SOTA baselines (RQ6.2)

RQ2 asks how Stresa compares with state-of-the-art side adaptation-based multimodal sequential recommendation baselines. To this end, we compare our approach with IISAN (Fu, Ge, et al., 2024), XSMoE (Y. Qu et al., 2025), IISAN-Versa (Fu, Ge, Xin, Karatzoglou, et al., 2025), and CROSSAN (Fu, Ni, et al., 2025).[1] In these comparisons, Stresa uses Qwen3-VL-8B as

[1]We focus this comparison on the side adaptation paradigm because our goal is to study a fully ID-free setting, where recommendation is driven only by multimodal side information without introducing additional ID signals or auxiliary supervision. Many existing multimodal sequential recommendation methods incorporate extra informa-

Table 6.5: Ablation on Identity Preservation.

Method	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-50K				
w/ Identity Preservation	**0.0841**	**0.0456**	**0.1212**	**0.0549**
w/o Identity Preservation	0.0807	0.0433	0.1184	0.0528
MicroLens-100K				
w/ Identity Preservation	**0.1019**	**0.0559**	**0.1442**	**0.0666**
w/o Identity Preservation	0.0991	0.0545	0.1416	0.0652

Table 6.6: Ablation on stream-aware residual.

Method	HR@10	NDCG@10	HR@20	NDCG@20
MicroLens-50K				
w/o Manifold Constraint	**0.0829**	0.0444	**0.1188**	0.0534
w/ Manifold Constraint	0.0826	**0.0456**	0.1186	**0.0547**
MicroLens-100K				
w/o Manifold Constraint	**0.0999**	0.0545	0.1417	0.0650
w/ Manifold Constraint	0.0996	**0.0546**	**0.1424**	**0.0655**

the backbone. For the competing baselines, we preserve their original architectural settings and report their best results under those settings, because these methods were proposed in multi-tower side-adaptation frameworks, whereas our method is built on a single-tower Qwen3-VL-Embedding backbone. To provide a more direct reference under the same single-tower setting, we also include the Side Adapter baseline in the previous section. For IISAN-Versa, which considers the asymmetric case but still follows a two-tower design, we further evaluate a variant with Qwen3-VL-Embedding-8B on the text side to improve comparability.

Table 6.3 shows that Stresa consistently achieves the best overall performance across all datasets. On MicroLens-50K and MicroLens-100K, it slightly but consistently surpasses the strongest baseline, CROSSAN, on all four metrics. On Scientific, Stresa improves HR@10 from 0.0739 to 0.0794 and NDCG@20 from 0.0516 to 0.0551 over the strongest prior baseline. Overall, these gains are modest but stable across datasets, showing that Stresa generalises well across different scales and domains.

Answer to RQ6.2: Stresa consistently outperforms strong side adaptation-based baselines. This shows that its gains persist not only over a standard Side Adapter, but also over stronger side-adaptation variants, supporting the effectiveness of Stresa across competitive comparison settings.

tion beyond this setting, making direct comparison less fair. Moreover, side adaptation itself has already shown strong performance in prior work, making it a strong and meaningful basis for comparison. Therefore, for fairness and for a cleaner evaluation of the target setting, we mainly compare against representative side adaptation-based methods.

Table 6.7: Efficiency comparison between Stresa and Side Adapter.

Method	Params	Time (s/epoch)	FLOPs/step (10^{12})
Stresa	78.06M	392.41	5.311
Side Adapter	155.73M	338.20	4.395

6.5.3 Ablation Study (RQ6.3)

RQ3 investigates whether Stresa is driven by a single dominant component or by the complementary effects of its modules. To answer this question, we perform comprehensive ablation studies in Tables 6.4, 6.5, and 6.6, examining the contributions of ReSA, SHAF, and additional design choices, including the manifold-constrained residual connection (Xie et al., 2025) in ReSA and identity preservation in SHAF.

The results show a clear progression. In Table 6.4, adding ReSA on top of the Side Adapter consistently improves performance on both datasets, and further adding SHAF (Stresa) yields the best overall results across all metrics. For example, on MicroLens-100K, HR@10/NDCG@10 improve from 0.0977/0.0531 with Side Adapter to 0.0999/0.0545 with ReSA, and further to 0.1019/0.0559 with SHAF; similar gains are observed on HR@20/NDCG@20 and on MicroLens-50K. Table 6.6 further evaluates a ReSA variant with the recently proposed manifold-constrained residual connection (Xie et al., 2025). Although it brings modest gains on several NDCG metrics, the overall performance is comparable while incurring higher computational cost due to manifold operations. We thus do not use it as the default design. Table 6.5 shows that identity preservation is critical, and removing it causes substantial performance drops.

Answer to RQ6.3: The gains come from complementary modules: both ReSA and SHAF are important for achieving them.

6.5.4 Visualisation (RQ6.4)

To answer this question, we analyse layer-wise representation similarity for the frozen backbone, Side Adapter, and Stresa by comparing each layer's cosine similarity to the first and final layers. Figure 6.4 shows that Stresa changes representations more smoothly across depth. Compared to Side Adapter, its similarity to the first layer decreases less abruptly, indicating better preservation of early information, while its similarity to the final layer increases more progressively, indicating stable layer-wise refinement rather than sharp late-stage shifts.

Answer to RQ6.4: The underlying learning mechanism is that Stresa better reconciles identity preservation with progressive adaptation. This leads to a more stable representation trajectory across layers and, consequently, stronger adaptation performance, especially for larger backbones.

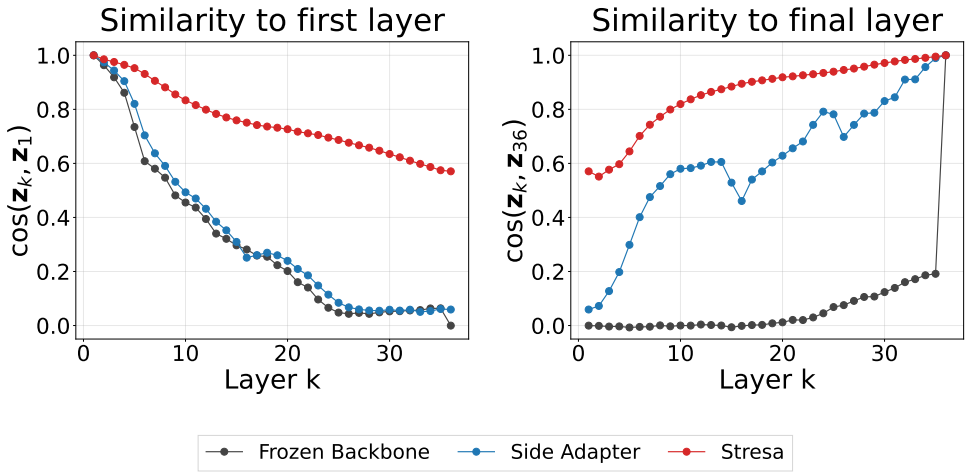


Figure 6.4: Layer-wise cosine similarity on MicroLens-100K for the frozen Qwen3-VL backbone, Side Adapter, and Stresa. Left: $\cos(\mathbf{z}_k, \mathbf{z}_1)$ versus backbone layer index k. Right: $\cos(\mathbf{z}_k, \mathbf{z}_{36})$ versus k. Each series averages over items; layers $k = 1, \ldots, 36$ correspond to the 36 adapter depths.

6.6 Efficiency

Table 6.7 reports the efficiency comparison between Stresa and the standard Side Adapter. Stresa reduces the number of trainable parameters, while introducing a modest increase in computation due to the additional stream-aware routing and mixing operations. In practice, this overhead is not significant: the increases in training time per epoch and FLOPs per step remain moderate, while Stresa achieves stronger recommendation performance in the main experiments.

6.7 Conclusion and Future Work

In this thesis, we study how to make side adaptation for large pre-trained multimodal embedding models more effective under deep adaptation. We propose *Stresa*, a stream-aware side-adaptation framework achieving selective residual update and earlier representation preservation fusion. Across our experiments, Stresa consistently outperforms frozen two-stage adaptation, standard Side Adapter, and strong side adaptation-based baselines. The results further show that its advantage is especially clear when adaptation goes deeper, suggesting that stable memory preservation and stream-aware refinement are key to fully exploiting hidden states in frozen multimodal backbones. Overall, these findings indicate that stream-aware side adaptation is a promising direction for multimodal sequential recommendation.

For future work, it would be valuable to evaluate Stresa on broader multimodal backbones, datasets, and recommendation domains, and to explore more adaptive strategies for selecting or scheduling adapted layers. Another important direction is to connect this framework with more practical deployment settings, such as online or continual adaptation. We hope this work can encourage further research on stable and efficient adaptation for frozen multimodal foundation models in recommendation.

Chapter 7

DIGER: Differentiable Semantic IDs for Generative Recommendation

Publication Note

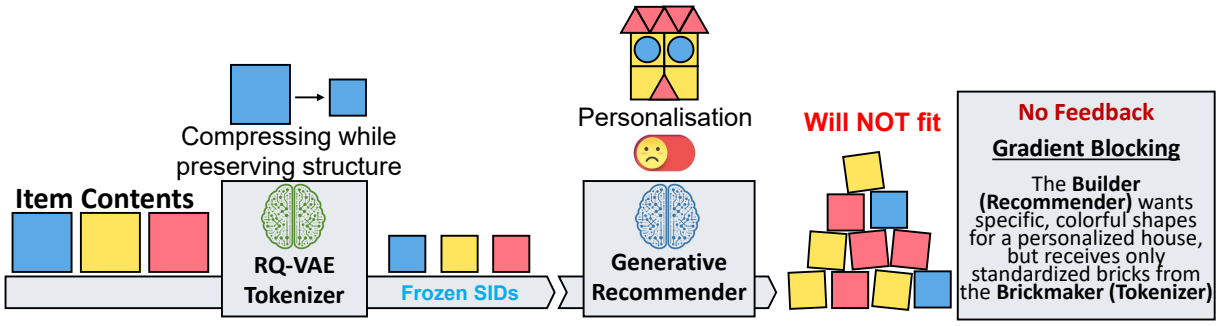
This chapter is adapted from: Junchen Fu, Xuri Ge, Alexandros Karatzoglou, Ioannis Arapakis, Suzan Verberne, Joemon M. Jose, and Zhaochun Ren. Differentiable Semantic ID for Generative Recommendation. SIGIR 2026.

Chapter Positioning

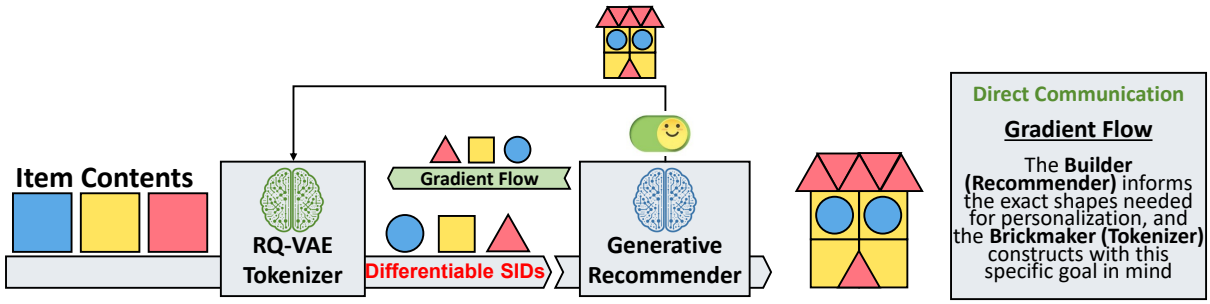
The preceding chapters study continuous representation adaptation of MFMs, jointly training side-adaptation modules and the recommender through real-valued representations. DIGER extends this joint-optimisation objective to semantic-ID-based generative recommendation. It addresses non-differentiable code assignments and codebook collapse, two fundamental obstacles to stable joint training. By tackling these obstacles, DIGER lays a methodological foundation for a unified large recommendation model in which MFM-based content encoding, semantic-ID construction and generative recommendation can ultimately be jointly optimised end to end.

7.1 Introduction

Generative recommendation has introduced a paradigm shift in modern recommender systems (Geng et al., 2022; Rajput et al., 2023; J. Zhao, Wang, Pan, Verberne, & Ren, 2026). Rather than representing items with continuous embeddings alone, recent methods leverage rich item content, most notably textual descriptions, to construct discrete semantic identifiers (SIDs) (Han et al., 2025; Y. Hou, Zhang, et al., 2025; Tay et al., 2022; H. Wang, Guo, et al., 2025) and thereafter formulate recommendation as next-SID generation. This design introduces



(a) Conventional pipeline freezes RQ-VAE semantic IDs, causing gradient blocking from the recommender and objective mismatch.



(b) Our differentiable semantic IDs enable gradient flow for joint optimisation toward recommendation utility.

Figure 7.1: Conventional vs. Generative recommendation with Differentiable SID. A "brickmaker–builder" sketch is used to illustrate the tokeniser and recommender.

desirable properties, such as compact indexing, semantic interpretability, and a natural interface to sequence modelling (M. Hou et al., 2025).

Despite these advantages, current practices still suffer from a mismatch problem. First, a vector-quantised model such as RQ-VAE (D. Lee et al., 2022) is trained with a reconstruction objective to learn a discrete codebook and assign each item a fixed sequence of semantic IDs. Second, given a user's interaction history, a sequence generator (e.g., a Transformer) is trained to predict the next-item semantic IDs (Rajput et al., 2023; Tay et al., 2022). While this pipeline is effective, the indexing loss and the recommendation loss are optimised independently. It introduces a fundamental issue: *semantic indexing is not aligned with the downstream recommendation objective*. This mismatch can degrade recommendation performance, since reconstruction-oriented semantic IDs are not optimised for ranking and remain predefined during user interest evolution (Ye et al., 2025), limiting personalisation and preference-aware representation learning.

Since semantic IDs are precomputed and frozen, the recommendation loss cannot propagate gradients back to the codebook or the ID assignment. In other words, the indexing space is optimised for content reconstruction rather than for next-item prediction. As illustrated in Figure 7.1(a), the tokeniser learns semantic IDs for reconstruction, not personalisation. Because these SIDs are frozen, recommendation gradients cannot reach the tokeniser, so the recom-

mender is stuck with reconstruction-oriented codes that are misaligned with recommendation utility. Consequently, the overall effectiveness of generative recommendations is constrained due to this misalignment.

An intuitive solution to the objective mismatch is to make semantic indexing differentiable, allowing recommendation gradients to jointly optimise the encoder and the generative semantic IDs (Figure 7.1(b)), similar to MoRec (R. Li et al., 2025; Yuan et al., 2023) in traditional modality-based recommendation, where jointly training BERT-based item encoders with recommendation objectives significantly outperforms frozen features. Yet, directly differentiable semantic IDs for semantic indexing remain underexplored (M. Hou et al., 2025). A main challenge in passing gradients through a codebook is its discrete nature: backpropagating through discrete SIDs is inherently non-trivial. The most common workaround — the straight-through estimator (STE) (Bengio et al., 2013) used in RQ-VAE training (D. Lee et al., 2022) — is unstable in practice and can trigger semantic ID collapse (Huh, Cheung, Agrawal, & Isola, 2023). As shown in Figure 7.2, naive differentiable optimisation with STE quickly becomes over-confident in the early stages of training, causing a small subset of codes to dominate, while the rest are rarely selected, i.e., severely reduced code utilisation. This collapse is further observed along with large training-time variance: STE exhibits a markedly larger standard deviation in code balance (error bars) across codebook levels, indicating volatile and inconsistent code usage throughout training. Such an imbalance directly harms recommendation quality (e.g., NDCG@10), making "naively" differentiable semantic indexing brittle and frequently under-performing.

In this work, we propose **DIGER** (Differentiable Semantic ID for GEnerative Recommendation), which enables semantic IDs to be learned jointly with a generative recommender[1]. To enable stable joint training, we design a differentiable SID framework, DRIL, that uses Gumbel noise for exploratory learning (Jang et al., 2016) to address the codebook collapse issues. Motivated by the exploration-exploitation paradigm (Gupta, Smith, & Shalley, 2006), we further propose two uncertainty decay strategies which gradually reduce the stochasticity induced by the injected noise to better align with the training objective. As shown in Figure 7.2, training remains stable, with balanced code usage and steadily improving performance. The injected Gumbel noise promotes early-stage exploration over codes, alleviating collapse and improving codebook utilisation. This design enables stable joint optimisation and consistently improves next-SID generation. Our contributions are as follows:

- We propose DIGER, an effective differentiable semantic indexing framework. To the best of our knowledge, it represents a pioneering effort to effectively enable direct joint optimisation of semantic IDs and generative recommenders.
- To realise DIGER, we introduce a two-part paradigm consisting of Differentiable Semantic ID with Exploratory Learning (DRIL), an exploration–exploitation–inspired frame-

[1]DIGER is metaphorically used as a digger that breaks through and opens a direct gradient pathway between the generative recommender and the semantic ID.

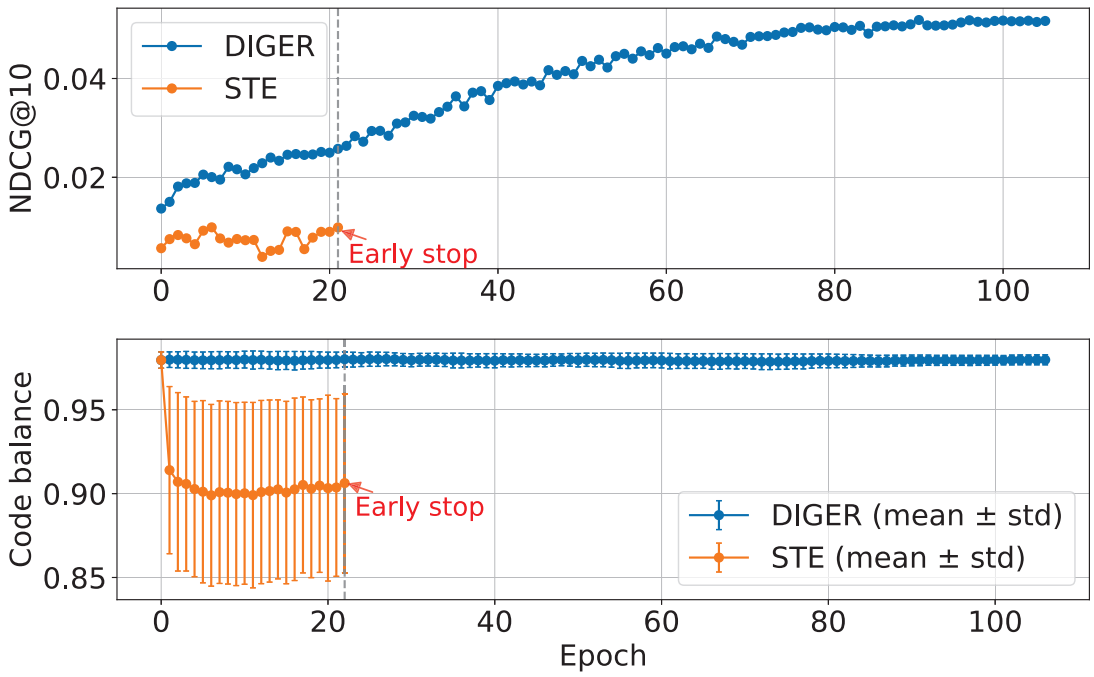


Figure 7.2: Comparison of DIGER and STE on B-Shop Dataset. Top: validation NDCG@10 across epochs. Bottom: code balance summarised by the mean over three codebook levels' coverage, with error bars indicating the standard deviation across levels. Grey dashed vertical line marks the STE early-stop epoch.

work equipped with two Uncertainty Decay (UD) strategies.

- Experiments on three public datasets demonstrate the effectiveness of jointly training with differentiable semantic IDs.

7.2 Related Work

Joint Optimisation for Recommendation. Jointly training both item and user encoders for recommendation has been extensively studied in modality-based sequential recommender systems, motivated by the objective mismatch between pretrained representation encoders and sequential recommendation tasks (Elsayed et al., 2022; Fu, Ge, et al., 2024; Fu, Yuan, et al., 2024; C. Wu et al., 2021a; Yuan et al., 2023). A central theme is to adapt modality encoders jointly with the recommender so that learned representations are optimised for sequential prediction. Representative efforts include joint training of visual backbones, such as ResNet (Elsayed et al., 2022), and fine-tuning language encoders like BERT, to better align textual representations with user–item interactions (C. Wu et al., 2021a). Recent work has extended this paradigm by leveraging

large language models (LLMs) as task backbones or feature generators for recommendation (Fu, Ge, Xin, Karatzoglou, et al., 2025; R. Li et al., 2025; Q. Liu, Chen, et al., 2024). Altogether, these studies leverage joint optimisation to address the representation–task misalignment challenge, by explicitly allowing recommendation gradients to flow into the representation module (e.g., connecting item encoders with user modelling for joint backpropagation), thereby aligning learned features with the downstream objective, while also highlighting practical optimisation challenges when the encoder is large or contains discrete components.

Generative Recommendation. Generative recommender systems formulate recommendation as a sequence generation problem over discrete item identifiers (Han et al., 2025; Y. Hou, Li, et al., 2025; Y. Hou, Zhang, et al., 2025; W. Li et al., 2025; X. Li et al., 2025; X. Lin et al., 2025; C. Liu et al., 2025; Z. Liu et al., 2025; D. Wang et al., 2025; H. Wang, Guo, et al., 2025; Y. Wang et al., 2025; Zhai et al., 2025; L. Zhang et al., 2025; Y. Zhang et al., 2025; J. Zhao et al., 2025; Zheng et al., 2025). To improve generalisation, recent approaches replace random IDs with content-informed, semantic IDs. For example, P5 (Geng et al., 2022) unifies heterogeneous recommendation signals into natural-language sequences and adopts a text-to-text model for recommendation. TIGER (Rajput et al., 2023) introduces a Transformer-based indexing mechanism, where an RQ-VAE codebook assigns each item a sequence of semantic tokens that are subsequently predicted by a decoder. LETTER (W. Wang et al., 2024) further improves semantic ID quality by injecting hierarchical semantics and collaborative signals into residual vector quantisation. However, relatively few studies have investigated the objective mismatch between the semantic ID tokeniser and the downstream generative recommender.

Recently, ETEGRec (E. Liu et al., 2024) proposed a distillation-based approach for aligning tokenisers with generative recommendation models. However, rather than enabling joint optimisation via differentiable semantic IDs, it relies on an alternating distillation scheme to handle the discreteness of semantic IDs, where the recommender and the semantic ID tokeniser are trained by distilling knowledge into each other in separate stages. Since the recommendation loss cannot be directly backpropagated through the discrete indexing process, the indexing module is optimised only through indirect surrogate objectives. Consequently, the core mismatch caused by the blocked gradient flow remains unresolved.

In contrast, our work enables *direct* joint learning for generative recommendation by making semantic IDs differentiable, so that the semantic indexing mechanism and the recommender can be optimised jointly throughout training.

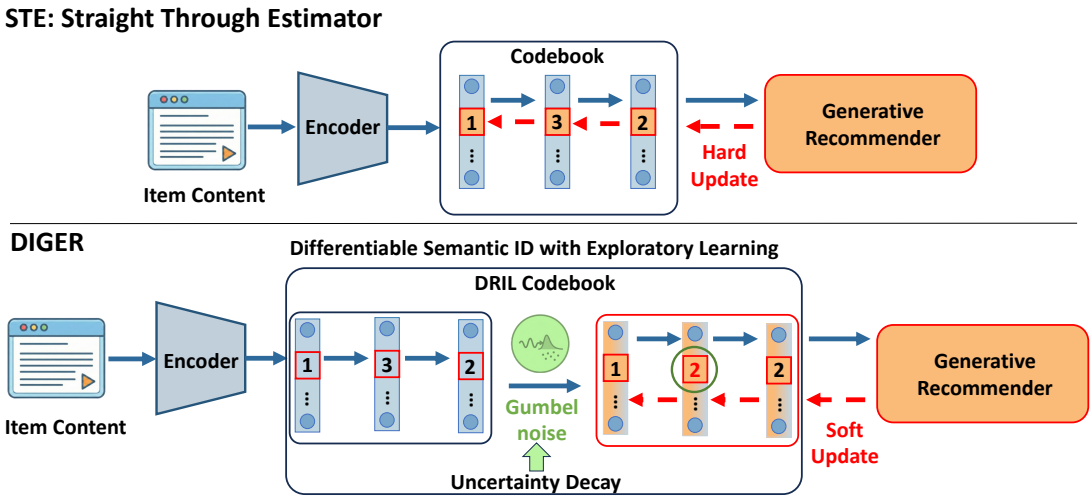


Figure 7.3: Comparison of STE vs. DIGER. **STE:** Relies on deterministic hard selection (e.g., sequence 1-3-2), where gradients only backpropagate to the selected indices (*Hard Update*). **DIGER:** Introduces *Gumbel noise* to encourage exploratory learning. For instance, the noise may shift a deterministic assignment from code 3 to 2 (changing the path from 1-3-2 to 1-2-2), exploring alternative semantic representations. Crucially, DIGER employs *Soft Update*, allowing gradients to flow to all codebook entries, weighted by their Gumbel-Softmax probabilities, while the noise level is progressively reduced via uncertainty decay strategies.

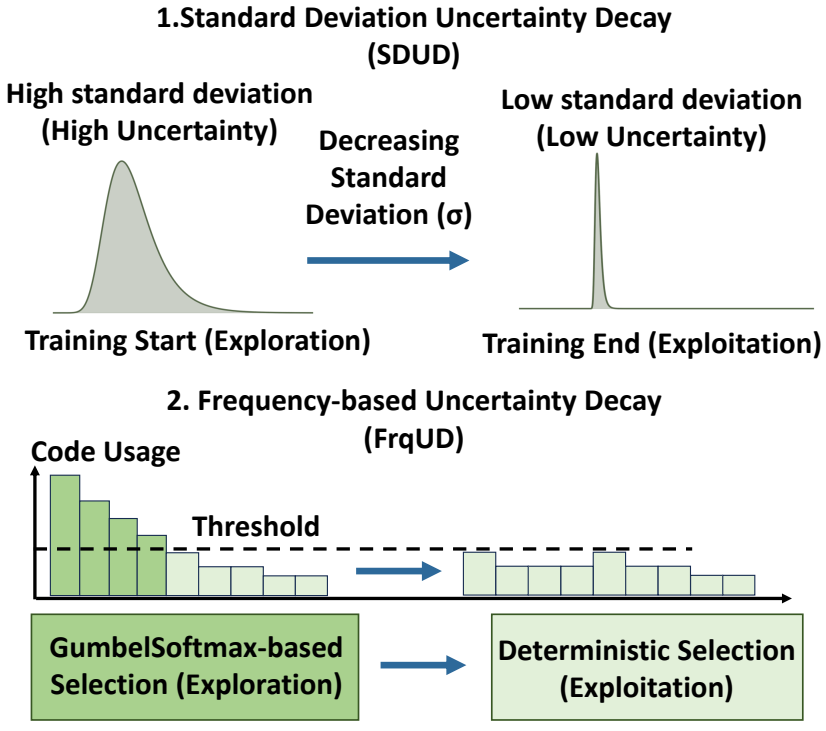


Figure 7.4: Two uncertainty decay strategies for exploration–exploitation: standard deviation–based uncertainty decay (SDUD) and frequency-based uncertainty decay (FrqUD).

7.3 Preliminaries

Following (Rajput et al., 2023), we formulate generative recommendation as an auto-regressive *next semantic ID (SID) prediction problem*, i.e., *given a user's interaction history, a generative model predicts the SID sequence of the next item the user will interact with.*

Let $\mathscr{V}$ denote the set of items. We obtain discrete SIDs for items using an RQ-VAE based tokeniser: given an item $v \in \mathscr{V}$ (e.g., its content features), the tokeniser maps it to a length-m discrete code sequence (its Semantic ID):

$$\mathbf{z}_v = (c_{v,1}, \ldots, c_{v,m}), \quad c_{v,j} \in \{1, \ldots, K\}, \tag{7.1}$$

where K is the SID vocabulary size. Each code $c_{v,j}$ corresponds to an entry in a shared embedding/codebook and is used as the discrete token fed into the generative model.

For a user u, suppose the interaction history up to time t is $\mathbf{h}_u^{(t)} = (v_1, \ldots, v_t)$. We convert the history into a token sequence by concatenating the SIDs of previously interacted items in chronological order:

$$\mathbf{x}_u^{(t)} = \mathbf{z}_{v_1} \oplus \cdots \oplus \mathbf{z}_{v_t}, \tag{7.2}$$

where $\oplus$ denotes sequence concatenation. The generative recommender is a sequence prediction model (e.g., a Transformer decoder) that learns the conditional distribution of the next-item SID: $p_\theta(\mathbf{z}_{v_{t+1}} \mid \mathbf{x}_u^{(t)})$, and produces the SID in an auto-regressive manner:

$$p_\theta(\mathbf{z}_{v_{t+1}} \mid \mathbf{x}_u^{(t)}) = \prod_{j=1}^{m} p_\theta\left(c_{v_{t+1},j} \mid \mathbf{x}_u^{(t)}, c_{v_{t+1},<j}\right). \tag{7.3}$$

where the index $j \in \{1, \ldots, m\}$ denotes the position of a discrete code within the Semantic ID sequence of the predicted item v_{t+1}.

Training minimises the negative log-likelihood of the ground-truth next-item SID:

$$\mathscr{L}_{\text{gen}} = -\sum_u \sum_t \log p_\theta(\mathbf{z}_{v_{t+1}} \mid \mathbf{x}_u^{(t)}). \tag{7.4}$$

At inference time, the model generates the next SID, which realises the *Next SID Prediction* process, and serves as the basis for next-item recommendation. In particular, the generated SID can be deterministically mapped back to an item via the codebook (or a retrieval step over items sharing the same SID), enabling efficient next-item recommendation.

7.4 Our Method: DIGER

Conventional generative recommendation (Rajput et al., 2023) typically adopts a two-stage training paradigm. In contrast, we show that joint optimisation operates over a strictly larger feasible

solution space, while objective mismatch in the two-stage setting can induce arbitrarily large suboptimality even under exact optimisation (Section 7.9, Remark 7.3). To effectively achieve joint optimisation via direct gradient flow, we propose **DIGER** (Differentiable Semantic ID for GEnerative Recommendation). DIGER is trained via a two-part learning scheme inspired by the exploration–exploitation paradigm: **DRIL** (DiffeRentiable Semantic ID with Exploratory Learning) provides Gumbel-noise–based stochastic exploration to learn discrete semantic IDs with direct gradient flow, and **Uncertainty Decay** progressively reduces exploration and steers the model toward stable exploitation. Figure 7.3 illustrates the overall framework.

7.4.1 DRIL: Differentiable Semantic ID with Exploratory Learning

Directly applying the straight-through estimator (STE) yields deterministic hard assignments, which often lead to overconfident early decisions and premature codebook collapse (Figure 7.2), resulting in imbalanced code usage and unstable optimisation. From a distributional viewpoint, such deterministic assignments correspond to highly concentrated, low-entropy selection distributions that limit exploration over the codebook. In Section 7.9, Remark 7.5, we show that increasing assignment entropy increases the effective number of utilised codes, providing a principled motivation for entropy-aware design. Based on this insight, we inject noise into the assignment logits to spread probability mass across multiple codes, and propose a noise-driven assignment mechanism, DRIL, to encourage controlled exploration.

Compared to common Gaussian noise (Z. Lu, Ie, & Sha, 2020), we adopt Gumbel noise Jang et al. (2016), which better captures the probabilistic nature of categorical sampling, inducing a softmax-based selection process in which higher-similarity codes are exponentially more likely to be selected. This aligns more closely with the actual selection process over a codebook, where stronger matches naturally receive higher sampling probabilities. By injecting Gumbel noise into the logits, our method enables stochasticity in code selection, facilitating early-stage exploration while preserving differentiability for end-to-end optimisation.

For an item v, let the (LLM $\to$ RQ-VAE encoder) pipeline output a continuous representation $\mathbf{r}_v$ (or $\mathbf{r}_{v,j}$ for the j-th code position under position-wise quantisation). For each position $j \in \{1,\dots,m\}$, we compute similarity logits to the codebook $\{\mathbf{e}_i\}_{i=1}^{K}$:

$$\ell_{v,j,i} = \mathrm{sim}(\mathbf{r}_{v,j}, \mathbf{e}_i), \quad i \in \{1,\dots,K\}. \tag{7.5}$$

To add the Gumbel noise, we then obtain a Gumbel-Softmax distribution:

$$\tilde{y}_{v,j,i} = \frac{\exp\big((\ell_{v,j,i} + g_{v,j,i})/\tau\big)}{\sum_{k=1}^{K} \exp\big((\ell_{v,j,k} + g_{v,j,k})/\tau\big)}, \tag{7.6}$$

where $g_{v,j,i} \sim \mathrm{Gumbel}(0,1)$ are i.i.d., τ is the temperature.

Hard SID for forward indexing. In the forward pass, we take a hard code by argmax over the

noisy logits (equivalently over $\tilde{\mathbf{y}}_{v,j}$):

$$c_{v,j} = \arg\max_i \left(\ell_{v,j,i} + g_{v,j,i}\right), \tag{7.7}$$

yielding the Semantic ID $\mathbf{z}_v = (c_{v,1}, \ldots, c_{v,m})$, which is used to index the embedding table in the generative recommender (Section 7.3).

Soft update for backpropagation. Gradients are computed through the soft probabilities $\tilde{\mathbf{y}}_{v,j}$. Concretely, we use the soft embedding

$$\bar{\mathbf{e}}_{v,j} = \sum_{i=1}^{K} \tilde{y}_{v,j,i}\, \mathbf{e}_i \tag{7.8}$$

to perform a *soft* update of the RQ-VAE codebook/embedding table, while keeping the hard $c_{v,j}$ to form the discrete SID token sequence. The Gumbel noise encourages exploration and improves codebook utilisation.

7.4.2 Two Uncertainty Decay Strategies

DRIL promotes exploration over discrete codes by sampling diverse code selections during training, but the injected noise perturbs the forward pass and can create a mismatch with inference. Since validation and inference use deterministic semantic ID assignment (for example, a hard argmax over code similarities) for stability and reproducibility, large Gumbel noise in training later can cause the semantic IDs effectively used during training to differ from those used at inference. Therefore, we adopt an exploration-to-exploitation schedule that reduces the uncertainty introduced by Gumbel noise in the later stage of training, better aligning the training objective with the inference-time objective. To this end, we propose two uncertainty decay strategies to reduce the uncertainty[2]:

(1) Standard Deviation Uncertainty Decay (SDUD). Gumbel noise's standard deviation σ controls the stochasticity of SID assignment. A larger σ corresponds to higher uncertainty and encourages exploration, whereas $\sigma \approx 0$ yields near-deterministic assignments that are better aligned with inference-time selection. Therefore, we design an auxiliary objective whose *stationary point* (Graves, 2011) provides an explicit link between the task loss and the optimal noise level, making the training dynamics more interpretable.

[2]In many Gumbel-Softmax applications (e.g., computer vision and speech) (Jang et al., 2016; Takida et al., 2022), the discrete token is not the end goal; it is a means to quantise/compress representations. Accordingly, prior work typically anneals the temperature τ so that assignments become increasingly one-hot. In our setting, however, the semantic ID itself is the target and must be produced as a discrete output at inference, so temperature annealing is less effective. We instead keep τ fixed and reduce the uncertainty toward zero. Empirically, varying τ has little effect on our results.

Specifically, we couple the main generative objective $\mathscr{L}_{\text{gen}}$ with σ as:

$$\mathscr{L}_{\sigma} = \frac{\mathscr{L}_{\text{gen}}}{2(\sigma+\lambda)^2} + \log(\sigma+\lambda), \tag{7.9}$$

where $\mathscr{L}_{\text{gen}}$ is the next-SID prediction loss, $\sigma \geq 0$ is the noise scale (learned through optimisation), and $\lambda > 0$ is a hyperparameter. Importantly, λ not only avoids numerical issues when $\sigma \to 0$, but also determines when σ can effectively reach 0 (i.e., the desired turning point from exploration to exploitation). In practice, we search λ so that $\sigma = 0$ happens near the optimal transition stage.

Taking the partial derivative w.r.t. σ and setting it to zero gives a closed-form equilibrium (we detail the steps in Section 7.10):

$$\frac{\partial \mathscr{L}_{\sigma}}{\partial \sigma} = 0 \quad \Rightarrow \quad (\sigma+\lambda)^2 = \mathscr{L}_{\text{gen}} \quad \Rightarrow \quad \sigma^{\star} = \max\bigl(0, \sqrt{\mathscr{L}_{\text{gen}}} - \lambda\bigr).$$

This demonstrates that as $\mathscr{L}_{\text{gen}}$ decreases during training, the optimal noise scale $\sigma^{\star}$ shrinks accordingly. Furthermore, when $\sqrt{\mathscr{L}_{\text{gen}}} \approx \lambda$, $\sigma^{\star}$ approaches 0, yielding nearly deterministic SID assignment and reducing the train–test mismatch.

(2) Frequency-based Uncertainty Decay (FrqUD). Another uncertainty decay strategy is derived directly from codebook utilisation in the implementation. The key idea is that *high-frequency* codes indicate over-reuse (i.e., some codes are sampled too often), so we apply Gumbel noise to these "hot" codes to encourage exploration and improve code coverage. Conversely, *low-frequency* codes are already under-used, so we keep their sampling stable and close to inference by using the standard deterministic assignment. Concretely, we compute an (EMA-smoothed) usage frequency for each code and threshold it by a ratio of the uniform baseline (Gardner Jr, 2006).

Let $f_i^{(e)}$ denote the usage frequency of code i at epoch e, estimated either empirically within epoch e or via an exponential moving average (EMA) over epochs:

$$f_i^{(e)} \leftarrow \beta f_i^{(e-1)} + (1-\beta)\hat{f}_i^{(e)}, \quad \beta \in [0,1), \tag{7.10}$$

where $\hat{f}_i^{(e)}$ is the raw empirical frequency at epoch e and $\sum_{i=1}^{K} f_i^{(e)} = 1$. Under uniform usage, the average frequency is $\bar{f} = 1/K$. We define the hot-code threshold as:

$$\gamma = r\bar{f} = \frac{r}{K}, \tag{7.11}$$

where $r > 0$ is a tunable *threshold ratio* (e.g., $r = 1.5$). We then identify over-used (hot) codes:

$$\mathscr{I}_{\text{high}}^{(e)} = \{\, i \mid f_i^{(e)} > \gamma \}, \qquad \mathscr{I}_{\text{low}}^{(e)} = \{1,\ldots,K\} \setminus \mathscr{I}_{\text{high}}^{(e)}. \tag{7.12}$$

For $i \in \mathscr{I}^{(e)}_{\text{high}}$, we update assignments with Gumbel noise. For $i \in \mathscr{I}^{(e)}_{\text{low}}$, we disable Gumbel noise, yielding a deterministic distribution:

$$y_i = \begin{cases} \dfrac{\exp((\ell_i + g_i)/\tau)}{\sum_{j=1}^{K} \exp\big((\ell_j + g_j)/\tau\big)}, & i \in \mathscr{I}^{(e)}_{\text{high}}, \\ \dfrac{\exp(\ell_i/\tau)}{\sum_{j=1}^{K} \exp\big(\ell_j/\tau\big)}, & i \in \mathscr{I}^{(e)}_{\text{low}}. \end{cases} \tag{7.13}$$

where ℓ_i denotes the similarity logits produced by the encoder for code i, and τ is the temperature.

Both strategies aim to reduce Gumbel-noise-induced uncertainty later in training while retaining the benefits of exploration earlier on, thereby improving training–inference objective alignment.

7.4.3 Model Training

We train the whole framework with joint optimisation, so gradients flow through semantic ID assignment into the RQ-VAE codebook and the generative recommender. The training is primarily optimised using $\mathscr{L}_{\text{gen}}$, the next-SID autoregressive generation loss (defined in Section 7.3). To prevent the SID representations from drifting too far from the original data distribution, we additionally incorporate $\mathscr{L}_{\text{vq}}$ and $\mathscr{L}_{\text{recon}}$, the standard RQ-VAE losses, implemented following common formulations (D. Lee et al., 2022; Rajput et al., 2023). The main difference is that the

Table 7.1: Dataset statistics.

Dataset	#Users	#Items	#Interactions	Avg. Length	Sparsity
B-Shop	22,363	12,101	198,502	8.88	0.9993
I-Shop	24,772	9,922	206,153	8.32	0.9992
Yelp	30,431	20,033	304,524	10.01	0.9995

RQ-VAE decoder is omitted from our training and inference pipeline: because our RQ-VAE's objective is to produce semantic IDs, we evaluate the reconstruction term directly on the RQ-quantised representations, rather than decoding them back to the original modality. In practice, the model is initialised from a pretrained RQ-VAE, under which both $\mathscr{L}_{\text{vq}}$ and $\mathscr{L}_{\text{recon}}$ remain at a consistently smaller scale during training. As a result, parameter updates are primarily driven by $\mathscr{L}_{\text{gen}}$, while the other terms stabilise optimisation. We therefore optimise the joint objective $\mathscr{L}_{\text{gen}} + \mathscr{L}_{\text{vq}} + \mathscr{L}_{\text{recon}}$, which corresponds to a unified training objective (D. Liang, Krishnan, Hoffman, & Jebara, 2018). Note that although this study addresses the code collapse problem, we intentionally did not incorporate a code-diversity loss (W. Wang et al., 2024). This was done to maintain clarity and to isolate the effect of the differentiable SID alone.

7.5 Experimental Setup

Datasets. We conducted experiments on three public datasets commonly used in generative recommendation research, including two shopping datasets and one food-review dataset. For simplicity, we refer to them as B-Shop, I-Shop, and Yelp. B-Shop contains user interactions with cosmetic products, while I-Shop contains user interactions with music-related products. Dataset statistics are summarised in Table 7.1. The B-Shop and I-Shop datasets consist of product review data accompanied by item descriptions; following common practice, we use the associated product text for semantic encoding. The Yelp dataset contains restaurant reviews, for which the restaurant name and business categories are incorporated as content features. For all three datasets, review texts are not used for item representation and are only employed to construct interaction records. We adopt the data processing pipeline proposed in (W. Wang et al., 2024), which employs an LLaMA-7B model to generate item content from the original titles and descriptions. For each dataset, we treated a user's interaction history as the input sequence and the subsequent interaction as the prediction target. We followed standard preprocessing (Kang & McAuley, 2018): filtering users and items with fewer than five interactions (Hua et al., 2023).

Evaluation. In this thesis, we used Recall@10 and NDCG@10 (Normalised Discounted Cumulative Gain) as evaluation metrics. We followed the widely used leave-one-out evaluation protocol (X. He et al., 2017; E. Liu et al., 2024; W. Wang et al., 2024). All reported results were computed on the test set. Additionally, the predicted item was ranked against the full item set.

Baselines. We first adopt STE (Bengio et al., 2013) as our primary baseline, which represents a naive approach for learning differentiable Semantic IDs and is the default optimisation choice in RQ-VAE-style discrete token learning. In addition, we compare with representative traditional collaborative-filtering and sequential recommendation methods, including MF (Koren et al., 2009), LightGCN (X. He et al., 2020), Caser (J. Tang & Wang, 2018), HGN (C. Ma, Kang, & Liu, 2019), SASRec (Kang & McAuley, 2018), and BERT4Rec (F. Sun et al., 2019). We further include recent generative recommenders, namely BIGRec (Bao et al., 2025), P5 (Geng et al., 2022), LETTER (W. Wang et al., 2024), and ETEGRec (E. Liu et al., 2024). Among them, TIGER (Rajput et al., 2023) serves not only as a strong generative baseline but also as the main reference two-stage framework for our method following the mainstream approaches (E. Liu et al., 2024; W. Wang et al., 2024); All implemented methods were tuned for each dataset using the validation set. For a fair comparison, we report most results from (W. Wang et al., 2024) and ensure that the experimental configurations match so the results are directly comparable.

Implementation Details. We followed the settings in (E. Liu et al., 2024), and initialised the item tokeniser with a pretrained RQ-VAE, set the codebook size to $K = 256$, and used code sequence length $m = 3$ plus one code as the conflict code (Rajput et al., 2023). Our generative recommender adopts a T5-style encoder–decoder backbone with 6 encoder and 6 decoder layers, hidden size 128. We trained the framework with AdamW with weight decay 0.05 and used early stopping based on validation performance. We tuned the Gumbel tem-

perature τ from $\{0.5, 1.0, 2.0, 4.0\}$ and set it to 2.0. We tuned the learning rate for the recommender in $\{0.01, 0.005, 0.001, 0.0005\}$, the learning rate for the tokeniser in $\{10^{-4}, 10^{-5}, 5 \times 10^{-5}, 10^{-6}\}$. We set β to 0.25 in all experiments and found the method to be insensitive to small variations of this value. We tuned the hyperparameters r and λ. Specifically, we perform a fine-grained grid search over λ $\{1.0, 1.2, 1.4, 1.8, 2.0\}$, and tune r over $\{1.0, 1.5, 2.0, 2.5, 3.0\}$. All experiments were conducted on two H100 GPUs.

Table 7.2: Comparison of conventional two-stage generative recommendation and training with differentiable semantic IDs.

Model	B-Shop				I-Shop				Yelp			
	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
Two-Stage	0.0395	0.0610	0.0262	0.0331	0.0870	0.1058	0.0737	0.0797	0.0253	0.0407	0.0164	0.0213
STE	0.0076	0.0134	0.0048	0.0067	0.0450	0.0554	0.0330	0.0360	0.0093	0.0147	0.0060	0.0077
DIGER (FrqUD)	0.0440	0.0683	**0.0294**	0.0372	**0.0915**	**0.1138**	**0.0772**	**0.0844**	0.0266	0.0432	0.0173	0.0227
DIGER (SDUD)	**0.0442**	0.0657	0.0292	0.0361	0.0905	0.1124	0.0753	0.0823	0.0267	**0.0439**	0.0171	0.0227
DIGER (SDUD+FrqUD)	**0.0439**	**0.0696**	0.0293	**0.0376**	0.0907	0.1127	0.0758	0.0829	**0.0273**	0.0437	**0.0175**	**0.0227**

7.6 Experimental results

In this section, we report our experimental results and address the following research questions:

- **RQ7.1:** Does training with differentiable SID improve the performance of generative recommendation models?
- **RQ7.2:** How does the proposed DIGER compare with state-of-the-art methods?
- **RQ7.3:** How do different training strategies and key factors influence the final performance?
- **RQ7.4:** How do semantic ID assignments evolve during training, and how does uncertainty decay affect their stability, consistency, and utilisation?

7.6.1 Comparison with Conventional and Differentiable SID Training (RQ7.1)

To address **RQ7.1**, we compare our differentiable semantic ID training framework DIGER with two representative alternatives: (i) a conventional two-stage generative recommendation pipeline (Two-Stage) (Rajput et al., 2023), and (ii) a naive differentiable SID approach based on STE (Bengio et al., 2013). We report experiments on three benchmarks (B-Shop, I-Shop, and Yelp). In addition, we evaluate several DIGER configurations with different uncertainty-decay strategies. The results are shown in Table 7.2.

Table 7.3: Overall performance comparison between the baselines on generative recommender models on three datasets. The bold results highlight the best performance in comparing the models. R and N stand for Recall and NDCG. We denote "*" as the p-value < 0.05 over the second best baseline.

Model	B-Shop		I-Shop		Yelp	
	R@10	**N@10**	**R@10**	**N@10**	**R@10**	**N@10**
MF	0.0474	0.0191	0.0735	0.0412	0.0381	0.0190
Caser	0.0347	0.0176	0.0710	0.0409	0.0263	0.0134
HGN	0.0512	0.0266	0.1048	0.0774	0.0326	0.0159
BERT4Rec	0.0347	0.0170	0.0822	0.0608	0.0291	0.0159
LightGCN	0.0511	0.0260	0.1000	0.0728	0.0407	0.0207
SASRec	0.0588	0.0313	0.0947	0.0690	0.0296	0.0152
BigRec	0.0299	0.0198	0.0576	0.0491	0.0169	0.0142
P5-SID	0.0584	0.0335	0.0964	0.0730	0.0324	0.0170
P5-CID	0.0597	0.0347	0.0987	0.0751	0.0347	0.0181
LETTER	<u>0.0672</u>	<u>0.0364</u>	<u>0.1122</u>	<u>0.0831</u>	<u>0.0426</u>	**0.0231**
TIGER	0.0610	0.0331	0.1058	0.0797	0.0407	0.0213
ETEGRec	0.0615	0.0335	0.1106	0.0810	0.0415	0.0214
DIGER (Ours)	**0.0683***	**0.0372***	**0.1138***	**0.0844***	**0.0432**	<u>0.0227</u>

Table 7.2 shows that DIGER consistently improves over the conventional two-stage pipeline across all datasets and metrics. For instance, on B-Shop, DIGER raises R@10 from 0.0610 (Two-Stage) to 0.0657–0.0696 (DIGER), and N@10 from 0.0331 to 0.0361–0.0376. Similar trends are observed for I-Shop, where R@10 increases from 0.1058 to 0.1124–0.1138 and N@10 from 0.0797 to 0.0823–0.0844. On Yelp, DIGER also yields consistent gains (e.g., R@10 improves from 0.0407 to 0.0432–0.0439, and N@10 from 0.0213 to 0.0227). In contrast, the naive STE baseline performs poorly on all datasets (e.g., B-Shop R@10 drops to 0.0134), indicating severe optimisation instability; empirically, this aligns with *codebook collapse* behaviour that prevents effective joint learning when discretisation is handled in a purely straight-through manner.

Overall, these results provide empirical support for **RQ7.1**. Compared with a two-stage pipeline, DIGER unlocks additional gains by allowing the model to jointly optimise semantic ID construction and generation objectives. Moreover, the failure of STE highlights that *a naive differentiable SID design is insufficient*; stable joint training requires the proposed DIGER to avoid collapse and to translate differentiability into measurable recommendation improvements.

7.6.2 Comparison with SOTA baselines (RQ7.2)

To address **RQ7.2**, we benchmark DIGER against a broad suite of strong sequential and generative recommenders, including classical collaborative filtering baselines, as well as recent generative recommendation methods. For fair comparison, we report R@10 and N@10 on the same three datasets (Table 7.3). Unless otherwise stated, our main DIGER result corresponds

to the configuration with frequency-based uncertainty decay (**FrqUD**) as it provides strong and stable performance in practice (cf. Table 7.2).

As shown in Table 7.3, DIGER achieves **state-of-the-art or best** performance on B-Shop and I-Shop, outperforming all listed baselines. On Yelp, DIGER remains highly competitive: it achieves the best R@10 (0.0432), while its N@10 (0.0227) is comparable to the strongest baseline LETTER (0.0231). Notably, DIGER also consistently surpasses prior two-stage generative backbones such as TIGER across all datasets. For ETEGRec, its performance does not compare favourably with DIGER. However, it outperforms the conventional approach, TIGER, slightly across all three datasets. This indicates that alternative distillation strategies for objective alignment are effective to some extent; nonetheless, the remaining gap to DIGER is expected, as our method aligns representations through direct gradient-based joint optimisation with differentiable semantic IDs.

Overall, we find that DIGER is competitive with, and often surpasses, state-of-the-art baselines on generative recommendation (**RQ7.2**). The small gap to LETTER on Yelp in N@10 is expected, as LETTER leverages additional collaborative signals (e.g., SASRec-derived embeddings) beyond pure text modelling, whereas DIGER is trained in a *text-only* manner to better align with the core objective of generative recommendation.[3]

7.7 Ablation Study (RQ7.3)

To answer **RQ7.3**, which investigates how different training strategies and key design choices influence the final performance, we conduct a comprehensive ablation study on DIGER. Our analysis focuses on three aspects: (i) the contribution of core components in differentiable semantic ID learning, (ii) the robustness to key uncertainty decay-related hyperparameters, and (iii) the effect of codebook capacity and semantic ID length. Unless otherwise stated, experiments are conducted on the B-Shop dataset, as similar trends are consistently observed on other datasets.

Effect of core components. Table 7.4 reports the impact of removing individual components. Removing the uncertainty decay strategies already causes a clear performance drop, highlighting their crucial role in facilitating the transition from exploration to exploitation during joint optimisation. For the second part, we perform a consistent ablation based on DIGER-w/o UD, which retains Gumbel noise and the soft update. Removing the Gumbel noise causes a significant performance drop, highlighting its importance for effective exploration. The performance remains clearly better than STE, indicating that the soft update is superior to hard updates. Further removing the soft codebook update leads to a slight additional degradation. Furthermore, incorporating the design with temperature annealing yields similar but slightly inferior results,

[3]Incorporating external collaborative embeddings can be viewed as a form of model ensembling or auxiliary information injection; exploring such hybridisation is orthogonal to our focus and is therefore left as future work.

Table 7.4: Ablation of DIGER components on B-Shop Dataset. We omit other datasets due to page limitations.

Model	R@5	R@10	N@5	N@10
Two-Stage	0.0395	0.0610	0.0262	0.0331
Naive E2E (STE)	0.0076	0.0134	0.0048	0.0067
DIGER-w/ UD	**0.0440**	**0.0683**	**0.0294**	**0.0372**
DIGER-w/o UD	0.0424	0.0679	0.0283	0.0365
-w/o Gumbel Noise	0.0168	0.0283	0.0104	0.0141
-w/o soft update	0.0422	0.0650	0.0281	0.0354
-w/ Gumbel tau annealing	0.0419	0.0654	0.0273	0.0348
-w/ Gaussian Noise	0.0389	0.0620	0.0253	0.0327

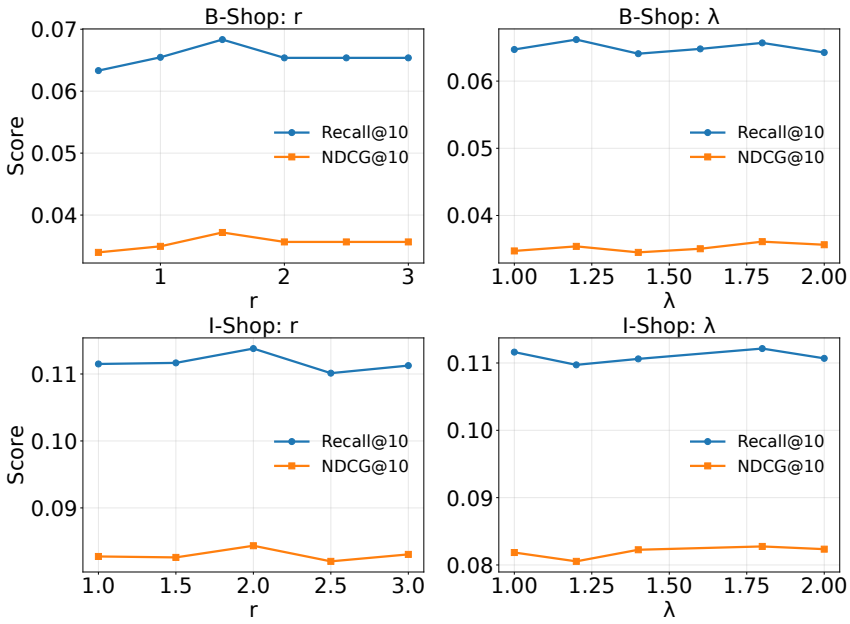


Figure 7.5: Hyperparameter analysis on B-Shop and I-Shop. We vary the hot-threshold r and λ, and report Recall@10 and NDCG@10.

while substituting Gumbel noise with Gaussian noise causes a more pronounced degradation. This suggests that the asymmetric Gumbel distribution, which is designed to approximate categorical sampling by favoring higher-probability codes while preserving exploration, is better suited for discrete semantic ID learning than purely random perturbations.

Sensitivity to Uncertainty Decay hyperparameters. We analysed the sensitivity to λ, which controls when the uncertainty scale σ approaches zero, and r, the hot-code threshold ratio in frequency-based decay. As shown in Figure 7.5, DIGER remains stable across a broad range of values. Very small λ maintains excessive uncertainty late in training, while overly large values suppress early exploration; similarly, moderate r effectively balances exploration and convergence, whereas extreme values weaken one of the two effects. Overall, performance variations remain limited within reasonable ranges, indicating that DIGER does not heavily rely on delicate hyperparameter tuning.

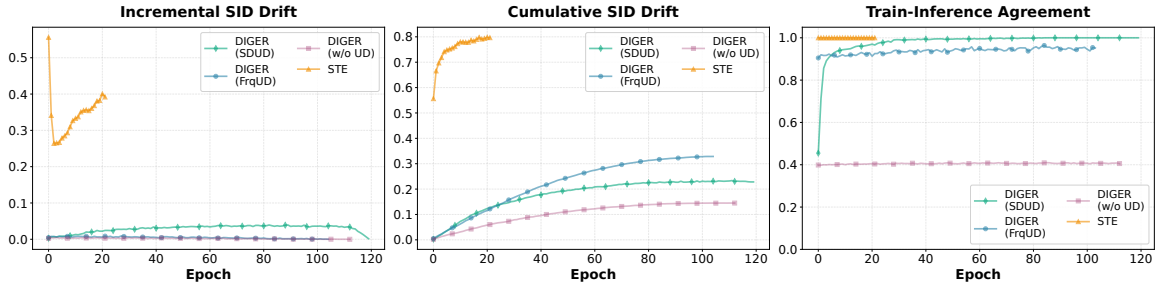


Figure 7.6: Comparison of four semantic ID assignment methods over training epochs: (a) **Incremental SID Drift** measures the fraction of items whose semantic IDs change relative to the previous epoch; (b) **Cumulative SID Drift** measures the fraction of items whose semantic IDs differ from their initial values; (c) **Train–Inference Agreement** measures the fraction of items whose training-time sampled semantic IDs match the inference-time deterministic semantic IDs. Methods: **DIGER (w/o UD)**, **DIGER (FrqUD)**, **DIGER (SDUD)**, and **STE**.

Table 7.5: Effect of codebook capacity (K) and SID length (m) on B-Shop Dataset.

K	m	R@5	R@10	N@5	N@10
128	3	0.0419	0.0655	0.0276	0.0352
256	3	**0.0440**	**0.0683**	**0.0294**	**0.0372**
512	3	0.0434	0.0660	0.0286	0.0359
256	2	0.0362	0.0566	0.0236	0.0301
256	3	**0.0440**	**0.0683**	0.0294	0.0372
256	4	0.0437	0.0673	**0.0297**	**0.0373**

Codebook capacity and semantic ID length. We further studied the effect of codebook size K and semantic ID length m, with results shown in Table 7.5. Performance peaks at $K = 256$, while both smaller and larger codebooks lead to inferior results, reflecting a trade-off between representational capacity and optimisation stability. Reducing m to 2 causes a substantial performance drop, indicating insufficient compositional capacity, whereas increasing m to 4 yields only marginal gains. Overall, $K = 256$ and $m = 3$ provide a favourable balance between expressivity and stable training.

Overall, we find that DIGER's effectiveness stems from the joint design of controlled stochastic code selection, soft codebook updates, and uncertainty decay. The method is robust to hyperparameter choices and exhibits predictable behaviour with respect to codebook capacity and semantic ID length **(RQ7.3)**.

7.8 Semantic ID Dynamics and Stability (RQ7.4)

To answer **RQ7.4**, we analysed how semantic ID (SID) assignments evolve during joint training and how uncertainty decay influences their stability, consistency, and codebook utilisation. We focused on three aspects: SID drift over training, training–inference agreement, and the code usage distribution. Representative results are reported on the B-Shop dataset. Other datasets are

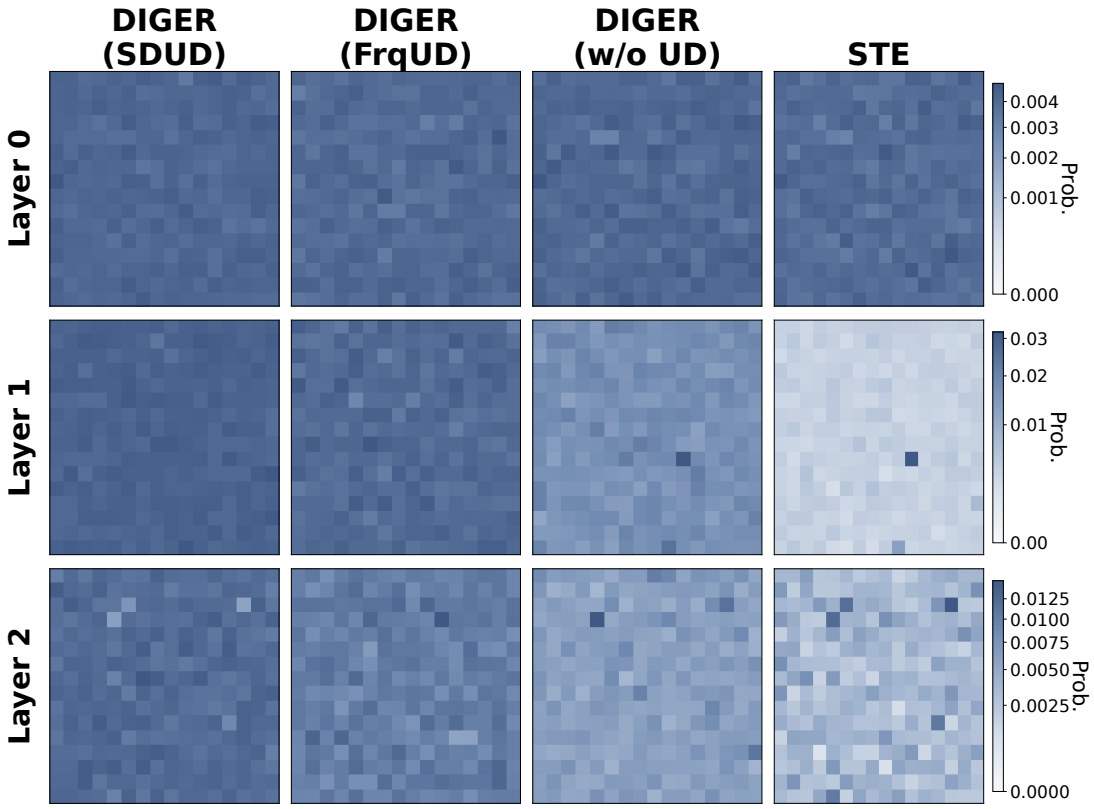


Figure 7.7: Code usage distribution at the best checkpoint for four methods across three quantisation layers. Each 16×16 grid visualises the 256 codebook entries, with colour intensity indicating usage probability.

omitted due to page limit.

SID drift during joint training. To characterise how semantic IDs evolve, we measure both incremental SID drift, which captures epoch-to-epoch assignment changes, and cumulative SID drift, which measures deviations from the initial pretrained assignments. These two metrics reflect different aspects of SID dynamics and are not directly convertible. As shown in Figure 7.6, naive STE exhibits much larger and more abrupt incremental drift than all DIGER variants, particularly in early training, where a large fraction of items change their SIDs within a single epoch. This overly deterministic behaviour leads to unstable optimisation and is consistent with the observed collapse phenomena. In contrast, all DIGER variants evolve more gradually. While DIGER with uncertainty decay, especially FrqUD, shows larger cumulative drift, it remains below 40%, indicating that most semantic IDs are preserved and only selectively refined. DIGER without uncertainty decay exhibits the smallest cumulative drift, below 15%, suggesting limited adaptation. Importantly, variants with uncertainty decay consistently achieve better performance, supporting the benefit of allowing controlled SID changes during later-stage exploitation.

Training–inference agreement. Because Gumbel noise introduces stochasticity during training, we further examine the agreement between training-time sampled semantic IDs and

inference-time deterministic (argmax) semantic IDs. As shown in Figure 7.6, STE achieves perfect agreement by construction, but this rigid determinism coincides with severe instability in semantic ID learning and poor recommendation performance. DIGER without uncertainty decay exhibits persistently low agreement, indicating a large mismatch between stochastic training assignments and deterministic inference behaviour. In contrast, DIGER with uncertainty decay rapidly attains and maintains high agreement after a brief early phase. Notably, FrqUD enforces frequency-aware uncertainty from the beginning by applying stochastic sampling only to overused codes, leading to consistently high agreement throughout training, while SDUD gradually increases agreement by progressively reducing the Gumbel noise's standard deviation σ. Together, these results show that uncertainty decay effectively reconciles stochastic exploration with inference-time determinism, enabling stable and reliable joint optimisation.

Code usage distribution. We further inspect the code usage distribution at the best checkpoint across all quantisation layers, as shown in Figure 7.7. DIGER with uncertainty decay exhibits the most balanced utilisation, where the majority of codes are frequently used across layers. DIGER without uncertainty decay shows moderate imbalance, while STE suffers from severe code collapse, with many codes rarely or never selected, especially in deeper layers. Notably, all methods exhibit relatively uniform usage at the first quantisation layer, but differences become pronounced at higher layers, where uncertainty decay plays a critical role in preventing collapse and encouraging effective use of the codebooks.

Overall, SIDs evolve throughout training, and uncertainty decay makes this evolution stable and effective **(RQ7.4)**. It aligns training and inference assignments, prevents abrupt SID drift and code collapse, and promotes balanced code usage, enabling controlled refinement of semantic IDs with reliable inference behaviour.

7.9 Theoretical Analysis

Theoretical justification. Let θ be the recommender parameters and ϕ be the semantic indexing parameters (codebooks). Let the target (recommendation) objective be $\mathscr{L}_{\text{rec}}(\theta, \phi)$.

Theorem 7.1 (Two-stage freezing as restricted minimisation over ϕ)**.** *Assume the two-stage scheme first selects ϕ from a subset $\mathscr{A} \subseteq \Phi$ (e.g., $\mathscr{A} = \arg\min_\phi \mathscr{L}_{\text{aux}}(\phi)$ or the set of outputs of a pretraining routine), and then optimises θ with ϕ fixed. Define the value function $g(\phi) := \inf_{\theta \in \Theta} \mathscr{L}_{\text{rec}}(\theta, \phi)$. Then the best two-stage performance satisfies $J^\star_{\text{2st}} = \inf_{\phi \in \mathscr{A}} g(\phi)$, $J_{\text{e2e}} = \inf_{\phi \in \Phi} g(\phi)$, and hence $J_{\text{e2e}} \leq J^\star_{\text{2st}}$. Moreover, if g is continuous and $\mathscr{A}$ is compact and does not contain any global minimiser of g over Φ, then $J_{\text{e2e}} < J^\star_{\text{2st}}$.*

Proof. The identities follow from definition of g and the two-stage protocol. Since $\mathscr{A} \subseteq \Phi$, we have $\inf_{\phi \in \Phi} g(\phi) \leq \inf_{\phi \in \mathscr{A}} g(\phi)$. The strict inequality under the stated conditions follows by the extreme value theorem and the assumption that $\mathscr{A}$ excludes all minimisers of g. □

Theorem 7.2 (Arbitrary suboptimality under objective mismatch)**.** *Assume there exists* $\phi^\star \in \arg\min_\phi \min_\theta \mathscr{L}_{\text{rec}}(\theta,\phi)$ *and a (possibly different)* $\phi_{\text{aux}} \in \arg\min_\phi \mathscr{L}_{\text{aux}}(\phi)$ *with* $\phi_{\text{aux}} \neq \phi^\star$*. Then there exist continuous losses* $\mathscr{L}_{\text{rec}}$ *and* $\mathscr{L}_{\text{aux}}$ *such that the optimal two-stage solution is strictly worse than the jointly optimised optimum, and the gap can be made arbitrarily large:*

$$\inf_\theta \mathscr{L}_{\text{rec}}(\theta,\phi_{\text{aux}}) - \min_{\theta,\phi} \mathscr{L}_{\text{rec}}(\theta,\phi) \geq M, \quad \forall M > 0.$$

Proof (constructive). Take $\phi \in \mathbb{R}$ and define $\mathscr{L}_{\text{aux}}(\phi) = (\phi-1)^2$, so $\phi_{\text{aux}} = 1$. Let $\mathscr{L}_{\text{rec}}(\theta,\phi) = (\phi-0)^2 + \theta^2$, so the jointly optimised optimum is attained at $(\theta^\star,\phi^\star) = (0,0)$ with value 0. The best two-stage value (freezing $\phi_{\text{aux}} = 1$) is $\inf_\theta \mathscr{L}_{\text{rec}}(\theta,1) = 1$. Scaling $\mathscr{L}_{\text{rec}}$ by M yields gap M. □

Remark 7.3. By Theorem 7.1, two-stage training optimises the recommendation loss over a restricted subset of indexing parameters. Theorem 7.2 shows that if this subset excludes the joint optimum due to objective mismatch, the resulting suboptimality can be arbitrarily large, even under exact optimisation.

Noise/entropy and codebook utilisation. Let $q \in \Delta^{K-1}$ be the marginal code-usage distribution (e.g., $q_i = \mathbb{E}_v[y_i(v)]$), and define the effective number of used codes as $\text{Eff}(q) := \exp(H(q))$ where $H(q) = -\sum_{i=1}^K q_i \log q_i$.

Theorem 7.4 (Entropy regularisation maximises effective code usage)**.** *For any* $K \geq 2$*,* $\text{Eff}(q)$ *is maximised uniquely at the uniform distribution* $q_i = 1/K$*, achieving* $\text{Eff}(q) = K$*. Equivalently,* $H(q) \leq \log K$ *with equality iff* q *is uniform.*

Proof. This is the standard maximal-entropy property on the probability simplex: $H(q)$ is strictly concave on Δ^{K-1} and attains its maximum at the uniform distribution. □

Remark 7.5. Therefore, adding an entropy bonus that increases the entropy of assignments (or of the induced marginal q) has a direct, formally justified effect: it pushes the solution toward higher $\text{Eff}(q)$, i.e., better codebook utilisation. This is a property of the objective itself, independent of a particular optimiser.

7.10 Derivation of the optimal σ

Let

$$L_\sigma(\sigma) = \frac{L_{\text{gen}}}{2(\sigma+\lambda)^2} + \log(\sigma+\lambda), \tag{7.14}$$

where L_{gen} and λ are treated as constants w.r.t. σ. Differentiating w.r.t. σ yields

$$\frac{\partial L_\sigma}{\partial \sigma} = \frac{L_{\text{gen}}}{2} \cdot (-2)(\sigma+\lambda)^{-3} + (\sigma+\lambda)^{-1} = -\frac{L_{\text{gen}}}{(\sigma+\lambda)^3} + \frac{1}{\sigma+\lambda}. \tag{7.15}$$

Setting $\partial L_\sigma / \partial \sigma = 0$ and multiplying both sides by $(\sigma+\lambda)^3$ gives

$$-(L_{\text{gen}}) + (\sigma+\lambda)^2 = 0 \quad \Rightarrow \quad (\sigma+\lambda)^2 = L_{\text{gen}}. \tag{7.16}$$

The domain of $\log(\sigma+\lambda)$ requires $\sigma+\lambda > 0$, hence we take the positive root:

$$\sigma+\lambda = \sqrt{L_{\text{gen}}} \quad \Rightarrow \quad \sigma^\star = \sqrt{L_{\text{gen}}} - \lambda. \tag{7.17}$$

Finally, if we additionally enforce the practical constraint $\sigma \geq 0$, the closed-form solution becomes

$$\sigma^\star = \max\{0, \sqrt{L_{\text{gen}}} - \lambda\}. \tag{7.18}$$

7.11 Conclusions

This work revisits a key limitation in generative recommendation systems: semantic IDs are typically learned for reconstruction and then frozen, creating a mismatch between semantic indexing and the recommendation objective. Differentiable semantic IDs are therefore required to bridge this gap. However, the naive solution based on straight-through estimators (STE) suffers from severe code collapse issues. To address this issue, DIGER introduces differentiable semantic ID exploratory learning (DRIL), which supports effective SID exploration, together with two uncertainty decay strategies that guide a smooth transition from exploration to exploitation. DIGER stabilises training, prevents semantic ID collapse, improves codebook utilisation, and achieves competitive results on public benchmarks.

Looking ahead, DIGER opens several directions for future work. Differentiable semantic ID can be extended beyond item-side representations, such as learning user-side or interaction-level discrete structures. More generally, exploring alternative estimators and optimisation strategies for discrete latent variables may further improve stability and expressivity. Finally, integrating differentiable semantic IDs with richer collaborative signals or large language model–based recommendation frameworks remains a promising avenue for advancing generative recommendation for jointly training SID and recommender.

II. Recommendation-Oriented Generation Adaptation

The second technical part turns from representation to generation. Here, multimodal foundation models are studied as generators of recommendation-relevant content: missing product modalities for catalogues and micro-video content for user engagement.

This part asks how generated multimodal content should be evaluated when its value is not only visual or textual plausibility, but also its utility within a recommendation ecosystem.

Chapter 8

MMPCBench: Platform-Side Missing Modality Completion for Product Recommendation

Publication Note

This chapter is adapted from: Junchen Fu, Wenhao Deng, Kaiwen Zheng, Ioannis Arapakis, Yu Ye, Yongxin Ni, Joemon M. Jose, and Xuri Ge. Benchmarking Multimodal Large Language Models for Missing Modality Completion in Product Catalogues. Pattern Recognition, 2026.

Chapter Positioning

Part II shifts from adapting representations for recommendation to adapting generative capability for recommendation ecosystems. MMPCBench takes the platform perspective: catalogues often miss images or descriptions, and foundation models can generate the missing modality before downstream recommendation consumes the item representation.

8.1 Introduction

Online marketplaces thrive on rich multimodal product content—photos that convey appearance and text that clarifies specifications, usage, and brand story (H. Li et al., 2020). In practice, however, catalogue entries are frequently incomplete (Malitesta et al., 2024): sellers omit images, provide terse or noisy descriptions, or mislabel key attributes, leaving significant portions of the inventory partially observed.[1] Such gaps propagate downstream, weakening retrieval, matching,

[1] Although platforms have long mandated that sellers provide both product descriptions and images in compliance with strict regulations, some listings on major platforms such as Amazon still face these issues (Y. Zhu et al., 2021). Possible causes include outdated standards, inconsistent practices among third-party sellers, broken images,

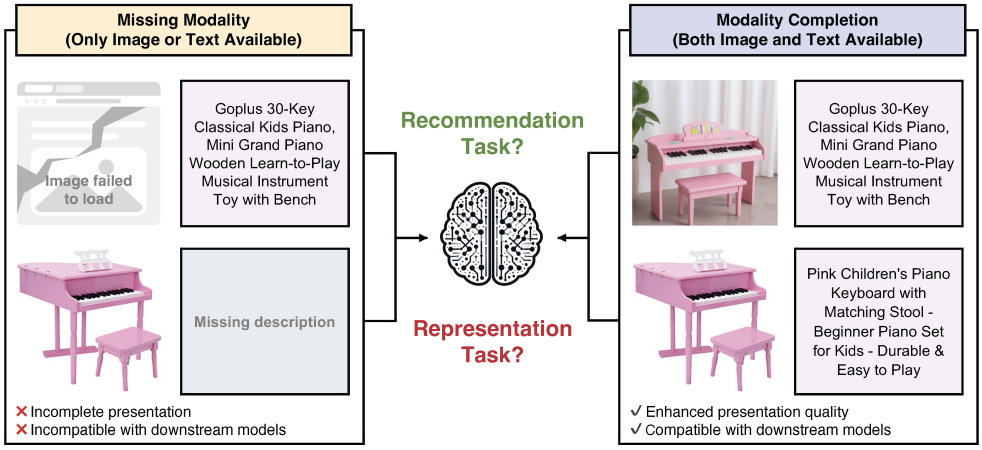


Figure 8.1: Missing-Modality Completion for E-Commerce Products: Given only one modality (either image or text), the task is to generate the missing counterpart. Missing modalities lead to incomplete product presentation, which can negatively impact downstream tasks such as representation learning and recommendation.

and recommendation pipelines that depend on both visual and textual signals to infer user intent and item relevance. In practice, among the various forms of incomplete product information, product images and textual descriptions are the two primary unstructured modalities that support item presentation, multimodal understanding, and downstream recommendation. Accordingly, MMPCBench focuses on missing-modality completion for unstructured content, rather than general completion of all product fields. As recent work in multimodal recommendation and attribute extraction notes, missing or unreliable modalities are widespread and materially degrade system quality if left unaddressed (K. Wang et al., 2023). Therefore, completing the modality information provides a more informative presentation for users and enables greater flexibility for downstream tasks, as illustrated in Figure 8.1.

Classical remedies—manual curation or statistical imputation (e.g., MICE (van Buuren & Groothuis-Oudshoorn, 2011), MissForest (Stekhoven & Bühlmann, 2012))—are effective for low-dimensional tabular fields but fall short when the goal is to reconstruct semantically faithful images or free-form textual descriptions. Most existing approaches to handling missing modalities, especially in domains like e-commerce, operate primarily at the feature level—focusing on imputing embeddings or attributes rather than generating realistic content. Recently, however, the rise of Multimodal Large Language Models (MLLMs) like GPT-4o, Qwen2.5-VL, and Gemma-3—has opened up promising new possibilities. In fields like medical imaging and scientific visualisation, these models have already demonstrated impressive performance in reconstructing missing modalities from available inputs (Ke et al., 2025). Yet in the context of e-commerce, the dominant paradigm remains feature imputation, with limited exploration of MLLM-based generation.

corrupted text, or other technical factors. As a result, this remains an active area of research, particularly in the field of recommendation (Fu, Ge, Xin, Karatzoglou, et al., 2025; Malitesta et al., 2024; C. Wang et al., 2018).

To this end, we present the **Missing Modality Product Completion Benchmark (MMPCBench)**, comprising the *Content Quality Completion Benchmark (CQBench)* and the *Recommendation Benchmark (RecBench)*. A key novelty of MMPCBench is that it jointly evaluates both the content quality of completed modalities and their downstream utility in recommendation. Six state-of-the-art MLLMs from the Qwen2.5-VL and Gemma-3 families are evaluated on *Image-to-Text* (I→T) and *Text-to-Image* (T→I) completion tasks. Experiments reveal that performance varies across categories and model scales in a non-monotonic manner, exhibits *task-wise* asymmetries between I→T and T→I, and shows only a weak correlation between content quality and recommendation performance. To address misalignment of the existing MLLMs with e-commerce product completion, we explore **Group Relative Policy Optimisation (GRPO)**, which improves I→T but remains limited on T→I.

Our contributions are:

- We introduce **MMPCBench**, the first benchmark for missing-modality product completion in e-commerce, comprising two sub-benchmarks: the *CQBench* and the *RecBench*, covering both I→T and T→I tasks across diverse product categories.
- We conduct a systematic evaluation of six state-of-the-art MLLMs, spanning three sizes each from the Qwen2.5-VL and Gemma-3 families, under a unified protocol for cross-modal completion.
- We explore **Group Relative Policy Optimisation (GRPO)** for multimodal completion to enhance alignment of MLLMs in missing-modality completion.

8.2 Related Work

Missing Modality Completion in Multimodal Learning Multimodal learning often encounters scenarios where one or more data modalities are unavailable due to sensor failures, cost constraints, or data collection issues (M. Ma et al., 2021). This challenge has led to increasing interest in missing modality completion, which aims to enhance model robustness in the presence of incomplete modality inputs.

Early efforts approached this as a data imputation problem, relying on manual curation or classical statistical techniques such as MICE (van Buuren & Groothuis-Oudshoorn, 2011) and MissForest (Stekhoven & Bühlmann, 2012). Neural network-based methods, including generative approaches like multimodal autoencoders and GANs, have also been proposed (M. Ma et al., 2021). Another line of studies moved away from explicit imputation, instead designing architectures that are inherently resilient to modality dropouts (M. Ma, Ren, Zhao, Testuggine, & Peng, 2022). For example, Y.-L. Lee, Tsai, Chiu, and Lee (2023) proposed multimodal prompting to compensate for missing modalities, while Z. Wang, Wan, and Wan (2020) introduced a fusion mechanism to complete missing modalities for sentiment analysis tasks. Z. Chen et al.

(2025) addresses the missing modality problem in multimodal Alzheimer's disease biomarker detection. MCE (B. Zhao et al., 2025) handles missing modalities under imbalanced missing rates. MIMAR-OSA (Qiu et al., 2025) addresses obstructive sleep apnea diagnosis using multimodal data with missing modality reconstruction. However, these methods typically require training on complete multimodal data. Recent advances leverage large multimodal language models (MLLMs) to perform zero-shot missing modality completion in tasks such as visual question answering and healthcare (Ke et al., 2025), achieving state-of-the-art performance. These encouraging results motivate us to explore the application of MLLMs for missing modality completion in the product domain.

Product Missing Modality Completion In e-commerce and product-centric applications, a persistent issue is the incompleteness of multimodal product data—such as missing images, incomplete product descriptions, or absent attribute tags. This can result from cataloging errors, crowdsourcing inconsistencies, or metadata loss during system migrations. C. Wang et al. (2018) pioneered latent regression-based methods to estimate missing product content, followed by adaptive feature sampling (Shi et al., 2019) for handling incomplete product features. More recently, F. Liu et al. (2022) and Malitesta et al. (2024) propose graph-based feature propagation approaches to recover missing attributes or modality features for products in large-scale catalogues. These approaches leverage product–product similarity or co-purchase relations to impute realistic content. Malitesta et al. (2024) shows that dropping products with missing modalities is wasteful, and proposes modular imputation pipelines applicable before model training. J. Kim et al. (2025) advance this line by developing DGMRec, a generative model that synthesises missing modalities using disentangled latent representations of available product features, enabling cold-start and zero-modality product recovery. However, most existing approaches are all designed specifically for recommendation downstream tasks, requiring task-specific training or fine-tuning, which limits their applicability to missing modality completion in other scenarios, such as presentation or alternative tasks.

Despite the growing interest in product modality completion and the promising potential of using Multimodal Large Language Models (MLLMs) for this task, no prior work has systematically explored leveraging advanced MLLMs—such as Qwen-VL and Gemma-3—for completing missing modalities in product data. MLLMs are inherently capable of integrating textual and visual contexts and generating missing components. However, their potential for product-level modality generation, especially in large-scale, real-world scenarios such as e-commerce catalogues, remains largely underexplored. This gap motivates our study, which presents the first comprehensive benchmark for the product missing modality completion task. Recent work has further underscored the need for comprehensive and reliable multimodal benchmarks. This survey provides a systematic overview of the evaluation landscape for large multimodal models (Z. Zhang, Wang, Wen, et al., 2025). This benchmark-oriented research further outlines key

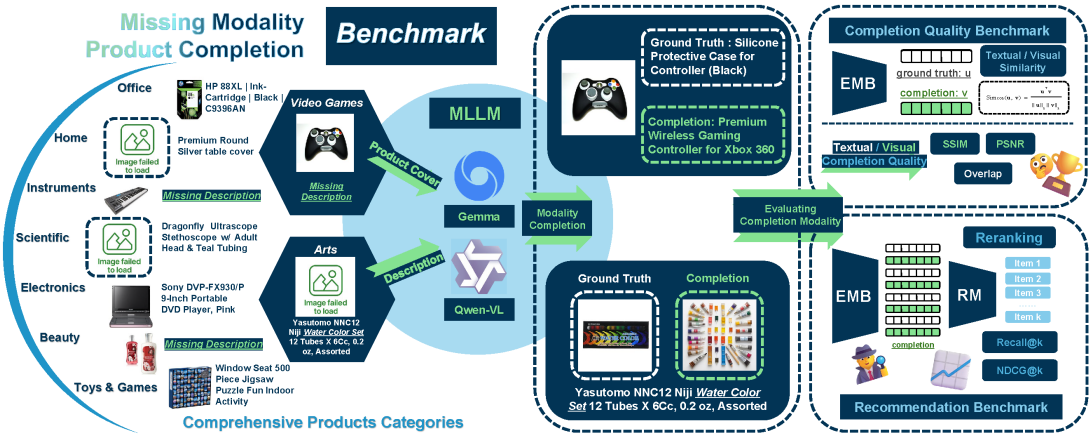


Figure 8.2: Overview of MMPCBench. MMPCBench is a benchmark designed to evaluate MLLMs on the task of completing missing product modalities—either text or image—across diverse product categories. EMB denotes the embedding models used for similarity-based evaluation, while RM refers to the recommender models employed for downstream recommendation tasks. The generated modalities are evaluated using these two components of the benchmark. Notably, completing missing images requires a diffusion model to visualise the image prompts generated by the MLLMs.

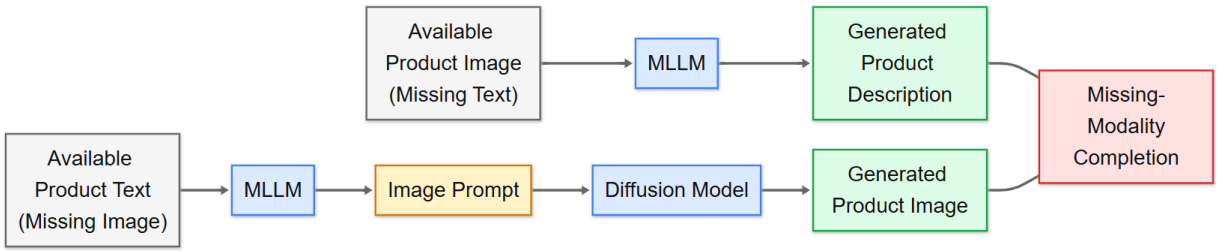


Figure 8.3: Overview of the missing-modality completion framework for the Image→Text and Text→Image tasks.

principles for trustworthy AI evaluation (Z. Zhang, Wang, Guo, et al., 2025). Related work also highlights the importance of versatile multimedia quality assessment frameworks for visual communication settings (Z. Zhang et al., 2026).

8.3 Methodology

8.3.1 Problem Formulation

To identify the most critical elements for presentation and downstream tasks, we limit our study to the text description and product cover image, as they constitute the most essential components for on-site demonstration and downstream applications such as multimodal recommenda-

tion (X. Zhou, 2023).[2]

We formulate the missing modality completion task using Multimodal Large Language Models (MLLMs) in an e-commerce context, where each product is represented as a multimodal pair (x^T, x^I), consisting of a textual description x^T and a product image x^I. However, in real-world scenarios, one modality may be missing due to annotation gaps or system constraints (Malitesta et al., 2024). To address this, we define two complementary tasks: **Text-to-Image Completion**, which aims to generate an image $\hat{x}^I$ from the available text x^T. In this task, we leverage the product description as input to the MLLM to produce an image prompt, which is then passed to a Text-to-Image diffusion model to synthesise the image.

We consider two complementary tasks. In Text→Image, the MLLM generates a visually grounded prompt from x^T, and a frozen diffusion model renders $\hat{x}^I$. In Image→Text, the MLLM directly generates a textual description $\hat{x}^T$ from the product image x^I. Figure 8.3 illustrates both pipelines. Figure 8.3 illustrates the pipelines of the two missing-modality completion tasks, namely Image-to-Text and Text-to-Image.

Table 8.1: Recommendation Benchmark Category Description. For the Content Quality Benchmark, we sampled 1,000 items per category.

Category Dataset	User	Item	Interactions
All Beauty	1,068	4,094	10,198
Arts, Crafts & Sewing	8,692	4,250	55,714
Electronics	7,191	3,056	91,087
Home & Kitchen	6,896	3,163	68,085
Industrial & Scientific	5,437	4,034	33,026
Musical Instruments	12,890	3,640	91,603
Office Products	14,167	4,275	90,011
Toys & Games	12,919	4,542	81,937
Video Games	25,501	3,461	189,692

To evaluate the effectiveness of modality completion, we adopt a dual-perspective framework. First, we assess *presentation-level fidelity*, which measures the semantic alignment and perceptual quality of the generated modality. Second, we examine the *recommendation-level impact*, which evaluates whether the generated modality can serve as an effective substitute for the missing input in downstream tasks such as personalised ranking. Specifically, we compare the recommendation performance using the generated modality against that of the original modality. This twofold evaluation ensures both the intrinsic plausibility and the extrinsic utility of the generated content. The detailed evaluation procedure is described in the following section.

[2]Other attributes such as brand, price, and specifications are excluded because they are typically schema-constrained and verifiable fields, whose completion is better framed as database imputation or knowledge-based completion rather than open-ended multimodal generation. We therefore leave structured-attribute completion to future work.

8.3.2 MMPCBench

To systematically evaluate the capability of MLLMs to complete missing modalities for product data, we introduce MMPCBench, a problem-oriented benchmark specifically designed for missing-modality completion in product catalogues, as illustrated in Figure 8.2. MMPCBench covers a comprehensive range of product categories to enable detailed and thorough evaluation. We evaluate three scales of open-source MLLMs from the state-of-the-art Gemma-3 and Qwen2.5-VL families. As defined in the Problem Formulation Section, our evaluation is twofold: completion performance is assessed with two benchmarks, namely the *Completion Quality Benchmark* and the *Recommendation Benchmark*.

Dataset MMPCBench is built upon a recent publicly available Amazon product dataset. We chose this dataset because Amazon is one of the largest real-world e-commerce platforms, and its scale and recency make it a realistic foundation for studying missing-modality issues in product catalogues. To mitigate potential data leakage—since many MLLMs may have been pre-trained on earlier versions of the Amazon Review dataset—we use the most recent version, the Amazon Review Dataset[3], released in March 2024. Following prior works (Y. Hou, Mu, et al., 2022; S. Zhang et al., 2025), we focus on nine widely used product categories for multimodal recommendation: *"All Beauty"*, *"Arts, Crafts & Sewing"*, *"Electronics"*, *"Home & Kitchen"*, *"Industrial & Scientific"*, *"Musical Instruments"*, *"Office Products"*, *"Toys & Games"*, and *"Video Games"*. As detailed in Table 8.1, we follow the five-core filtering procedure from (X. Zhou, 2023), retaining only items with both textual description and product cover. For presentation benchmarking, we further sample 1,000 items from each category.

Evaluation MMPCBench is organised into two evaluation components: *Content Quality Benchmark* and *Recommendation Benchmark*, which are designed to assess different aspects of modality completion performance in e-commerce scenarios. Each subcategory encompasses both *Image-to-Text* and *Text-to-Image* tasks.
In the *Completion Quality Benchmark* (CQBench), for each sample, we mask either the product image or the textual description and use a Multimodal Large Language Model (MLLM), such as Qwen2.5-VL or Gemma-3, to complete the missing modality. For Text-to-Image tasks, the MLLMs generate a product-cover prompt, which is then fed into a diffusion-based model to produce the final images following (Ke et al., 2025). The quality of the generated modality is evaluated using similarity measures and standard automatic metrics. For each reported value, we present the mean along with its standard deviation.

For text completion, we evaluate: **Cosine Similarity** – cosine of the angle between TF-IDF vectors of original and generated texts; **Euclidean Similarity** — L2 distance on TF–IDF; add 1 and take the reciprocal for similarity; **Token Overlap** – proportion of shared tokens,

[3] https://amazon-reviews-2023.github.io/

reflecting lexical similarity; **BERTScore** (T. Zhang et al., 2019) – semantic similarity based on contextual BERT embeddings. **SBERT Similarity (Reimers & Gurevych, 2019)**: cosine similarity between Sentence-BERT embeddings of generated and reference descriptions, measuring sentence-level semantic alignment. **METEOR (Banerjee & Lavie, 2005)**: a recall-oriented metric with stemming and synonym matching, making it robust to paraphrasing. **MoverScore (W. Zhao et al., 2019)**: an optimal-transport-based metric that measures semantic similarity by aligning generated and reference text.

For image completion, we assess: **PSNR** (Hore & Ziou, 2010) – pixel-level reconstruction quality; **SSIM** (Z. Wang et al., 2004) – perceived image quality via luminance, contrast, and structure; **MSE** (Tan, Li, Tan, Rahardja, & Yeo, 2013) – mean squared pixel-wise difference; **LPIPS** (R. Zhang et al., 2018) – perceptual similarity using deep features; **CLIP Similarity** (Radford, Kim, Hallacy, et al., 2021) – cosine similarity of CLIP embeddings for semantic alignment; **FID** measures the feature-space distance between generated and real images, where lower is better (Heusel et al., 2017).

To complement automatic metrics, we conduct a blind human evaluation on a 1–5 Likert scale. We randomly sample 54 products from CQBench with balanced category coverage (6 items from each of 9 categories) using a fixed random seed. Because Image→Text and Text→Image generate outputs in different modalities, we adopt task-specific criteria following prior work (Van der Lee et al., 2021): for Image→Text, *Factual Accuracy*, *Completeness*, and *Marketing Appeal*; for Text→Image, *Visual Plausibility*, *Attribute Fidelity*, and *Commercial Readiness*. Although different in wording, these dimensions reflect the same underlying aspects of intrinsic quality, faithfulness to the input, and practical utility. For each sampled product, annotators compare outputs from all six models side by side, with model identities hidden as *Model A–F* in randomised order. For Image→Text, they see the product image and generated text; for Text→Image, they see the source text and generated image. Ground-truth outputs are not shown, so ratings reflect absolute generation quality. Five annotators independently score each output on each dimension.

In the *Recommendation Benchmark*, we evaluate the effectiveness of the generated modality as a substitute in multimodal recommendation tasks. Specifically, we test whether replacing the original modality with the generated one affects recommendation performance, using three classic and popular multimodal recommendation models: **VBPR** (R. He & McAuley, 2016): A shallow model that extends matrix factorisation by incorporating multimodal features extracted from product images and textual descriptions. **BM3** (X. Zhou et al., 2023): A self-supervised learning framework designed for multimodal recommendation. **FREEDOM** (X. Zhou & Shen, 2023): A state-of-the-art multimodal recommender that simultaneously freezes the item-item graph and denoises the user-item interaction graph to enhance recommendation performance. RecBench is designed to evaluate whether completed modalities are *usable substitutes* for downstream recommendation, i.e., whether missing-modality completion preserves enough semantic signal for

ranking. It is therefore primarily a downstream sanity check for robustness rather than a highly sensitive leaderboard for fine-grained completion quality.

We evaluate the performance difference of using the completed modality with top-k metrics: **Recall@k** – proportion of relevant items in the top-k list, and **NDCG@k** – relevance-aware metric giving higher scores to relevant items ranked earlier. We report results for $k = \{5, 10, 20, 50\}$ to evaluate performance across different recommendation list lengths.

Overall, MMPCBench provides a unified and practical testbed to evaluate both the fidelity of modality completion and its downstream impact on recommendation quality in realistic e-commerce scenarios.

8.3.3 GRPO Training

To further improve the generation quality of MLLMs under missing modality conditions, we explore a reinforcement learning approach. Specifically, we utilise Group Relative Policy Optimisation (GRPO) (Shao et al., 2024), aiming to better align existing MLLMs with the modality completion tasks. We chose GRPO for its training efficiency and its strong alignment with our rule-based reward function, which measures the similarity between the generated content and the original data. We apply this approach to our defined two tasks from the Problem Formulation: **Image-to-Text** and **Text-to-Image** generation.

For each task, we design task-specific reward functions to measure the similarity between the generated and target modality outputs, thereby providing effective learning signals. In the Image-to-Text task, the MLLM completes textual descriptions from the product cover. To evaluate semantic fidelity, we compute the reward as the cosine similarity between the generated and ground-truth texts using embeddings from the "`sentence-transformers/all-MiniLM-L6-v2`" Embedding model following (Ghalandari, Hokamp, & Ifrim, 2022). In the Text-to-Image task, we employ the advanced and efficient Text-to-Image diffusion model "`stabilityai/stable-diffusion-3.5-large-turbo`" from the Hugging Face platform[4] to synthesise images based on the MLLM-completed prompt for product cover generation. The reward is then calculated using a CLIP-based (Vaze, Carion, & Misra, 2023) similarity model to assess the alignment between the generated image and the intended content. For the training dataset, we sample a total of 500 items per category, resulting in 4,500 images across nine categories. To prevent data leakage, all sampled items are selected from outside the benchmark dataset; moreover, although some GRPO reward components partially overlap with automatic evaluation metrics, all reported results are evaluated on unseen CQBench test instances under strict train–test separation.

[4]https://huggingface.co/

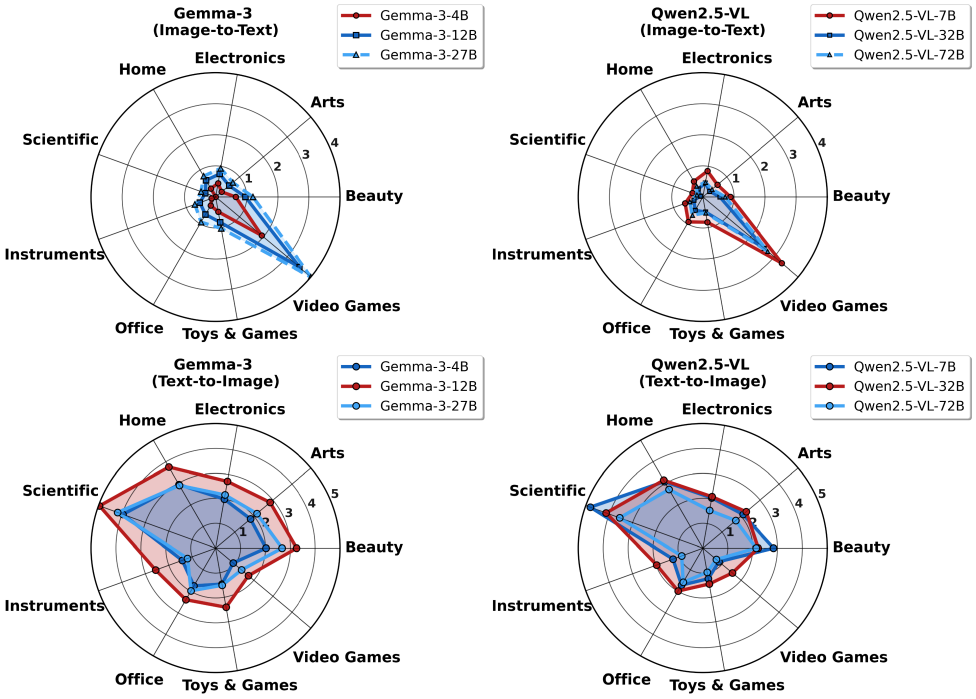


Figure 8.4: Radar Charts for Both T→I and I→T Tasks on Content Quality Benchmark. Each metric is normalised to a [0, 1] range and then summed, resulting in a maximum score of 4 for I→T and 5 for T→I tasks.

8.4 Experiments

8.4.1 Implementation Details

We employ the MLLMs `Qwen/Qwen2.5-VL-7B-Instruct`, `Qwen/Qwen2.5-VL-32B-Instruct`, `Qwen/Qwen2.5-VL-72B-Instruct`, `google/gemma-3-4b-it`, `google/gemma-3-12b-it`, and `google/gemma-3-27b-it` from the Hugging Face platform[5] with 4-bit quantisation for inference, and adopt `stabilityai/stable-diffusion-3.5-large-turbo` for efficient, high-quality image generation using the default 4-step generation setting. For GRPO fine-tuning, we utilise the Transformer Reinforcement Learning (TRL)[6] training framework following the GAD-cell/vlm-grpo implementation[7], set the LoRA rank to 16 and the learning rate in $\{1 \times 10^{-6}, 1 \times 10^{-5}, 1 \times 10^{-4}\}$ and select 1×10^{-5} as the default learning rate. We set the group size to 16. To stabilise the training of the Text-to-Image diffusion model, for each group sample, we generate four images and average their scores. Training continues until the reward curve converges. All experiments are conducted on a single 80 GB H100 GPU. We will release our code and datasets for future research.

[5] `https://huggingface.co/`
[6] `https://huggingface.co/docs/trl/`
[7] `https://github.com/GAD-cell/vlm-grpo`

8.4.2 Benchmark Analysis

In this section, we analyse the results of our two benchmarks: the **Completion Quality Benchmark** and the **Recommendation Benchmark**.

Table 8.2: Average Text Quality Metrics (mean ± std) Across All Categories for I→T tasks on Content Quality Benchmark. ↑: larger is better.

Model	Cosine ↑	Euclidean ↑	Overlap ↑	BERTScore ↑	SBERT ↑	METEOR ↑	MoverScore ↑
Gemma-3-4B	0.166 ± 0.043	0.438 ± 0.008	0.213 ± 0.038	0.862 ± 0.008	0.640 ± 0.032	0.244 ± 0.066	0.471 ± 0.019
Gemma-3-12B	0.196 ± 0.062	0.447 ± 0.022	0.251 ± 0.063	0.869 ± 0.012	0.662 ± 0.033	0.271 ± 0.078	0.492 ± 0.035
Gemma-3-27B	**0.210 ± 0.067**	**0.451 ± 0.028**	**0.269 ± 0.064**	**0.872 ± 0.012**	**0.682 ± 0.034**	**0.299 ± 0.077**	**0.505 ± 0.037**
Qwen2.5-VL-7B-Instruct	0.196 ± 0.061	0.446 ± 0.017	0.254 ± 0.060	0.870 ± 0.011	0.650 ± 0.038	0.263 ± 0.076	0.489 ± 0.034
Qwen2.5-VL-32B-Instruct	0.169 ± 0.053	0.439 ± 0.011	0.225 ± 0.050	0.864 ± 0.011	0.645 ± 0.033	0.268 ± 0.077	0.473 ± 0.026
Qwen2.5-VL-72B-Instruct	0.178 ± 0.053	0.441 ± 0.013	0.235 ± 0.048	0.865 ± 0.011	0.651 ± 0.033	0.270 ± 0.078	0.477 ± 0.026

Table 8.3: Image quality metrics (mean ± std) across all categories. ↑: larger is better; ↓: smaller is better

Model	PSNR ↑	SSIM ↑	MSE ↓	LPIPS ↓	CLIP ↑	FID ↓
Gemma-3-4B	8.164 ± 0.576	0.347 ± 0.048	0.171 ± 0.018	0.723 ± 0.012	0.716 ± 0.020	71.81 ± 6.46
Gemma-3-12B	**8.516 ± 0.662**	**0.373 ± 0.049**	**0.160 ± 0.020**	**0.700 ± 0.017**	0.727 ± 0.020	70.78 ± 7.54
Gemma-3-27B	8.184 ± 0.621	0.349 ± 0.050	0.174 ± 0.022	0.716 ± 0.016	0.722 ± 0.018	71.57 ± 6.61
Qwen2.5-VL-7B-Instruct	8.260 ± 0.662	0.340 ± 0.056	0.169 ± 0.022	0.721 ± 0.020	0.723 ± 0.022	70.83 ± 8.95
Qwen2.5-VL-32B-Instruct	8.184 ± 0.572	0.342 ± 0.051	0.170 ± 0.018	0.716 ± 0.013	**0.727 ± 0.020**	**69.53 ± 6.23**
Qwen2.5-VL-72B-Instruct	7.970 ± 0.599	0.332 ± 0.052	0.179 ± 0.021	0.731 ± 0.013	0.716 ± 0.021	72.54 ± 7.13

Table 8.4: Category-wise Average Text Quality Metrics (mean ± std) Across All Models for Image-to-Text tasks on Content Quality Benchmark. ↑: larger is better

Category	Cosine ↑	Euclidean ↑	Overlap ↑	BERTScore ↑	SBERT ↑	METEOR ↑	MoverScore ↑
All Beauty	0.196 ± 0.018	0.443 ± 0.003	0.243 ± 0.019	0.865 ± 0.004	0.683 ± 0.017	0.255 ± 0.020	0.489 ± 0.013
Arts, Crafts & Sewing	0.159 ± 0.013	0.437 ± 0.002	0.216 ± 0.016	0.862 ± 0.003	0.651 ± 0.013	0.246 ± 0.014	0.473 ± 0.010
Electronics	0.168 ± 0.015	0.438 ± 0.003	0.235 ± 0.017	0.865 ± 0.004	0.644 ± 0.019	0.230 ± 0.020	0.482 ± 0.012
Home & Kitchen	0.161 ± 0.015	0.437 ± 0.002	0.218 ± 0.015	0.862 ± 0.003	0.651 ± 0.014	0.235 ± 0.014	0.474 ± 0.009
Industrial & Scientific	0.144 ± 0.012	0.434 ± 0.002	0.199 ± 0.016	0.858 ± 0.004	0.607 ± 0.017	0.220 ± 0.017	0.460 ± 0.011
Musical Instruments	0.156 ± 0.014	0.436 ± 0.002	0.212 ± 0.018	0.863 ± 0.004	0.611 ± 0.016	0.240 ± 0.018	0.462 ± 0.011
Office Products	0.171 ± 0.017	0.439 ± 0.003	0.234 ± 0.019	0.865 ± 0.004	0.663 ± 0.014	0.246 ± 0.018	0.482 ± 0.011
Toys & Games	0.176 ± 0.017	0.439 ± 0.003	0.226 ± 0.019	0.867 ± 0.004	0.664 ± 0.015	0.274 ± 0.020	0.476 ± 0.012
Video Games	**0.341 ± 0.042**	**0.490 ± 0.026**	**0.389 ± 0.050**	**0.897 ± 0.008**	**0.723 ± 0.022**	**0.478 ± 0.029**	**0.564 ± 0.034**

Table 8.5: Category-wise average image quality metrics (mean ± std) across all models. ↑: larger is better; ↓: smaller is better

Category	PSNR ↑	SSIM ↑	MSE ↓	LPIPS ↓	CLIP ↑	FID ↓
All Beauty	8.536 ± 0.232	**0.422 ± 0.014**	0.158 ± 0.007	0.705 ± 0.015	0.708 ± 0.007	70.24 ± 2.08
Arts, Crafts & Sewing	8.758 ± 0.173	0.288 ± 0.011	0.155 ± 0.005	0.731 ± 0.009	0.698 ± 0.005	80.43 ± 2.26
Electronics	7.697 ± 0.168	0.381 ± 0.016	0.187 ± 0.006	0.726 ± 0.012	0.752 ± 0.005	**61.80 ± 1.14**
Home & Kitchen	**8.997 ± 0.217**	0.357 ± 0.015	**0.144 ± 0.005**	0.709 ± 0.008	**0.756 ± 0.005**	63.74 ± 1.73
Industrial & Scientific	8.950 ± 0.270	0.419 ± 0.012	0.149 ± 0.009	**0.688 ± 0.013**	0.735 ± 0.006	65.14 ± 1.30
Musical Instruments	7.304 ± 0.258	0.348 ± 0.023	0.203 ± 0.013	0.722 ± 0.015	0.724 ± 0.005	66.97 ± 1.23
Office Products	8.289 ± 0.132	0.330 ± 0.010	0.168 ± 0.003	0.737 ± 0.007	0.705 ± 0.005	72.99 ± 1.70
Toys & Games	7.962 ± 0.238	0.275 ± 0.020	0.174 ± 0.009	0.718 ± 0.014	0.708 ± 0.006	78.38 ± 0.84
Video Games	7.424 ± 0.078	0.303 ± 0.018	0.195 ± 0.005	0.725 ± 0.013	0.711 ± 0.009	80.89 ± 5.55

Completion Quality Benchmark For the CQBench, we evaluate six advanced MLLMs from the Gemma-3 and Qwen2.5-VL families. A general observation can be drawn from Tables 8.2

and 8.3: BERTScore and CLIP similarity are consistently high with minimal variation across models, indicating strong semantic alignment. In contrast, word- and patch-level metrics (e.g., Cosine, Euclidean, Overlap, PSNR, SSIM, MSE, LPIPS) yield much lower score ranges, suggesting that current MLLMs capture semantic meaning well but struggle with exact word-level or pixel-level correspondence.

More detailed result analysis is reported from two perspectives—model scale and modality category—aiming to reveal the capability of MLLMs to complete missing modalities of e-commerce products.

Table 8.6: Human evaluation results (5 annotators, 54 items, and 9 categories). Scores are on a 1–5 Likert scale; **bold** indicates the best result in each column.

Model	**Image→Text**				**Text→Image**			
	Factual Accuracy	Completeness	Marketing Appeal	Overall	Visual Plausibility	Attribute Fidelity	Commercial Readiness	Overall
Gemma-3-4B	2.42	2.41	3.46	2.76	4.01	3.29	3.16	3.49
Gemma-3-12B	2.98	2.60	3.79	3.12	3.97	3.44	3.26	3.56
Gemma-3-27B	3.00	**2.73**	**3.96**	**3.23**	**4.20**	**3.83**	**3.70**	**3.91**
Qwen2.5-VL-7B	**3.29**	2.44	3.37	3.03	4.11	3.58	3.37	3.68
Qwen2.5-VL-32B	2.83	2.52	3.92	3.09	4.12	3.44	3.36	3.64
Qwen2.5-VL-72B	3.07	2.51	3.70	3.09	4.08	3.59	3.41	3.69

Model Scales. As shown in Figure 8.4, within the Gemma-3 family, performance on the **Image-to-Text** task improves with model scale. Specifically, the 27B variant obtains the highest cosine similarity (0.210) and BERTScore (0.872) among all six models, as shown in Table 8.2. In terms of cosine similarity, the 4B model (0.166) is approximately 21% lower than the 27B model, whereas the 12B model (0.196) is much closer, with a gap of only about 7%.

In sharp contrast, Qwen2.5-VL peaks at its 7B scale; both the 32B and 72B versions slide by roughly 10%–15% on the same metrics. For **Text-to-Image**, scale is not monotonic: Gemma's 12B delivers the best PSNR (8.52) and SSIM (0.373), while Qwen again favours the 7B version (PSNR 8.26) as described in Table 8.3. These opposite trends indicate that capacity alone does not dictate cross-modal quality.

Category-level divergence. Radar visualisations expose orthogonal difficulty profiles between the two tasks as described in Figure 8.4. On **Image-to-Text**, *Video Games* is an outlier—average cosine similarity soars to 0.341, almost double the mean as described in Table 8.4. Switching to **Text-to-Image**, gaps narrow overall, yet polarity flips: *Home & Kitchen* items yield the most similar reconstructions (PSNR 8.997), whereas *Video Games* sinks to the worst (7.42). The same product category, *Video Games*, thus oscillates between the "simplest" and the "hardest" tasks.

This observation validates the benchmark by showing that, even within the same category, the two tasks exhibit completely different levels of difficulty.

Table 8.7: Benchmark of MLLMs on Image-to-Text (I→T) and Text-to-Image (T→I). Best and second-best per metric are **bold** and underlined. "Rel. Diff. (vs. GT)" denotes the relative difference from the ground truth.

Task	Completion Models	R@5	R@10	R@20	R@50	N@5	N@10	N@20	N@50
	Ground Truth	0.0346	0.0552	0.0851	0.1461	0.0224	0.0291	0.0367	0.0488
I→T	Gemma-3-4B	**0.0344**	0.0547	**0.0856**	**0.1469**	**0.0221**	0.0287	**0.0366**	**0.0488**
	Gemma-3-12B	0.0342	0.0546	0.0846	0.1443	0.0221	0.0287	0.0364	0.0483
	Gemma-3-27B	0.0339	0.0547	0.0844	0.1454	0.0219	0.0287	0.0362	0.0483
	Qwen2.5-VL-7B-Instruct	0.0339	0.0546	0.0852	0.1462	0.0219	0.0286	0.0364	0.0485
	Qwen2.5-VL-32B-Instruct	0.0342	0.0545	0.0855	0.1458	0.0220	0.0286	0.0365	0.0485
	Qwen2.5-VL-72B-Instruct	0.0340	**0.0550**	0.0854	0.1463	0.0219	**0.0287**	0.0365	0.0486
	Rel. Diff. (vs. GT)	−0.58%	−0.36%	+0.58%	+0.55%	−1.34%	−1.37%	−0.27%	0.00%
T→I	Gemma-3-4B	0.0340	0.0542	0.0841	0.1437	0.0221	0.0286	0.0362	0.0481
	Gemma-3-12B	0.0340	0.0542	0.0837	**0.1448**	0.0220	0.0286	0.0360	**0.0482**
	Gemma-3-27B	**0.0340**	0.0538	0.0843	0.1441	**0.0221**	0.0285	0.0362	0.0482
	Qwen2.5-VL-7B-Instruct	0.0339	0.0542	**0.0844**	0.1436	0.0220	**0.0286**	**0.0363**	0.0481
	Qwen2.5-VL-32B-Instruct	0.0339	0.0538	0.0840	0.1440	0.0219	0.0284	0.0361	0.0480
	Qwen2.5-VL-72B-Instruct	0.0339	**0.0543**	0.0842	0.1443	0.0220	0.0286	0.0362	0.0482
	Rel. Diff. (vs. GT)	−1.73%	−1.63%	−0.82%	−0.89%	−1.34%	−1.72%	−1.09%	−1.23%

Human Evaluation In addition to automated metrics, we also perform a blind human evaluation. As shown in Table 8.6, larger models generally achieve better human-perceived completion quality, with Gemma-3-27B obtaining the best overall performance on both tasks. A consistent pattern is that fluency- or appearance-oriented dimensions are stronger than attribute-faithfulness ones: Marketing Appeal is higher than Factual Accuracy for Image→Text, and Visual Plausibility is higher than Attribute Fidelity for Text→Image, suggesting that current MLLMs are better at producing appealing text and visually coherent images than at faithfully preserving fine-grained product attributes. Inter-annotator agreement is measured with two complementary statistics. At the item level, the two-way random-effects intraclass correlation coefficient for the mean of $k = 5$ raters (ICC(2,k), absolute agreement) averages **0.90** across the six evaluation dimensions, indicating *excellent* reliability. At the model level, the mean pairwise Spearman rank correlation on per-model mean scores is $\bar{r} = \mathbf{0.70}$, showing consistent model rankings across annotators. Together, these results support the reliability of the reported mean scores and the robustness of our comparisons across the six models. Note that human evaluation serves as a complementary diagnostic signal alongside automatic metrics. Given its limited scale (54 products, 5 annotators), we do not treat human scores as standalone evidence, but use them to support and contextualise the automatic results.

Recommendation Benchmark Across nine categories and three recommenders, replacing the ground-truth modality with generated content causes only minor changes: Recall and NDCG shift by at most 2%, with relative deltas between −0.6% and +0.6% across eight metrics. Gemma-3-4B performs best on **Image-to-Text** completion (Recall@10 = 0.0547), while **Text-to-Image** results are tightly grouped, with Qwen-7B achieving three best scores and one runner-up. At the category level, *Electronics* is hardest under missing text. A notable outlier is Gemma-3-12B on *Beauty*, likely due to the category's small dataset size, since its performance nor-



Figure 8.5: Heatmap Illustrating the Differences Among MLLM Completion Methods Across Various Recommendation Categories. Colours closer to blue indicate greater performance degradation compared to using the ground-truth multimodal inputs.

Table 8.8: Summary of paired Wilcoxon signed-rank tests (n=9 datasets, α=0.05). Each cell reports the mean relative difference (%) from ground truth and the number of non-significant backbone comparisons out of 3 (VBPR / FREEDOM / BM3). **All 3/3** cells indicate full non-significance across all backbones. † indicates that at least one backbone yields $p < 0.05$.

	I→T (text missing)		**T→I (image missing)**	
Completion Model	R@20	N@20	R@20	N@20
Gemma-3-4B	−1.0% **(3/3)**	−1.2% **(3/3)**	+0.0% (2/3)†	−1.3% (2/3)†
Gemma-3-12B	−1.2% (2/3)†	−1.4% (2/3)†	−1.6% **(3/3)**	−2.3% (2/3)†
Gemma-3-27B	−0.5% **(3/3)**	−0.8% **(3/3)**	−2.4% **(3/3)**	−3.1% **(3/3)**
Qwen2.5-VL-7B	−0.5% **(3/3)**	−0.8% **(3/3)**	−0.7% (2/3)†	−2.2% **(3/3)**
Qwen2.5-VL-32B	−0.6% (2/3)†	−1.1% (2/3)†	+0.2% **(3/3)**	−1.4% **(3/3)**
Qwen2.5-VL-72B	−1.0% **(3/3)**	−1.2% **(3/3)**	−0.4% (2/3)†	−1.8% **(3/3)**
Overall	*32/36 non-significant (88.9%)*		*31/36 non-significant (86.1%)*	

malises on larger categories. Unlike CQBench, recommendation performance with completed modalities correlates only weakly with content quality, likely because recommendation depends more on semantic-level representations than low-level differences (X. Zhou & Shen, 2023).

To examine whether the recommendation differences between LLM-completed and ground-truth modalities are meaningful, we conduct paired Wilcoxon signed-rank tests on Recall@20 and NDCG@20, treating the 9 benchmark datasets as paired observations. For each completion model, recommender backbone, and task direction, we compare the recommendation results obtained with completed modalities against those with ground-truth modalities. As summarised in Table 8.8, most comparisons are not statistically significant, with 63 out of 72 tests (87.5%) yielding $p \geq 0.05$. The mean relative differences are also small (−0.80% for Recall@20 and −1.56% for NDCG@20), indicating that LLM-completed modalities provide recommendation performance largely comparable to ground-truth features in most settings. The generally small

and often non-significant differences suggest that (for the evaluated recommenders and categories) recommendation performance is largely driven by semantic-level representations and is relatively robust to moderate modality-generation noise. This indicates downstream usability of completed modalities in many cases, but also implies that RecBench in its current form may offer limited sensitivity for fine-grained ranking among strong completion methods.

Summary Our benchmark yields six key insights. For the *CQBench*, (1) current MLLMs effectively capture semantic meaning but struggle with precise word-level and pixel-level correspondence; (2) model scale exhibits non-monotonic effects across architectures and tasks, indicating that larger size alone does not guarantee better multimodal completion; (3) completion difficulty varies substantially by task direction and product category, revealing inherent asymmetries; and (4) human evaluation further shows that current MLLMs are better at producing fluent text and visually plausible images than at faithfully preserving fine-grained product attributes. From the *Recommendation Benchmark*, we further observe that (5) most MLLM-generated modalities achieve performance comparable to ground-truth modalities, except for a few specific cases; and (6) recommendation performance with completed modalities does not strongly correlate with the Content Quality Benchmark, from both model-wise and category-wise perspectives.

8.4.3 GRPO Results

Motivated by the performance gaps observed in the Completion Quality benchmark—where existing MLLMs exhibit limited alignment with task objectives, likely due to insufficient exposure during pretraining—we investigate *Group Relative Policy Optimisation (GRPO)* as a fine-tuning strategy to better align model behaviour with both directions of the presentation track. We select *Qwen2.5-VL-7B-Instruct* as the base model, given its strong performance on our benchmark and its computational efficiency. In this section, we present only three metrics due to space limitations; the remaining metrics exhibit consistent trends.

Table 8.9: Average Text Quality Metrics (mean $\pm$ std) Across All Categories for GRPO Training for Qwen2.5-VL-7B-Instruct. $\uparrow$: larger is better

Model	Cosine $\uparrow$	Overlap $\uparrow$	BERTScore $\uparrow$
Before GRPO	0.196 ± 0.061	0.254 ± 0.060	0.870 ± 0.011
After GRPO	$\mathbf{0.249 \pm 0.084}$	$\mathbf{0.295 \pm 0.085}$	$\mathbf{0.873 \pm 0.017}$

For *Image-to-Text*, GRPO yields substantial improvements; for example, cosine similarity increases from 0.196 to 0.249 (+27%), as shown in Table 8.9, with similar gains in the other two metrics. BERTScore shows a relatively marginal improvement, primarily because its initial performance is already high, making further enhancement challenging. This confirms the effectiveness of GRPO in this direction. However, for *Text-to-Image* as shown in Table 8.10, no clear

Table 8.10: Average Image Quality Metrics (mean ± std) Across All categories for GRPO Training. ↑: larger is better; ↓: smaller is better

Model	SSIM ↑	MSE ↓	CLIP ↑
Before GRPO	0.340 ± 0.056	0.169 ± 0.022	0.723 ± 0.022
After GRPO	0.339 ± 0.056	0.170 ± 0.022	0.723 ± 0.022

gains are observed, with the reward signal showing no upward trend during training—indicating that further investigation is needed to improve alignment in this more challenging direction.

These findings demonstrate that GRPO is highly effective for aligning models in the Image-to-Text direction, substantially improving generation quality. However, its limited impact on Text-to-Image highlights a more complex optimisation landscape, suggesting the need for task-specific strategies and further exploration in this direction.

8.5 Conclusion and Future Work

We study the underexplored task of missing-modality completion in e-commerce by introducing MMPCBench, the first systematic benchmark for product missing-modality completion, which takes a dual-perspective evaluation design by jointly assessing presentation-oriented completion quality and recommendation-oriented downstream utility through the Content Quality Benchmark (CQBench) and the Recommendation Benchmark (RecBench). MMPCBench covers both Image-to-Text and Text-to-Image tasks across nine product categories with six state-of-the-art MLLMs from the Qwen2.5-VL and Gemma-3 families. Experiments on CQBench show that current MLLMs capture high-level semantics but struggle with precise word- and pixel-level alignment. Scaling does not reliably improve completion: Qwen2.5-VL performs best at 7B, while Gemma-3 improves with scale. Difficulty also varies by task and category, with Video Games strong in I→T but hardest for T→I. On RecBench, completed modalities usually approach ground-truth recommendation performance, yet content quality correlates only weakly with accuracy, suggesting semantic alignment matters more than exact reconstruction. GRPO substantially improves I→T but has limited impact on T→I, highlighting a key weakness in current alignment methods. MMPCBench provides a standardised, reproducible benchmark for fair comparison, and our findings support prioritising semantic alignment over low-level fidelity while guiding model choice by domain.

Future work will tackle the T→I bottleneck through multi-level reward mechanisms combining perceptual and semantic metrics, potentially using two-stage diffusion refinement. A limitation of the current Text→Image track is that absolute image quality depends on the diffusion backbone; evaluating multiple renderers and/or renderer-free prompt quality is an important direction for future work. Another limitation is that our human study is intentionally small-scale and exploratory (54 items, 5 annotators). It is meant to complement automatic met-

rics rather than replace them. Scaling to larger and more diverse annotator pools is left to future exploration. Improving RecBench sensitivity is another important direction, e.g., evaluating settings with stronger reliance on fine-grained attributes (cold-start, visually distinctive items, attribute-aware objectives), adding additional recommender backbones that depend more heavily on pixel/patch-level cues, and exploring task variants where modality quality is known to matter more. Exploring parameter-efficient alignment methods and transfer learning across categories could improve accessibility and reduce domain-specific training requirements for broader deployment.

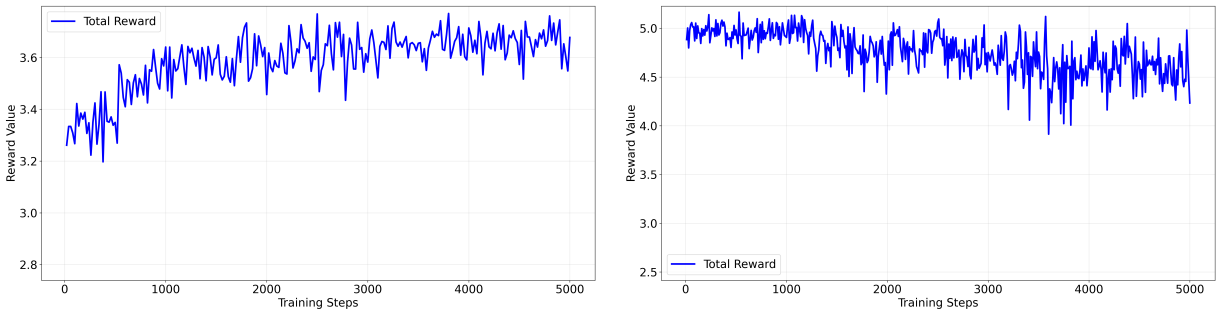
8.6 Additional Details

8.6.1 More Results of GRPO

As shown in Figures 8.6a and 8.6b, we present the training rewards for the Image-to-Text (I→T) and Text-to-Image (T→I) tasks. We adopt the Qwen2.5-VL-7B-Instruct as the default model for GRPO training. Each task incorporates an output JSON-format checking reward combined with a similarity reward. For the T→I task, we additionally apply a scaling factor to adjust the reward range; we tuned and set 2 for I→T, while 4 for T→I, which explains the differences of absolute values.

The I→T reward exhibits a clear upward trend, improving rapidly within the first 1,000 steps and then continuing to rise steadily thereafter. In contrast, the T→I reward remains relatively stable during the initial 1,000 steps and subsequently begins to decline. This pattern suggests that the I→T task is comparatively easier to optimise with GRPO, while the T→I task may require more sophisticated reward designs to achieve consistent improvement, which should be studied in future work.

After showing GRPO's overall gains on I→T, we report per-category results in Table 8.11. GRPO improves most categories and metrics: for *Video Games*, Cosine similarity rises from 0.37 to 0.48 and BERTScore from 0.90 to 0.92. Gains are larger for Cosine, Euclidean, and



(a) Image-to-Text task (b) Text-to-Image task

Figure 8.6: Reward Curves of different tasks.

Table 8.11: Text Quality Metrics (mean ± std) by Method and Category. ↑: larger is better

Method	Category	Cosine ↑	Euclidean ↑	Overlap ↑	BERTScore ↑
Before GRPO	All Beauty	0.198 ± 0.130	0.443 ± 0.023	0.248 ± 0.136	0.866 ± 0.030
	Arts, Crafts & Sewing	0.165 ± 0.129	0.439 ± 0.033	0.228 ± 0.149	0.865 ± 0.031
	Electronics	0.181 ± 0.118	0.440 ± 0.020	0.249 ± 0.139	0.869 ± 0.032
	Home & Kitchen	0.166 ± 0.105	0.438 ± 0.024	0.224 ± 0.122	0.864 ± 0.027
	Industrial & Scientific	0.152 ± 0.117	0.436 ± 0.019	0.208 ± 0.139	0.860 ± 0.034
	Musical Instruments	0.163 ± 0.127	0.438 ± 0.022	0.225 ± 0.148	0.866 ± 0.032
	Office Products	0.189 ± 0.142	0.443 ± 0.039	0.252 ± 0.168	0.869 ± 0.034
	Toys & Games	0.184 ± 0.131	0.441 ± 0.024	0.235 ± 0.138	0.870 ± 0.031
	Video Games	**0.365 ± 0.255**	**0.492 ± 0.101**	**0.420 ± 0.248**	**0.901 ± 0.043**
After GRPO	All Beauty	0.270 ± 0.140	0.455 ± 0.026	0.263 ± 0.121	0.857 ± 0.024
	Arts, Crafts & Sewing	0.202 ± 0.143	0.445 ± 0.032	0.269 ± 0.160	0.870 ± 0.033
	Electronics	0.233 ± 0.136	0.449 ± 0.030	0.300 ± 0.150	0.880 ± 0.034
	Home & Kitchen	0.205 ± 0.119	0.444 ± 0.027	0.268 ± 0.141	0.870 ± 0.029
	Industrial & Scientific	0.190 ± 0.135	0.443 ± 0.034	0.251 ± 0.152	0.867 ± 0.033
	Musical Instruments	0.195 ± 0.144	0.444 ± 0.028	0.255 ± 0.168	0.869 ± 0.036
	Office Products	0.226 ± 0.150	0.450 ± 0.044	0.292 ± 0.174	0.876 ± 0.035
	Toys & Games	0.244 ± 0.132	0.451 ± 0.024	0.230 ± 0.115	0.854 ± 0.024
	Video Games	**0.477 ± 0.318**	**0.575 ± 0.208**	**0.528 ± 0.301**	**0.916 ± 0.052**

Overlap, while BERTScore improves less because the base model is already strong. In a few cases, such as *All Beauty* and *Toys & Games*, BERTScore drops slightly, but most categories show consistent improvement. Overall, GRPO reinforcement learning strengthens alignment between generated captions and ground-truth descriptions.

8.6.2 More Analysis of the Results

(1) CQBench: Image-to-Text Task. From Table 8.12 it is clear that the Video Games category is the best for Image-to-Text generation: across every model and metric it posts the highest scores. For example, with Gemma-3-27B the results reach Cosine 0.394 ± 0.292, Euclidean 0.530 ± 0.175, Overlap 0.446 ± 0.283, and BERTScore 0.906 ± 0.051.

Looking at the model families, Gemma-3 consistently outperforms Qwen2.5-VL. Both the 27B and 12B checkpoints of Gemma-3 score higher across almost all datasets than any of the three Qwen2.5-VL variants. Within families the scaling behaviour diverges: Gemma-3 improves steadily as its parameter count grows, whereas Qwen2.5-VL peaks at the 7B model, followed by the 72B version, with the 32B model trailing behind.

(2) CQBench: Text-to-Image Task. Table 8.13 shows that the Home & Kitchen and Industrial & Scientific datasets achieve the strongest results in the Text-to-Image task. Across models, Gemma-3-12B is the strongest overall. In the “All Beauty" dataset, Gemma-3-12B hits the top SSIM 0.446 ± 0.181, lowest MSE 0.148 ± 0.073, and lowest LPIPS 0.684 ± 0.117. Similar to Table 8.12, scaling helps Gemma, but not Qwen2.5-VL. Gemma’s metrics improve steadily from 4B → 12B → 27B. For Qwen2.5-VL, however, the 7B model outperforms the 32B and 72B variants on PSNR, SSIM, and LPIPS.

(3) RecBench: Image-to-Text Task. According to Tables 8.14, 8.15, and 8.16, the captions generated by multimodal large-language models (MLLMs) achieve nearly the same performance



Figure 8.7: Representative qualitative success and failure cases for Image→Text and Text→Image. HE denotes the average human-evaluation score.

as the original texts on all three recommenders. In VBPR, BM3, and FREEDOM, the evaluation scores for MLLM-completed captions remain very close to the real-text baselines. For example, in the FREEDOM model for the “Arts" dataset, the ground-truth Recall@20 is 0.1446, while the caption created by Gemma-3-27B posts an almost identical 0.1442. For the “Video Games" category under BM3, the ground-truth NDCG@50 is 0.0814—the exact same value achieved by Qwen2.5-VL-7B. This consistency shows that the generated captions successfully preserve the core semantic signals required by these different recommendation algorithms.

Unlike the patterns in Tables 8.12 and 8.13, no single MLLM is universally superior for this task; the optimal model depends strongly on the product category and the recommender being used. With FREEDOM, for instance, Qwen2.5-VL-7B and Qwen2.5-VL-72B perform best in the “Instruments" category (NDCG@20 = 0.0665), whereas Gemma-3-27B excels in “Toys & Games" (NDCG@20 = 0.0382). These variations highlight that, although generated captions can effectively replace missing modality information, the best generator is context-specific.

(4) RecBench: Text-to-Image Task. According to Tables 8.14, 8.15, and 8.16, the Text-to-Image task shows that replacing ground-truth images with MLLM-generated ones has minimal impact on the recommendation performance of VBPR, BM3, and FREEDOM. The Recall and NDCG metrics for completions are consistently on par with the ground-truth baseline. For example, using the FREEDOM recommender on the “Arts" dataset, the ground-truth Recall@10 is 0.0912, while the Qwen2.5-VL-32B model not only matches but slightly exceeds this with a score of 0.0926. Similarly, for the BM3 model in the “Home" category, the ground-truth NDCG@50 is 0.0473, and Gemma-3-12B achieves a higher score of 0.0494. This demonstrates that the generated images are effective surrogates for original images in recommendation tasks.

(5) Case study. Figure 8.7 presents representative qualitative success and failure cases for both

Image→Text and Text→Image. As shown in the figure, successful Image→Text generation can correctly recover the product category together with key attributes such as brand, colour, and item count, while successful Text→Image generation produces images that are visually consistent with the input description at both the category and style levels. In contrast, the failure cases reveal two common limitations. For Image→Text, the model may capture only coarse semantic cues but still confuse the exact product type or hallucinate fine-grained attributes. For Text→Image, the generated image may deviate substantially from the intended product category and instead reflect only partial textures or vague visual patterns mentioned or implied by the text. These examples suggest that, although current models can often preserve high-level semantics, improving fine-grained attribute faithfulness and precise cross-modal grounding remains an important challenge for missing-modality completion.

8.6.3 Additional Implementation Details for Text→Image

Text-to-Image Generation. All images are synthesised using Stable Diffusion 3.5-Large-Turbo with 4 inference steps, a classifier-free guidance scale of 0.0, an output resolution of 1024×1024, a fixed random seed of 42, and no negative prompts.

MLLM Prompt. The multimodal LLM is prompted under a fixed structured protocol to ensure fair and reproducible comparison across models. We use a unified system instruction that enforces (i) JSON-only output with a single `"prompt"` key, (ii) English-only text, and (iii) a hard limit of 77 tokens, matching the CLIP text encoder capacity. We keep this prompt structure identical for all models to reduce prompt-induced variance, rather than applying model-specific prompt engineering.

GRPO Training. We fine-tune Qwen2.5-VL-7B-Instruct with LoRA (r=16, α=32, dropout 0.1) using GRPO. We use a learning rate of 1×10^{-5} with a cosine scheduler and 100 warm-up steps, an AdamW-8bit optimiser with weight decay 0.01, and clip ratio ε=0.2. Two reward signals are used: (1) an *image-quality reward* that averages CLIP similarity (rescaled to $[0,1]$) over 4 generated images (four seeds), scaled by a factor of 4; and (2) a *length reward* that linearly scales prompt token count up to the 77-token limit and penalises over-length outputs.

Table 8.12: Text Quality Metrics (mean ± std) by Model and Category. ↑: larger is better

Model	Category	Cosine ↑	Euclidean ↑	Overlap ↑	BERTScore ↑	SBERT ↑	METEOR ↑	MoverScore ↑
Gemma-3-4B	All Beauty	0.186 ± 0.111	0.441 ± 0.019	0.224 ± 0.113	0.862 ± 0.025	0.679 ± 0.143	0.242 ± 0.139	0.483 ± 0.074
	Arts, Crafts & Sewing	0.149 ± 0.113	0.435 ± 0.020	0.195 ± 0.121	0.857 ± 0.027	0.640 ± 0.173	0.228 ± 0.138	0.464 ± 0.078
	Electronics	0.155 ± 0.097	0.436 ± 0.016	0.217 ± 0.113	0.862 ± 0.028	0.627 ± 0.160	0.209 ± 0.124	0.475 ± 0.074
	Home & Kitchen	0.149 ± 0.087	0.435 ± 0.013	0.200 ± 0.102	0.859 ± 0.024	0.644 ± 0.140	0.218 ± 0.111	0.468 ± 0.068
	Industrial & Scientific	0.129 ± 0.097	0.432 ± 0.015	0.176 ± 0.111	0.854 ± 0.026	0.594 ± 0.178	0.194 ± 0.131	0.449 ± 0.075
	Musical Instruments	0.137 ± 0.105	0.433 ± 0.017	0.185 ± 0.118	0.857 ± 0.028	0.587 ± 0.178	0.210 ± 0.151	0.448 ± 0.079
	Office Products	0.149 ± 0.102	0.435 ± 0.017	0.205 ± 0.120	0.859 ± 0.027	0.647 ± 0.153	0.215 ± 0.134	0.466 ± 0.076
	Toys & Games	0.164 ± 0.103	0.437 ± 0.017	0.207 ± 0.111	0.863 ± 0.025	0.658 ± 0.139	0.258 ± 0.152	0.468 ± 0.072
	Video Games	0.279 ± 0.187	0.460 ± 0.043	0.312 ± 0.174	0.883 ± 0.035	0.686 ± 0.157	0.424 ± 0.262	0.515 ± 0.101
Gemma-3-12B	All Beauty	0.208 ± 0.120	0.444 ± 0.021	0.250 ± 0.124	0.867 ± 0.028	0.693 ± 0.150	0.260 ± 0.150	0.496 ± 0.081
	Arts, Crafts & Sewing	0.168 ± 0.122	0.438 ± 0.021	0.222 ± 0.136	0.863 ± 0.029	0.656 ± 0.179	0.248 ± 0.156	0.478 ± 0.086
	Electronics	0.178 ± 0.114	0.440 ± 0.020	0.241 ± 0.128	0.866 ± 0.030	0.650 ± 0.161	0.233 ± 0.137	0.487 ± 0.082
	Home & Kitchen	0.172 ± 0.097	0.438 ± 0.015	0.227 ± 0.112	0.864 ± 0.025	0.661 ± 0.145	0.238 ± 0.127	0.480 ± 0.072
	Industrial & Scientific	0.152 ± 0.106	0.435 ± 0.017	0.205 ± 0.122	0.860 ± 0.028	0.620 ± 0.176	0.220 ± 0.144	0.467 ± 0.082
	Musical Instruments	0.163 ± 0.120	0.438 ± 0.021	0.218 ± 0.136	0.864 ± 0.030	0.616 ± 0.176	0.242 ± 0.160	0.468 ± 0.089
	Office Products	0.172 ± 0.118	0.439 ± 0.020	0.232 ± 0.142	0.864 ± 0.029	0.660 ± 0.168	0.236 ± 0.150	0.483 ± 0.091
	Toys & Games	0.187 ± 0.117	0.441 ± 0.020	0.235 ± 0.129	0.869 ± 0.027	0.670 ± 0.145	0.278 ± 0.173	0.483 ± 0.082
	Video Games	0.367 ± 0.272	0.508 ± 0.146	0.425 ± 0.268	0.902 ± 0.048	0.732 ± 0.176	0.487 ± 0.298	0.589 ± 0.171
Gemma-3-27B	All Beauty	0.223 ± 0.120	0.447 ± 0.021	0.273 ± 0.129	0.871 ± 0.027	0.711 ± 0.148	0.292 ± 0.151	0.511 ± 0.083
	Arts, Crafts & Sewing	0.178 ± 0.121	0.440 ± 0.021	0.237 ± 0.139	0.866 ± 0.029	0.675 ± 0.177	0.270 ± 0.158	0.488 ± 0.088
	Electronics	0.185 ± 0.109	0.441 ± 0.018	0.259 ± 0.130	0.870 ± 0.031	0.679 ± 0.159	0.268 ± 0.142	0.502 ± 0.086
	Home & Kitchen	0.183 ± 0.103	0.440 ± 0.017	0.239 ± 0.118	0.867 ± 0.026	0.676 ± 0.143	0.259 ± 0.131	0.490 ± 0.076
	Industrial & Scientific	0.159 ± 0.109	0.437 ± 0.017	0.219 ± 0.131	0.863 ± 0.029	0.635 ± 0.176	0.247 ± 0.153	0.478 ± 0.086
	Musical Instruments	0.174 ± 0.122	0.439 ± 0.021	0.234 ± 0.146	0.866 ± 0.032	0.635 ± 0.182	0.264 ± 0.173	0.478 ± 0.095
	Office Products	0.190 ± 0.121	0.442 ± 0.028	0.254 ± 0.141	0.870 ± 0.030	0.688 ± 0.158	0.266 ± 0.158	0.496 ± 0.090
	Toys & Games	0.201 ± 0.120	0.443 ± 0.022	0.254 ± 0.133	0.872 ± 0.028	0.690 ± 0.142	0.311 ± 0.170	0.494 ± 0.085
	Video Games	**0.394 ± 0.292**	**0.530 ± 0.175**	**0.446 ± 0.283**	**0.906 ± 0.051**	**0.752 ± 0.177**	**0.511 ± 0.298**	**0.605 ± 0.185**
Qwen2.5-VL-7B-Instruct	All Beauty	0.198 ± 0.130	0.443 ± 0.023	0.248 ± 0.136	0.866 ± 0.030	0.672 ± 0.161	0.237 ± 0.157	0.486 ± 0.084
	Arts, Crafts & Sewing	0.165 ± 0.129	0.439 ± 0.033	0.228 ± 0.149	0.865 ± 0.031	0.647 ± 0.184	0.243 ± 0.164	0.478 ± 0.095
	Electronics	0.181 ± 0.118	0.440 ± 0.020	0.249 ± 0.139	0.869 ± 0.032	0.639 ± 0.169	0.229 ± 0.151	0.486 ± 0.089
	Home & Kitchen	0.166 ± 0.105	0.438 ± 0.024	0.224 ± 0.122	0.864 ± 0.027	0.638 ± 0.157	0.226 ± 0.134	0.474 ± 0.078
	Industrial & Scientific	0.152 ± 0.117	0.436 ± 0.019	0.208 ± 0.139	0.860 ± 0.034	0.590 ± 0.207	0.211 ± 0.163	0.462 ± 0.093
	Musical Instruments	0.163 ± 0.127	0.438 ± 0.022	0.225 ± 0.148	0.866 ± 0.032	0.611 ± 0.186	0.237 ± 0.179	0.467 ± 0.096
	Office Products	0.189 ± 0.142	0.443 ± 0.039	0.252 ± 0.168	0.869 ± 0.034	0.662 ± 0.182	0.249 ± 0.181	0.490 ± 0.105
	Toys & Games	0.184 ± 0.131	0.441 ± 0.024	0.235 ± 0.138	0.870 ± 0.031	0.657 ± 0.154	0.262 ± 0.182	0.479 ± 0.087
	Video Games	0.365 ± 0.255	0.492 ± 0.101	0.420 ± 0.248	0.901 ± 0.043	0.732 ± 0.178	0.476 ± 0.292	0.583 ± 0.154
Qwen2.5-VL-32B-Instruct	All Beauty	0.174 ± 0.113	0.439 ± 0.019	0.223 ± 0.119	0.860 ± 0.028	0.664 ± 0.160	0.243 ± 0.151	0.475 ± 0.077
	Arts, Crafts & Sewing	0.143 ± 0.109	0.434 ± 0.017	0.202 ± 0.125	0.859 ± 0.028	0.641 ± 0.178	0.243 ± 0.155	0.463 ± 0.080
	Electronics	0.151 ± 0.098	0.435 ± 0.016	0.220 ± 0.120	0.863 ± 0.031	0.635 ± 0.162	0.223 ± 0.141	0.472 ± 0.077
	Home & Kitchen	0.144 ± 0.087	0.434 ± 0.013	0.204 ± 0.105	0.860 ± 0.025	0.643 ± 0.149	0.237 ± 0.128	0.465 ± 0.070
	Industrial & Scientific	0.132 ± 0.097	0.432 ± 0.015	0.186 ± 0.121	0.855 ± 0.029	0.598 ± 0.188	0.225 ± 0.156	0.451 ± 0.082
	Musical Instruments	0.143 ± 0.114	0.434 ± 0.018	0.200 ± 0.136	0.861 ± 0.031	0.601 ± 0.181	0.240 ± 0.174	0.452 ± 0.087
	Office Products	0.156 ± 0.109	0.436 ± 0.018	0.220 ± 0.131	0.863 ± 0.029	0.656 ± 0.169	0.253 ± 0.159	0.472 ± 0.084
	Toys & Games	0.159 ± 0.110	0.437 ± 0.018	0.210 ± 0.120	0.864 ± 0.028	0.652 ± 0.154	0.269 ± 0.173	0.465 ± 0.079
	Video Games	0.316 ± 0.222	0.471 ± 0.061	0.363 ± 0.220	0.894 ± 0.043	0.717 ± 0.172	0.483 ± 0.296	0.545 ± 0.132
Qwen2.5-VL-72B-Instruct	All Beauty	0.184 ± 0.116	0.441 ± 0.021	0.239 ± 0.125	0.863 ± 0.028	0.677 ± 0.159	0.256 ± 0.160	0.483 ± 0.079
	Arts, Crafts & Sewing	0.151 ± 0.113	0.436 ± 0.019	0.211 ± 0.128	0.860 ± 0.029	0.647 ± 0.178	0.243 ± 0.160	0.467 ± 0.083
	Electronics	0.155 ± 0.101	0.436 ± 0.016	0.224 ± 0.120	0.862 ± 0.030	0.634 ± 0.158	0.219 ± 0.143	0.470 ± 0.077
	Home & Kitchen	0.152 ± 0.094	0.435 ± 0.015	0.216 ± 0.113	0.860 ± 0.026	0.646 ± 0.148	0.234 ± 0.133	0.469 ± 0.073
	Industrial & Scientific	0.143 ± 0.102	0.434 ± 0.016	0.199 ± 0.124	0.856 ± 0.030	0.602 ± 0.182	0.222 ± 0.160	0.455 ± 0.083
	Musical Instruments	0.154 ± 0.121	0.436 ± 0.021	0.211 ± 0.141	0.863 ± 0.031	0.613 ± 0.181	0.248 ± 0.181	0.456 ± 0.090
	Office Products	0.169 ± 0.120	0.439 ± 0.021	0.238 ± 0.146	0.865 ± 0.032	0.666 ± 0.166	0.253 ± 0.172	0.483 ± 0.095
	Toys & Games	0.163 ± 0.117	0.437 ± 0.020	0.213 ± 0.125	0.864 ± 0.028	0.653 ± 0.151	0.264 ± 0.181	0.465 ± 0.080
	Video Games	0.325 ± 0.238	0.478 ± 0.084	0.367 ± 0.230	0.895 ± 0.043	0.719 ± 0.168	0.488 ± 0.307	0.546 ± 0.144

Table 8.13: Image quality metrics (mean ± std) by model and category. ↑: larger is better; ↓: smaller is better

Model	Category	PSNR ↑	SSIM ↑	MSE ↓	LPIPS ↓	CLIP ↑	FID ↓
Gemma-3-4B	All Beauty	8.266 ± 2.141	0.410 ± 0.172	0.166 ± 0.077	0.723 ± 0.110	0.697 ± 0.102	73.19
	Arts, Crafts & Sewing	8.677 ± 2.454	0.283 ± 0.174	0.156 ± 0.079	0.738 ± 0.097	0.692 ± 0.106	82.73
	Electronics	7.674 ± 1.945	0.382 ± 0.153	0.186 ± 0.073	0.732 ± 0.095	0.748 ± 0.093	62.92
	Home & Kitchen	9.040 ± 2.462	0.361 ± 0.156	0.143 ± 0.072	0.714 ± 0.090	0.749 ± 0.094	64.60
	Industrial & Scientific	8.838 ± 2.540	0.413 ± 0.181	0.152 ± 0.083	0.698 ± 0.114	0.728 ± 0.108	66.67
	Musical Instruments	7.355 ± 1.820	0.352 ± 0.160	0.199 ± 0.077	0.721 ± 0.097	0.721 ± 0.113	67.52
	Office Products	8.275 ± 2.277	0.331 ± 0.170	0.168 ± 0.079	0.740 ± 0.099	0.702 ± 0.111	72.50
	Toys & Games	7.912 ± 1.770	0.276 ± 0.142	0.176 ± 0.074	0.718 ± 0.073	0.702 ± 0.107	78.33
	Video Games	7.444 ± 1.598	0.309 ± 0.168	0.192 ± 0.070	0.723 ± 0.089	0.711 ± 0.127	77.80
Gemma-3-12B	All Beauty	8.869 ± 2.313	**0.446 ± 0.181**	0.148 ± 0.073	0.684 ± 0.117	0.714 ± 0.096	69.17
	Arts, Crafts & Sewing	9.060 ± 2.633	0.309 ± 0.179	0.146 ± 0.081	0.716 ± 0.094	0.704 ± 0.103	80.93
	Electronics	7.956 ± 2.148	0.408 ± 0.154	0.178 ± 0.075	0.708 ± 0.093	0.758 ± 0.089	60.88
	Home & Kitchen	**9.384 ± 2.561**	0.386 ± 0.158	**0.135 ± 0.073**	0.694 ± 0.092	**0.761 ± 0.093**	62.19
	Industrial & Scientific	9.331 ± 2.613	0.439 ± 0.179	0.138 ± 0.079	**0.669 ± 0.113**	0.738 ± 0.103	65.36
	Musical Instruments	7.700 ± 1.866	0.385 ± 0.155	0.184 ± 0.072	0.697 ± 0.094	0.728 ± 0.107	65.92
	Office Products	8.481 ± 2.501	0.345 ± 0.175	0.163 ± 0.081	0.729 ± 0.104	0.707 ± 0.111	72.52
	Toys & Games	8.406 ± 1.778	0.312 ± 0.152	0.156 ± 0.064	0.694 ± 0.076	0.716 ± 0.108	78.29
	Video Games	7.460 ± 1.747	0.330 ± 0.169	0.193 ± 0.074	0.713 ± 0.095	0.714 ± 0.129	81.79
Gemma-3-27B	All Beauty	8.639 ± 2.273	0.430 ± 0.180	0.156 ± 0.082	0.694 ± 0.110	0.714 ± 0.092	68.09
	Arts, Crafts & Sewing	8.726 ± 2.607	0.288 ± 0.175	0.158 ± 0.087	0.730 ± 0.095	0.698 ± 0.104	81.73
	Electronics	7.729 ± 2.225	0.388 ± 0.158	0.189 ± 0.085	0.721 ± 0.095	0.751 ± 0.092	62.72
	Home & Kitchen	8.903 ± 2.366	0.354 ± 0.152	0.147 ± 0.074	0.710 ± 0.086	0.753 ± 0.093	64.73
	Industrial & Scientific	8.854 ± 2.634	0.413 ± 0.184	0.154 ± 0.089	0.688 ± 0.115	0.733 ± 0.101	66.08
	Musical Instruments	7.126 ± 2.042	0.342 ± 0.157	0.215 ± 0.096	0.730 ± 0.094	0.720 ± 0.110	68.46
	Office Products	8.297 ± 2.395	0.337 ± 0.171	0.169 ± 0.083	0.734 ± 0.097	0.705 ± 0.106	74.85
	Toys & Games	7.996 ± 1.995	0.279 ± 0.150	0.176 ± 0.084	0.711 ± 0.081	0.710 ± 0.103	78.79
	Video Games	7.387 ± 1.861	0.310 ± 0.168	0.199 ± 0.086	0.727 ± 0.092	0.716 ± 0.129	78.66
Qwen2.5-VL-7B-Instruct	All Beauty	8.691 ± 2.279	0.419 ± 0.177	0.153 ± 0.075	0.702 ± 0.108	0.715 ± 0.098	68.21
	Arts, Crafts & Sewing	8.767 ± 2.671	0.279 ± 0.177	0.155 ± 0.079	0.736 ± 0.094	0.698 ± 0.104	77.40
	Electronics	7.766 ± 2.163	0.375 ± 0.158	0.186 ± 0.078	0.723 ± 0.098	0.755 ± 0.089	60.86
	Home & Kitchen	8.981 ± 2.354	0.353 ± 0.153	0.144 ± 0.068	0.709 ± 0.092	0.760 ± 0.096	61.43
	Industrial & Scientific	9.203 ± 2.681	0.428 ± 0.184	0.141 ± 0.075	0.677 ± 0.122	0.740 ± 0.102	63.55
	Musical Instruments	7.227 ± 1.905	0.318 ± 0.169	0.207 ± 0.088	0.734 ± 0.096	0.723 ± 0.113	66.63
	Office Products	8.335 ± 2.354	0.329 ± 0.174	0.167 ± 0.080	0.735 ± 0.100	0.710 ± 0.110	72.44
	Toys & Games	7.937 ± 1.838	0.269 ± 0.147	0.175 ± 0.076	0.730 ± 0.082	0.708 ± 0.105	76.83
	Video Games	7.437 ± 1.743	0.286 ± 0.164	0.194 ± 0.076	0.746 ± 0.094	0.697 ± 0.131	90.17
Qwen2.5-VL-32B-Instruct	All Beauty	8.400 ± 2.059	0.412 ± 0.172	0.161 ± 0.074	0.713 ± 0.109	0.707 ± 0.097	71.81
	Arts, Crafts & Sewing	8.787 ± 2.558	0.289 ± 0.174	0.154 ± 0.080	0.726 ± 0.097	0.701 ± 0.105	77.84
	Electronics	7.602 ± 1.925	0.372 ± 0.150	0.189 ± 0.072	0.726 ± 0.095	0.754 ± 0.094	**60.57**
	Home & Kitchen	8.942 ± 2.349	0.348 ± 0.150	0.145 ± 0.070	0.708 ± 0.087	0.760 ± 0.094	63.44
	Industrial & Scientific	8.882 ± 2.541	0.415 ± 0.176	0.150 ± 0.079	0.688 ± 0.114	0.741 ± 0.102	63.60
	Musical Instruments	7.452 ± 1.853	0.360 ± 0.159	0.195 ± 0.075	0.711 ± 0.094	0.732 ± 0.109	65.31
	Office Products	8.274 ± 2.265	0.323 ± 0.171	0.167 ± 0.074	0.736 ± 0.101	0.710 ± 0.109	70.57
	Toys & Games	7.791 ± 1.770	0.257 ± 0.142	0.180 ± 0.071	0.725 ± 0.077	0.712 ± 0.105	78.73
	Video Games	7.525 ± 1.551	0.302 ± 0.173	0.188 ± 0.065	0.710 ± 0.091	0.723 ± 0.123	73.94
Qwen2.5-VL-72B-Instruct	All Beauty	8.354 ± 2.132	0.413 ± 0.171	0.163 ± 0.075	0.718 ± 0.116	0.703 ± 0.101	70.98
	Arts, Crafts & Sewing	8.533 ± 2.449	0.279 ± 0.171	0.162 ± 0.085	0.739 ± 0.091	0.694 ± 0.109	81.94
	Electronics	7.456 ± 1.939	0.362 ± 0.149	0.197 ± 0.085	0.743 ± 0.089	0.746 ± 0.092	62.84
	Home & Kitchen	8.730 ± 2.228	0.342 ± 0.149	0.151 ± 0.071	0.720 ± 0.083	0.753 ± 0.096	66.07
	Industrial & Scientific	8.590 ± 2.592	0.406 ± 0.180	0.162 ± 0.090	0.706 ± 0.120	0.728 ± 0.107	65.62
	Musical Instruments	6.964 ± 1.789	0.334 ± 0.152	0.219 ± 0.094	0.737 ± 0.090	0.722 ± 0.110	68.00
	Office Products	8.072 ± 2.101	0.317 ± 0.168	0.173 ± 0.074	0.749 ± 0.095	0.696 ± 0.112	75.07
	Toys & Games	7.733 ± 1.742	0.259 ± 0.138	0.182 ± 0.074	0.731 ± 0.071	0.703 ± 0.107	79.29
	Video Games	7.293 ± 1.770	0.281 ± 0.165	0.202 ± 0.083	0.732 ± 0.083	0.705 ± 0.129	83.01

Table 8.14: Performance comparison for model VBPR, combining Image-to-Text and Text-to-Image. R and N denote Recall and NDCG.

(a) Image-to-Text

		Datasets 1–5										Datasets 6–9							
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	**0.0162**	**0.0103**	0.0242	0.0129	0.0406	0.0170	0.0757	0.0239	Office	Gemma-3-4B	0.0183	0.0123	0.0295	0.0159	0.0453	0.0199	0.0822	0.0272
	Gemma-3-12B	0.0156	0.0100	0.0243	0.0128	0.0401	0.0168	0.0768	0.0241		Gemma-3-12B	**0.0193**	**0.0127**	0.0297	0.0161	**0.0474**	**0.0205**	**0.0842**	**0.0278**
	Gemma-3-27B	0.0158	0.0101	0.0249	0.0130	0.0408	0.0170	0.0767	0.0242		Gemma-3-27B	0.0180	0.0122	0.0292	0.0159	0.0458	0.0201	0.0822	0.0273
	Qwen2.5-VL-7B	0.0159	0.0101	0.0242	0.0128	**0.0416**	**0.0172**	**0.0772**	**0.0242**		Qwen2.5-VL-7B	0.0179	0.0122	**0.0300**	**0.0161**	0.0454	0.0199	0.0818	0.0272
	Qwen2.5-VL-32B	0.0159	0.0099	0.0246	0.0127	0.0414	0.0169	0.0768	0.0240		Qwen2.5-VL-32B	0.0189	0.0126	0.0297	0.0161	0.0474	0.0205	0.0836	0.0277
	Qwen2.5-VL-72B	0.0161	0.0101	**0.0252**	**0.0130**	0.0406	0.0169	0.0766	0.0241		Qwen2.5-VL-72B	0.0184	0.0122	0.0298	0.0159	0.0460	0.0200	0.0827	0.0272
	Ground Truth	0.0157	0.0100	0.0248	0.0129	0.0418	0.0171	0.0770	0.0242		Ground Truth	0.0187	0.0122	0.0300	0.0159	0.0458	0.0199	0.0819	0.0270
Beauty	Gemma-3-4B	0.0081	0.0044	0.0178	0.0079	0.0264	0.0101	0.0507	0.0151	Scientific	Gemma-3-4B	0.0206	0.0134	0.0328	0.0173	0.0481	0.0212	0.0817	0.0278
	Gemma-3-12B	0.0092	0.0051	0.0178	0.0081	0.0265	0.0103	0.0519	0.0155		Gemma-3-12B	0.0211	**0.0136**	0.0315	0.0170	0.0480	0.0212	0.0806	0.0276
	Gemma-3-27B	0.0094	0.0054	0.0175	0.0083	0.0281	0.0109	0.0492	0.0152		Gemma-3-27B	0.0206	0.0134	0.0327	0.0173	**0.0490**	**0.0214**	0.0806	0.0276
	Qwen2.5-VL-7B	0.0081	0.0045	0.0178	0.0079	0.0273	0.0103	0.0507	0.0151		Qwen2.5-VL-7B	**0.0212**	0.0135	**0.0332**	**0.0174**	0.0486	0.0213	**0.0819**	**0.0278**
	Qwen2.5-VL-32B	**0.0119**	**0.0065**	**0.0203**	**0.0094**	**0.0337**	**0.0129**	**0.0595**	**0.0181**		Qwen2.5-VL-32B	0.0204	0.0135	0.0319	0.0172	0.0474	0.0211	0.0793	0.0275
	Qwen2.5-VL-72B	0.0081	0.0044	0.0178	0.0078	0.0247	0.0097	0.0516	0.0152		Qwen2.5-VL-72B	0.0202	0.0132	0.0326	0.0172	0.0479	0.0211	0.0817	0.0277
	Ground Truth	0.0076	0.0045	0.0178	0.0080	0.0247	0.0098	0.0510	0.0153		Ground Truth	0.0214	0.0137	0.0323	0.0172	0.0478	0.0211	0.0804	0.0275
Electronics	Gemma-3-4B	0.0200	0.0129	0.0310	0.0166	0.0547	**0.0227**	0.1044	0.0330	Toys & Games	Gemma-3-4B	0.0125	0.0080	0.0198	0.0103	0.0302	0.0129	0.0564	0.0181
	Gemma-3-12B	0.0195	0.0127	0.0315	0.0167	0.0543	0.0225	0.1047	0.0329		Gemma-3-12B	0.0122	0.0079	0.0202	0.0104	0.0305	0.0130	**0.0566**	0.0181
	Gemma-3-27B	0.0193	0.0126	0.0315	0.0166	**0.0549**	0.0227	0.1045	0.0329		Gemma-3-27B	0.0125	0.0083	0.0203	0.0108	0.0324	0.0139	0.0558	**0.0185**
	Qwen2.5-VL-7B	0.0200	0.0128	0.0311	0.0165	0.0547	0.0226	0.1048	0.0330		Qwen2.5-VL-7B	**0.0129**	**0.0083**	0.0205	0.0107	**0.0331**	**0.0139**	0.0558	0.0183
	Qwen2.5-VL-32B	0.0199	0.0128	0.0321	0.0168	0.0549	0.0227	0.1047	0.0330		Qwen2.5-VL-32B	0.0125	0.0080	0.0200	0.0103	0.0308	0.0131	0.0562	0.0181
	Qwen2.5-VL-72B	**0.0210**	**0.0134**	**0.0332**	**0.0175**	0.0532	0.0226	**0.1067**	**0.0337**		Qwen2.5-VL-72B	0.0128	0.0082	**0.0209**	**0.0108**	0.0330	0.0139	0.0561	0.0184
	Ground Truth	0.0203	0.0136	0.0329	0.0179	0.0551	0.0236	0.1088	0.0346		Ground Truth	0.0124	0.0080	0.0201	0.0104	0.0307	0.0131	0.0565	0.0181
Home	Gemma-3-4B	0.0104	0.0067	0.0169	0.0089	0.0281	0.0117	0.0558	0.0172	Video Games	Gemma-3-4B	0.0237	0.0152	0.0398	0.0204	0.0647	0.0268	0.1186	0.0375
	Gemma-3-12B	0.0101	**0.0072**	0.0165	**0.0093**	**0.0302**	**0.0128**	**0.0616**	**0.0191**		Gemma-3-12B	0.0237	0.0151	0.0399	0.0204	0.0647	0.0267	**0.1198**	0.0377
	Gemma-3-27B	0.0108	0.0071	0.0173	0.0092	0.0297	0.0123	0.0600	0.0183		Gemma-3-27B	**0.0246**	**0.0157**	0.0392	0.0204	0.0645	0.0269	0.1191	0.0378
	Qwen2.5-VL-7B	**0.0108**	0.0069	0.0169	0.0090	0.0289	0.0120	0.0556	0.0173		Qwen2.5-VL-7B	0.0237	0.0152	**0.0409**	**0.0208**	**0.0649**	0.0269	0.1189	0.0377
	Qwen2.5-VL-32B	0.0102	0.0066	0.0171	0.0089	0.0276	0.0115	0.0580	0.0176		Qwen2.5-VL-32B	0.0235	0.0151	0.0399	0.0205	0.0649	0.0269	0.1185	0.0375
	Qwen2.5-VL-72B	0.0098	0.0063	**0.0174**	0.0088	0.0284	0.0116	0.0553	0.0170		Qwen2.5-VL-72B	0.0240	0.0156	0.0387	0.0204	0.0645	**0.0270**	0.1193	**0.0379**
	Ground Truth	0.0109	0.0069	0.0169	0.0088	0.0277	0.0116	0.0592	0.0179		Ground Truth	0.0244	0.0158	0.0402	0.0209	0.0651	0.0273	0.1206	0.0384
Instruments	Gemma-3-4B	0.0205	0.0139	0.0319	0.0177	0.0505	0.0224	0.0872	0.0297										
	Gemma-3-12B	0.0207	**0.0142**	**0.0324**	**0.0181**	0.0496	0.0224	0.0881	0.0301										
	Gemma-3-27B	0.0201	0.0139	0.0314	0.0176	0.0501	0.0223	0.0875	0.0298										
	Qwen2.5-VL-7B	0.0207	0.0141	0.0314	0.0177	0.0499	0.0224	0.0889	**0.0301**										
	Qwen2.5-VL-32B	**0.0215**	0.0142	0.0323	0.0178	**0.0510**	**0.0225**	**0.0893**	0.0301										
	Qwen2.5-VL-72B	0.0205	0.0140	0.0315	0.0176	0.0501	0.0223	0.0887	0.0300										
	Ground Truth	0.0219	0.0146	0.0319	0.0179	0.0519	0.0229	0.0913	0.0308										

(b) Text-to-Image

		Datasets 1–5										Datasets 6–9							
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	0.0144	0.0093	0.0250	0.0127	**0.0416**	0.0169	0.0738	0.0232	Office	Gemma-3-4B	0.0148	0.0097	0.0244	0.0127	0.0382	0.0162	0.0725	0.0229
	Gemma-3-12B	**0.0161**	0.0096	0.0251	0.0125	0.0410	0.0165	**0.0767**	0.0236		Gemma-3-12B	0.0146	0.0093	0.0248	0.0126	0.0384	0.0160	0.0697	0.0222
	Gemma-3-27B	0.0095	0.0060	0.0174	0.0086	0.0297	0.0117	0.0641	0.0184		Gemma-3-27B	0.0150	0.0095	0.0251	0.0127	0.0389	0.0162	0.0715	0.0227
	Qwen2.5-VL-7B	0.0131	0.0080	0.0248	0.0118	0.0397	0.0155	0.0750	0.0225		Qwen2.5-VL-7B	**0.0168**	**0.0112**	**0.0267**	**0.0144**	**0.0434**	**0.0185**	**0.0770**	**0.0252**
	Qwen2.5-VL-32B	0.0146	**0.0097**	**0.0262**	**0.0135**	0.0408	**0.0171**	0.0751	**0.0239**		Qwen2.5-VL-32B	0.0124	0.0078	0.0228	0.0111	0.0389	0.0152	0.0736	0.0220
	Qwen2.5-VL-72B	0.0151	0.0095	0.0255	0.0129	0.0403	0.0166	0.0758	0.0236		Qwen2.5-VL-72B	0.0144	0.0089	0.0244	0.0121	0.0395	0.0159	0.0743	0.0228
	Ground Truth	0.0157	0.0100	0.0248	0.0129	0.0418	0.0171	0.0770	0.0242		Ground Truth	0.0187	0.0122	0.0300	0.0159	0.0458	0.0199	0.0819	0.0270
Beauty	Gemma-3-4B	0.0072	0.0038	0.0109	0.0050	0.0195	0.0073	0.0488	0.0134	Scientific	Gemma-3-4B	0.0181	0.0113	0.0289	0.0148	0.0455	0.0190	**0.0880**	**0.0273**
	Gemma-3-12B	0.0041	0.0023	0.0081	0.0036	0.0173	0.0060	0.0425	0.0110		Gemma-3-12B	**0.0192**	**0.0126**	**0.0302**	**0.0161**	0.0445	0.0197	0.0817	0.0271
	Gemma-3-27B	0.0080	0.0050	0.0128	0.0066	0.0208	0.0088	0.0506	0.0149		Gemma-3-27B	0.0129	0.0080	0.0226	0.0111	0.0386	0.0151	0.0776	0.0228
	Qwen2.5-VL-7B	0.0066	0.0036	**0.0142**	0.0061	0.0248	0.0087	0.0557	0.0150		Qwen2.5-VL-7B	0.0140	0.0088	0.0240	0.0120	0.0379	0.0155	0.0764	0.0230
	Qwen2.5-VL-32B	0.0061	0.0037	0.0101	0.0051	**0.0283**	**0.0098**	0.0513	0.0145		Qwen2.5-VL-32B	0.0183	0.0111	0.0272	0.0140	0.0471	0.0190	0.0834	0.0262
	Qwen2.5-VL-72B	**0.0089**	**0.0054**	0.0133	**0.0069**	0.0198	0.0085	**0.0561**	**0.0160**		Qwen2.5-VL-72B	0.0169	0.0106	0.0292	0.0146	**0.0496**	**0.0197**	0.0832	0.0263
	Ground Truth	0.0076	0.0045	0.0178	0.0080	0.0247	0.0098	0.0510	0.0153		Ground Truth	0.0214	0.0137	0.0323	0.0172	0.0478	0.0211	0.0804	0.0275
Electronics	Gemma-3-4B	**0.0203**	**0.0135**	**0.0346**	**0.0182**	**0.0597**	**0.0248**	0.1090	**0.0349**	Toys & Games	Gemma-3-4B	0.0103	0.0063	0.0166	0.0083	0.0288	0.0114	0.0524	0.0161
	Gemma-3-12B	0.0185	0.0121	0.0328	0.0168	0.0563	0.0230	0.1048	0.0331		Gemma-3-12B	0.0091	0.0060	0.0155	0.0081	0.0267	0.0109	0.0483	0.0152
	Gemma-3-27B	0.0196	0.0129	0.0299	0.0163	0.0509	0.0218	0.0998	0.0317		Gemma-3-27B	**0.0118**	**0.0079**	**0.0185**	**0.0101**	0.0274	**0.0123**	0.0521	0.0172
	Qwen2.5-VL-7B	0.0171	0.0107	0.0324	0.0159	0.0574	0.0224	**0.1091**	0.0330		Qwen2.5-VL-7B	0.0094	0.0064	0.0152	0.0082	0.0264	0.0110	0.0474	0.0152
	Qwen2.5-VL-32B	0.0176	0.0111	0.0303	0.0155	0.0546	0.0218	0.1049	0.0322		Qwen2.5-VL-32B	0.0107	0.0065	0.0181	0.0089	0.0281	0.0114	0.0528	0.0162
	Qwen2.5-VL-72B	0.0165	0.0100	0.0296	0.0144	0.0518	0.0202	0.1036	0.0308		Qwen2.5-VL-72B	0.0105	0.0070	0.0182	0.0095	**0.0293**	0.0122	**0.0546**	**0.0172**
	Ground Truth	0.0203	0.0136	0.0329	0.0179	0.0551	0.0236	0.1088	0.0346		Ground Truth	0.0124	0.0080	0.0201	0.0104	0.0307	0.0131	0.0565	0.0181
Home	Gemma-3-4B	0.0100	0.0061	0.0177	0.0086	0.0310	0.0119	0.0622	0.0182	Video Games	Gemma-3-4B	0.0236	0.0154	0.0396	0.0206	0.0645	0.0270	0.1193	0.0379
	Gemma-3-12B	0.0096	0.0060	0.0161	0.0081	0.0265	0.0108	0.0530	0.0160		Gemma-3-12B	**0.0259**	0.0164	0.0410	0.0214	0.0663	0.0278	0.1196	0.0385
	Gemma-3-27B	**0.0123**	0.0078	**0.0190**	0.0100	**0.0327**	0.0135	**0.0675**	**0.0205**		Gemma-3-27B	0.0257	0.0168	0.0413	0.0219	0.0647	0.0278	0.1208	0.0390
	Qwen2.5-VL-7B	0.0110	**0.0079**	0.0182	**0.0103**	0.0317	**0.0137**	0.0603	0.0194		Qwen2.5-VL-7B	0.0241	0.0155	0.0405	0.0208	0.0644	0.0269	0.1155	0.0371
	Qwen2.5-VL-32B	0.0112	0.0071	0.0181	0.0094	0.0318	0.0129	0.0646	0.0194		Qwen2.5-VL-32B	0.0242	0.0152	0.0404	0.0205	0.0665	0.0272	0.1225	0.0384
	Qwen2.5-VL-72B	0.0093	0.0062	0.0178	0.0089	0.0289	0.0117	0.0579	0.0174		Qwen2.5-VL-72B	0.0255	**0.0169**	**0.0420**	**0.0222**	**0.0678**	**0.0288**	**0.1230**	**0.0398**
	Ground Truth	0.0109	0.0069	0.0169	0.0088	0.0277	0.0116	0.0592	0.0179		Ground Truth	0.0244	0.0158	0.0402	0.0209	0.0651	0.0273	0.1206	0.0384
Instruments	Gemma-3-4B	0.0206	0.0135	0.0333	0.0176	0.0524	0.0224	0.0939	0.0307										
	Gemma-3-12B	0.0199	0.0126	0.0324	0.0167	0.0520	0.0217	0.0922	0.0297										
	Gemma-3-27B	0.0199	0.0131	0.0324	0.0171	**0.0527**	0.0222	0.0939	0.0304										
	Qwen2.5-VL-7B	0.0189	0.0123	0.0307	0.0162	0.0498	0.0210	0.0916	0.0293										
	Qwen2.5-VL-32B	**0.0208**	**0.0141**	0.0336	**0.0182**	0.0513	**0.0228**	0.0923	**0.0309**										
	Qwen2.5-VL-72B	0.0197	0.0127	**0.0337**	0.0172	0.0524	0.0220	**0.0940**	0.0303										
	Ground Truth	0.0219	0.0146	0.0319	0.0179	0.0519	0.0229	0.0913	0.0308										

Table 8.15: Performance comparison for model BM3, combining Image-to-Text and Text-to-Image. R and N denote Recall and NDCG. All numeric entries are preserved exactly from the original six appendix tables.

(a) Image-to-Text

Datasets 1–5										Datasets 6–9									
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	**0.0560**	0.0344	0.0889	0.0449	0.1371	0.0571	0.2239	0.0743	Office	Gemma-3-4B	0.0545	0.0348	0.0853	0.0448	**0.1284**	0.0556	0.2054	0.0709
	Gemma-3-12B	0.0556	0.0342	0.0874	0.0445	0.1376	0.0572	0.2229	0.0741		Gemma-3-12B	0.0554	**0.0353**	**0.0854**	**0.0450**	0.1282	**0.0557**	0.2047	0.0709
	Gemma-3-27B	0.0554	0.0342	0.0873	0.0445	0.1374	0.0571	0.2253	0.0746		Gemma-3-27B	0.0549	0.0351	0.0846	0.0447	0.1277	0.0556	0.2045	0.0708
	Qwen2.5-VL-7B	0.0549	0.0338	0.0889	0.0448	0.1387	0.0574	0.2246	0.0744		Qwen2.5-VL-7B	0.0535	0.0341	0.0847	0.0441	0.1266	0.0547	0.2050	0.0702
	Qwen2.5-VL-32B	0.0558	**0.0346**	0.0881	**0.0450**	0.1386	**0.0577**	0.2264	**0.0751**		Qwen2.5-VL-32B	**0.0555**	0.0352	0.0852	0.0448	0.1273	0.0555	**0.2061**	**0.0711**
	Qwen2.5-VL-72B	0.0557	0.0340	**0.0890**	0.0447	**0.1391**	0.0574	**0.2265**	0.0746		Qwen2.5-VL-72B	0.0552	0.0352	0.0844	0.0447	0.1264	0.0553	0.2053	0.0710
	Ground Truth	0.0539	0.0334	0.0879	0.0443	0.1366	0.0566	0.2224	0.0737		Ground Truth	0.0544	0.0345	0.0848	0.0443	0.1278	0.0552	0.2048	0.0704
Beauty	Gemma-3-4B	0.0108	0.0070	**0.0173**	**0.0091**	**0.0241**	**0.0109**	**0.0460**	**0.0152**	Scientific	Gemma-3-4B	0.0434	0.0297	0.0738	0.0394	0.1099	0.0485	0.1726	0.0609
	Gemma-3-12B	0.0112	**0.0077**	0.0145	0.0088	0.0197	0.0101	0.0376	0.0137		Gemma-3-12B	0.0441	**0.0303**	**0.0746**	**0.0400**	0.1089	0.0487	0.1734	**0.0614**
	Gemma-3-27B	0.0112	0.0068	0.0126	0.0074	0.0211	0.0095	0.0378	0.0129		Gemma-3-27B	**0.0446**	0.0301	0.0737	0.0394	0.1083	0.0481	0.1722	0.0607
	Qwen2.5-VL-7B	**0.0112**	0.0065	0.0145	0.0075	0.0220	0.0094	0.0340	0.0119		Qwen2.5-VL-7B	0.0436	0.0298	0.0729	0.0392	0.1095	0.0484	0.1727	0.0609
	Qwen2.5-VL-32B	0.0112	0.0077	0.0126	0.0082	0.0198	0.0100	0.0359	0.0134		Qwen2.5-VL-32B	0.0438	0.0300	0.0727	0.0393	**0.1102**	**0.0488**	**0.1741**	0.0614
	Qwen2.5-VL-72B	0.0103	0.0064	0.0126	0.0072	0.0197	0.0091	0.0359	0.0123		Qwen2.5-VL-72B	0.0434	0.0295	0.0725	0.0388	0.1096	0.0482	0.1734	0.0608
	Ground Truth	0.0104	0.0074	0.0123	0.0079	0.0198	0.0098	0.0363	0.0132		Ground Truth	0.0445	0.0304	0.0737	0.0398	0.1089	0.0487	0.1746	0.0616
Electronics	Gemma-3-4B	0.0236	0.0142	**0.0406**	0.0199	0.0711	0.0278	**0.1324**	**0.0403**	Toys & Games	Gemma-3-4B	0.0342	0.0232	0.0511	0.0286	0.0759	0.0349	**0.1408**	**0.0476**
	Gemma-3-12B	0.0239	0.0143	0.0404	0.0198	**0.0717**	**0.0280**	0.1311	0.0401		Gemma-3-12B	0.0335	0.0225	0.0498	0.0278	0.0754	0.0342	0.1379	0.0465
	Gemma-3-27B	**0.0239**	0.0143	0.0403	0.0198	0.0713	0.0279	0.1316	0.0401		Gemma-3-27B	**0.0348**	**0.0233**	0.0513	0.0286	**0.0767**	**0.0349**	0.1398	0.0473
	Qwen2.5-VL-7B	0.0238	**0.0143**	0.0404	**0.0199**	0.0701	0.0276	0.1312	0.0401		Qwen2.5-VL-7B	0.0339	0.0228	0.0514	0.0284	0.0758	0.0345	0.1385	0.0469
	Qwen2.5-VL-32B	0.0238	0.0143	0.0397	0.0196	0.0709	0.0277	0.1301	0.0398		Qwen2.5-VL-32B	0.0343	0.0231	0.0510	0.0285	0.0757	0.0348	0.1395	0.0473
	Qwen2.5-VL-72B	0.0237	0.0143	0.0399	0.0197	0.0703	0.0277	0.1314	0.0401		Qwen2.5-VL-72B	0.0337	0.0228	**0.0520**	**0.0287**	0.0756	0.0346	0.1391	0.0471
	Ground Truth	0.0238	0.0145	0.0420	0.0206	0.0724	0.0285	0.1328	0.0408		Ground Truth	0.0351	0.0233	0.0520	0.0287	0.0765	0.0349	0.1406	0.0476
Home	Gemma-3-4B	0.0294	0.0185	0.0521	0.0258	0.0818	0.0332	0.1531	0.0474	Video Games	Gemma-3-4B	0.0582	0.0384	0.0933	0.0497	0.1425	0.0622	0.2352	0.0807
	Gemma-3-12B	0.0296	0.0186	**0.0523**	0.0259	0.0818	0.0332	0.1536	0.0475		Gemma-3-12B	0.0578	0.0382	0.0940	0.0499	0.1432	0.0624	0.2344	0.0806
	Gemma-3-27B	0.0293	0.0185	0.0507	0.0254	0.0820	0.0332	0.1528	0.0473		Gemma-3-27B	0.0586	0.0385	0.0938	0.0498	0.1427	0.0623	0.2347	0.0806
	Qwen2.5-VL-7B	**0.0303**	**0.0198**	0.0504	**0.0262**	**0.0825**	**0.0344**	**0.1540**	**0.0485**		Qwen2.5-VL-7B	**0.0590**	**0.0391**	**0.0945**	**0.0505**	**0.1436**	**0.0630**	**0.2360**	**0.0814**
	Qwen2.5-VL-32B	0.0294	0.0185	0.0512	0.0255	0.0823	0.0334	0.1533	0.0474		Qwen2.5-VL-32B	0.0580	0.0385	0.0940	0.0501	0.1421	0.0624	0.2354	0.0809
	Qwen2.5-VL-72B	0.0293	0.0185	0.0507	0.0254	0.0825	0.0334	0.1528	0.0473		Qwen2.5-VL-72B	0.0586	0.0386	0.0939	0.0500	0.1422	0.0623	0.2345	0.0807
	Ground Truth	0.0293	0.0185	0.0521	0.0258	0.0815	0.0332	0.1528	0.0473		Ground Truth	0.0594	0.0391	0.0944	0.0504	0.1434	0.0629	0.2362	0.0814
Instruments	Gemma-3-4B	0.0624	0.0398	0.0941	0.0500	0.1428	0.0623	0.2411	0.0818										
	Gemma-3-12B	0.0623	0.0408	**0.0953**	0.0514	**0.1437**	0.0637	0.2432	0.0835										
	Gemma-3-27B	0.0621	0.0401	0.0949	0.0507	0.1420	0.0627	0.2427	0.0827										
	Qwen2.5-VL-7B	0.0621	0.0404	0.0952	0.0510	0.1418	0.0629	0.2434	0.0830										
	Qwen2.5-VL-32B	0.0618	0.0401	0.0947	0.0508	0.1428	0.0630	**0.2454**	0.0834										
	Qwen2.5-VL-72B	**0.0630**	**0.0412**	0.0951	**0.0515**	0.1431	**0.0638**	0.2451	**0.0840**										
	Ground Truth	0.0630	0.0408	0.0955	0.0512	0.1420	0.0631	0.2437	0.0833										

(b) Text-to-Image

Datasets 1–5										Datasets 6–9									
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	0.0559	0.0348	0.0881	0.0452	0.1402	0.0584	**0.2271**	0.0756	Office	Gemma-3-4B	0.0549	0.0349	**0.0870**	0.0453	0.1284	0.0557	0.2041	0.0708
	Gemma-3-12B	0.0562	**0.0359**	0.0903	**0.0469**	0.1403	**0.0595**	0.2226	**0.0759**		Gemma-3-12B	0.0542	0.0343	0.0854	0.0444	0.1285	0.0553	0.2039	0.0703
	Gemma-3-27B	0.0561	0.0350	0.0908	0.0462	0.1395	0.0585	0.2205	0.0747		Gemma-3-27B	0.0539	0.0343	0.0868	0.0449	0.1269	0.0551	0.2039	0.0704
	Qwen2.5-VL-7B	**0.0566**	0.0355	0.0902	0.0463	0.1403	0.0590	0.2213	0.0752		Qwen2.5-VL-7B	0.0547	0.0347	0.0861	0.0449	**0.1304**	0.0561	**0.2057**	0.0710
	Qwen2.5-VL-32B	0.0560	0.0344	**0.0920**	0.0460	**0.1405**	0.0582	0.2260	0.0752		Qwen2.5-VL-32B	**0.0554**	**0.0354**	0.0865	**0.0455**	0.1298	**0.0564**	0.2049	**0.0712**
	Qwen2.5-VL-72B	0.0549	0.0344	0.0893	0.0455	0.1392	0.0581	0.2239	0.0750		Qwen2.5-VL-72B	0.0552	0.0347	0.0864	0.0448	0.1291	0.0555	0.2031	0.0702
	Ground Truth	0.0539	0.0334	0.0879	0.0443	0.1366	0.0566	0.2224	0.0737		Ground Truth	0.0544	0.0345	0.0848	0.0443	0.1278	0.0552	0.2048	0.0704
Beauty	Gemma-3-4B	0.0103	0.0066	0.0131	0.0075	0.0222	0.0098	**0.0432**	**0.0140**	Scientific	Gemma-3-4B	0.0455	0.0303	0.0737	0.0393	0.1087	0.0483	0.1760	0.0615
	Gemma-3-12B	0.0104	**0.0075**	0.0151	**0.0089**	0.0202	**0.0102**	0.0345	0.0131		Gemma-3-12B	**0.0476**	**0.0321**	0.0704	0.0394	0.1041	0.0479	0.1697	0.0609
	Gemma-3-27B	0.0095	0.0071	0.0141	0.0084	0.0204	0.0100	0.0349	0.0129		Gemma-3-27B	0.0466	0.0309	0.0731	0.0395	0.1097	0.0487	**0.1799**	0.0625
	Qwen2.5-VL-7B	0.0103	0.0063	0.0123	0.0069	0.0212	0.0092	0.0422	0.0135		Qwen2.5-VL-7B	0.0472	0.0317	**0.0752**	**0.0406**	**0.1116**	0.0498	0.1769	**0.0627**
	Qwen2.5-VL-32B	**0.0122**	0.0070	0.0146	0.0078	**0.0230**	0.0099	0.0404	0.0134		Qwen2.5-VL-32B	0.0451	0.0303	0.0734	0.0394	0.1064	0.0478	0.1736	0.0610
	Qwen2.5-VL-72B	0.0103	0.0061	**0.0151**	0.0077	0.0221	0.0094	0.0408	0.0131		Qwen2.5-VL-72B	0.0463	0.0316	0.0738	0.0404	0.1113	**0.0499**	0.1759	0.0626
	Ground Truth	0.0104	0.0074	0.0123	0.0079	0.0198	0.0098	0.0363	0.0132		Ground Truth	0.0445	0.0304	0.0737	0.0398	0.1089	0.0487	0.1746	0.0616
Electronics	Gemma-3-4B	0.0242	0.0148	0.0429	0.0211	0.0727	0.0289	0.1334	0.0413	Toys & Games	Gemma-3-4B	0.0332	0.0223	0.0512	0.0281	0.0772	0.0346	0.1392	0.0468
	Gemma-3-12B	0.0243	0.0148	**0.0431**	0.0211	0.0727	0.0289	0.1341	0.0414		Gemma-3-12B	**0.0340**	**0.0227**	**0.0523**	**0.0286**	0.0775	**0.0349**	0.1391	**0.0470**
	Gemma-3-27B	0.0243	0.0147	0.0430	0.0210	0.0716	0.0285	0.1332	0.0411		Gemma-3-27B	0.0327	0.0218	0.0512	0.0277	**0.0784**	0.0345	0.1397	0.0466
	Qwen2.5-VL-7B	**0.0250**	**0.0151**	0.0430	**0.0211**	0.0727	**0.0289**	**0.1344**	**0.0415**		Qwen2.5-VL-7B	0.0339	0.0224	0.0514	0.0280	0.0759	0.0342	**0.1404**	0.0469
	Qwen2.5-VL-32B	0.0242	0.0147	0.0366	0.0188	0.0730	0.0283	0.1290	0.0396		Qwen2.5-VL-32B	0.0331	0.0224	0.0516	0.0284	0.0768	0.0347	0.1386	0.0469
	Qwen2.5-VL-72B	0.0243	0.0148	0.0427	0.0210	**0.0731**	0.0289	0.1340	0.0413		Qwen2.5-VL-72B	0.0334	0.0224	0.0507	0.0280	0.0781	0.0348	0.1390	0.0468
	Ground Truth	0.0238	0.0145	0.0420	0.0206	0.0724	0.0285	0.1328	0.0408		Ground Truth	0.0351	0.0233	0.0520	0.0287	0.0765	0.0349	0.1406	0.0476
Home	Gemma-3-4B	0.0299	0.0205	0.0489	0.0266	0.0817	0.0349	0.1507	0.0487	Video Games	Gemma-3-4B	0.0615	0.0400	0.0970	0.0515	0.1462	0.0640	0.2369	0.0821
	Gemma-3-12B	0.0299	**0.0210**	0.0491	**0.0272**	**0.0826**	**0.0357**	**0.1516**	**0.0494**		Gemma-3-12B	**0.0623**	**0.0400**	0.0981	0.0516	0.1474	0.0641	0.2408	0.0828
	Gemma-3-27B	0.0299	0.0202	0.0484	0.0261	0.0812	0.0345	0.1503	0.0483		Gemma-3-27B	0.0615	0.0398	**0.0982**	**0.0517**	**0.1475**	**0.0642**	**0.2430**	**0.0833**
	Qwen2.5-VL-7B	0.0296	0.0203	0.0491	0.0266	0.0808	0.0347	0.1505	0.0486		Qwen2.5-VL-7B	0.0608	0.0399	0.0963	0.0513	0.1456	0.0639	0.2388	0.0825
	Qwen2.5-VL-32B	0.0296	0.0203	0.0490	0.0266	0.0812	0.0348	0.1511	0.0488		Qwen2.5-VL-32B	0.0602	0.0395	0.0972	0.0515	0.1450	0.0636	0.2370	0.0820
	Qwen2.5-VL-72B	**0.0301**	0.0202	**0.0492**	0.0263	0.0805	0.0343	0.1509	0.0484		Qwen2.5-VL-72B	0.0574	0.0377	0.0939	0.0495	0.1424	0.0618	0.2369	0.0807
	Ground Truth	0.0293	0.0185	0.0521	0.0258	0.0815	0.0332	0.1528	0.0473		Ground Truth	0.0594	0.0391	0.0944	0.0504	0.1434	0.0629	0.2362	0.0814
Instruments	Gemma-3-4B	**0.0654**	**0.0420**	**0.0981**	**0.0525**	0.1449	**0.0644**	0.2482	**0.0849**										
	Gemma-3-12B	0.0639	0.0414	0.0967	0.0520	0.1446	0.0641	0.2466	0.0844										
	Gemma-3-27B	0.0635	0.0413	0.0971	0.0521	0.1451	0.0642	0.2479	0.0846										
	Qwen2.5-VL-7B	0.0631	0.0407	0.0972	0.0517	0.1447	0.0637	0.2486	0.0844										
	Qwen2.5-VL-32B	0.0640	0.0414	0.0949	0.0513	0.1449	0.0640	0.2464	0.0841										
	Qwen2.5-VL-72B	0.0630	0.0405	0.0980	0.0518	**0.1458**	0.0640	**0.2487**	0.0845										
	Ground Truth	0.0630	0.0408	0.0955	0.0512	0.1420	0.0631	0.2437	0.0833										

Table 8.16: Performance comparison for model FREEDOM, combining Image-to-Text and Text-to-Image. R and N denote Recall and NDCG. All numeric entries are preserved exactly from the original six appendix tables.

(a) Image-to-Text

Datasets 1–5										Datasets 6–9									
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	0.0501	0.0323	0.0880	0.0445	0.1424	0.0583	0.2301	0.0757	Office	Gemma-3-4B	0.0509	0.0328	0.0818	**0.0428**	**0.1251**	**0.0537**	0.2007	0.0687
	Gemma-3-12B	0.0522	0.0332	0.0871	0.0445	0.1372	0.0570	0.2316	0.0758		Gemma-3-12B	0.0509	0.0325	**0.0824**	0.0427	0.1245	0.0533	**0.2044**	**0.0691**
	Gemma-3-27B	**0.0533**	0.0334	**0.0893**	0.0450	**0.1444**	**0.0589**	0.2307	0.0761		Gemma-3-27B	**0.0514**	**0.0328**	0.0804	0.0422	0.1246	0.0533	0.2028	0.0688
	Qwen2.5-VL-7B	0.0515	0.0332	0.0886	**0.0450**	0.1370	0.0571	0.2287	0.0755		Qwen2.5-VL-7B	0.0507	0.0319	0.0813	0.0417	0.1250	0.0528	0.1997	0.0676
	Qwen2.5-VL-32B	0.0493	0.0316	0.0859	0.0434	0.1364	0.0562	0.2290	0.0746		Qwen2.5-VL-32B	0.0497	0.0319	0.0810	0.0420	0.1245	0.0530	0.2017	0.0683
	Qwen2.5-VL-72B	0.0521	**0.0336**	0.0845	0.0440	0.1416	0.0585	**0.2329**	**0.0766**		Qwen2.5-VL-72B	0.0497	0.0318	0.0800	0.0415	0.1233	0.0524	0.2009	0.0678
	Ground Truth	0.0545	0.0349	0.0912	0.0467	0.1446	0.0603	0.2341	0.0782		Ground Truth	0.0533	0.0342	0.0859	0.0448	0.1279	0.0554	0.2065	0.0709
Beauty	Gemma-3-4B	0.0140	0.0100	0.0223	0.0126	0.0378	0.0167	0.0614	0.0213	Scientific	Gemma-3-4B	0.0515	0.0342	0.0798	0.0434	0.1178	0.0529	0.1796	0.0652
	Gemma-3-12B	0.0159	0.0099	0.0265	0.0134	0.0358	0.0158	0.0716	0.0232		Gemma-3-12B	0.0509	0.0333	0.0761	0.0415	0.1155	0.0514	0.1844	0.0651
	Gemma-3-27B	0.0176	0.0113	0.0265	0.0143	0.0409	0.0179	**0.0776**	**0.0255**		Gemma-3-27B	0.0515	0.0345	0.0793	**0.0436**	0.1158	0.0528	0.1840	**0.0664**
	Qwen2.5-VL-7B	**0.0197**	**0.0135**	**0.0331**	**0.0180**	**0.0421**	**0.0203**	0.0575	0.0236		Qwen2.5-VL-7B	**0.0525**	**0.0347**	0.0778	0.0429	0.1198	**0.0534**	0.1811	0.0655
	Qwen2.5-VL-32B	0.0187	0.0117	0.0256	0.0139	0.0366	0.0168	0.0609	0.0218		Qwen2.5-VL-32B	0.0514	0.0332	0.0767	0.0414	0.1143	0.0507	0.1817	0.0641
	Qwen2.5-VL-72B	0.0154	0.0101	0.0300	0.0147	0.0417	0.0177	0.0652	0.0225		Qwen2.5-VL-72B	0.0512	0.0330	**0.0799**	0.0423	**0.1204**	0.0525	**0.1864**	0.0656
	Ground Truth	0.0187	0.0106	0.0314	0.0147	0.0487	0.0192	0.0844	0.0264		Ground Truth	0.0541	0.0348	0.0788	0.0428	0.1180	0.0527	0.1908	0.0672
Electronics	Gemma-3-4B	**0.0319**	**0.0198**	0.0456	**0.0243**	0.0792	**0.0332**	0.1533	0.0483	Toys & Games	Gemma-3-4B	0.0355	0.0236	0.0534	0.0294	0.0797	0.0360	0.1356	0.0470
	Gemma-3-12B	0.0269	0.0163	**0.0475**	0.0232	0.0790	0.0314	0.1565	0.0472		Gemma-3-12B	0.0351	0.0235	0.0543	0.0297	0.0817	0.0366	0.1425	0.0486
	Gemma-3-27B	0.0238	0.0147	0.0389	0.0197	0.0729	0.0286	0.1403	0.0424		Gemma-3-27B	**0.0374**	**0.0251**	**0.0558**	**0.0310**	**0.0844**	**0.0382**	**0.1434**	**0.0499**
	Qwen2.5-VL-7B	0.0231	0.0150	0.0407	0.0207	0.0802	0.0309	**0.1579**	0.0467		Qwen2.5-VL-7B	0.0365	0.0247	0.0532	0.0301	0.0796	0.0367	0.1383	0.0483
	Qwen2.5-VL-32B	0.0229	0.0149	0.0410	0.0209	0.0791	0.0309	0.1464	0.0447		Qwen2.5-VL-32B	0.0352	0.0234	0.0544	0.0296	0.0800	0.0360	0.1408	0.0479
	Qwen2.5-VL-72B	0.0286	0.0181	0.0464	0.0239	**0.0810**	0.0330	0.1564	**0.0486**		Qwen2.5-VL-72B	0.0364	0.0239	0.0547	0.0298	0.0808	0.0364	0.1352	0.0471
	Ground Truth	0.0309	0.0191	0.0541	0.0267	0.0885	0.0358	0.1543	0.0492		Ground Truth	0.0376	0.0255	0.0559	0.0314	0.0851	0.0388	0.1452	0.0506
Home	Gemma-3-4B	0.0314	0.0214	0.0494	0.0273	0.0838	0.0360	0.1511	0.0495	Video Games	Gemma-3-4B	**0.0638**	**0.0417**	0.0997	0.0533	0.1528	0.0668	**0.2530**	0.0869
	Gemma-3-12B	0.0320	0.0217	0.0505	0.0277	0.0823	0.0358	0.1537	0.0501		Gemma-3-12B	0.0625	0.0409	0.0996	0.0530	0.1524	0.0664	0.2505	0.0860
	Gemma-3-27B	0.0323	0.0219	0.0495	0.0275	**0.0839**	0.0362	0.1528	0.0500		Gemma-3-27B	0.0624	0.0413	0.0995	**0.0533**	**0.1550**	**0.0675**	0.2528	**0.0870**
	Qwen2.5-VL-7B	0.0322	0.0219	0.0501	0.0276	0.0833	0.0361	**0.1581**	**0.0510**		Qwen2.5-VL-7B	0.0615	0.0403	0.1000	0.0528	0.1549	0.0667	0.2496	0.0857
	Qwen2.5-VL-32B	0.0327	0.0222	**0.0523**	**0.0286**	0.0839	**0.0366**	0.1534	0.0505		Qwen2.5-VL-32B	0.0622	0.0403	0.0979	0.0518	0.1538	0.0660	0.2516	0.0856
	Qwen2.5-VL-72B	**0.0327**	**0.0223**	0.0515	0.0285	0.0833	0.0365	0.1528	0.0505		Qwen2.5-VL-72B	0.0614	0.0402	**0.1005**	0.0528	0.1526	0.0661	0.2514	0.0859
	Ground Truth	0.0312	0.0215	0.0500	0.0276	0.0803	0.0353	0.1544	0.0501		Ground Truth	0.0623	0.0404	0.0985	0.0521	0.1523	0.0658	0.2508	0.0855
Instruments	Gemma-3-4B	0.0651	**0.0426**	**1.0124**	**0.0547**	0.1503	**0.0668**	0.2515	**0.0870**										
	Gemma-3-12B	**0.0656**	0.0417	0.1016	0.0534	0.1502	0.0657	0.2504	0.0856										
	Gemma-3-27B	0.0634	0.0410	0.1012	0.0531	0.1504	0.0656	0.2504	0.0854										
	Qwen2.5-VL-7B	0.0635	0.0415	0.1009	0.0537	**0.1515**	0.0665	**0.2519**	0.0865										
	Qwen2.5-VL-32B	0.0640	0.0416	0.1020	0.0539	0.1497	0.0660	0.2506	0.0861										
	Qwen2.5-VL-72B	0.0653	0.0424	0.1004	0.0537	0.1508	0.0665	0.2513	0.0865										
	Ground Truth	0.0651	0.0425	0.1031	0.0548	0.1539	0.0676	0.2544	0.0877										

(b) Text-to-Image

Datasets 1–5										Datasets 6–9									
Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50	Dataset	Completion Method	R@5	N@5	R@10	N@10	R@20	N@20	R@50	N@50
Arts	Gemma-3-4B	0.0551	0.0346	0.0919	0.0464	0.1445	0.0597	0.2348	0.0776	Office	Gemma-3-4B	0.0525	0.0335	0.0844	0.0438	0.1265	0.0545	0.2072	0.0704
	Gemma-3-12B	**0.0561**	0.0349	0.0902	0.0459	0.1440	0.0595	0.2352	0.0777		Gemma-3-12B	0.0535	0.0342	0.0859	0.0447	0.1269	0.0551	0.2070	0.0709
	Gemma-3-27B	0.0547	0.0348	0.0923	**0.0470**	**0.1458**	**0.0605**	**0.2366**	**0.0785**		Gemma-3-27B	0.0533	0.0341	**0.0862**	0.0447	0.1278	0.0552	0.2073	0.0709
	Qwen2.5-VL-7B	0.0547	0.0341	0.0916	0.0459	0.1450	0.0594	0.2363	0.0776		Qwen2.5-VL-7B	0.0533	0.0340	0.0859	0.0445	0.1269	0.0549	0.2064	0.0706
	Qwen2.5-VL-32B	0.0554	0.0348	**0.0926**	0.0467	0.1445	0.0598	0.2352	0.0779		Qwen2.5-VL-32B	**0.0538**	**0.0344**	0.0859	0.0447	0.1273	0.0552	0.2061	0.0708
	Qwen2.5-VL-72B	0.0554	**0.0350**	0.0919	0.0467	0.1451	0.0601	0.2360	0.0782		Qwen2.5-VL-72B	0.0529	0.0340	0.0860	**0.0447**	**0.1281**	**0.0553**	**0.2083**	**0.0711**
	Ground Truth	0.0545	0.0349	0.0912	0.0467	0.1446	0.0603	0.2341	0.0782		Ground Truth	0.0533	0.0342	0.0859	0.0448	0.1279	0.0554	0.2065	0.0709
Beauty	Gemma-3-4B	**0.0206**	0.0110	0.0295	0.0139	**0.0523**	0.0198	**0.0929**	0.0279	Scientific	Gemma-3-4B	**0.0532**	**0.0345**	0.0781	0.0426	0.1167	0.0523	**0.1913**	**0.0671**
	Gemma-3-12B	0.0164	0.0094	0.0318	0.0144	0.0476	0.0186	0.0825	0.0254		Gemma-3-12B	0.0518	0.0344	0.0771	0.0425	**0.1186**	**0.0530**	0.1887	0.0669
	Gemma-3-27B	0.0178	0.0097	0.0318	0.0144	0.0476	0.0185	0.0813	0.0252		Gemma-3-27B	0.0493	0.0327	**0.0814**	**0.0430**	0.1184	0.0522	0.1898	0.0664
	Qwen2.5-VL-7B	0.0197	0.0112	0.0304	0.0147	0.0499	0.0198	0.0904	**0.0280**		Qwen2.5-VL-7B	0.0519	0.0336	0.0804	0.0428	0.1178	0.0522	0.1905	0.0666
	Qwen2.5-VL-32B	0.0183	0.0104	0.0304	0.0144	0.0457	0.0186	0.0825	0.0259		Qwen2.5-VL-32B	0.0521	0.0344	0.0786	0.0429	0.1178	0.0528	0.1862	0.0664
	Qwen2.5-VL-72B	0.0201	**0.0121**	**0.0337**	**0.0165**	0.0467	**0.0200**	0.0789	0.0265		Qwen2.5-VL-72B	0.0498	0.0329	0.0795	0.0426	0.1182	0.0523	0.1911	0.0667
	Ground Truth	0.0187	0.0106	0.0314	0.0147	0.0487	0.0192	0.0844	0.0264		Ground Truth	0.0541	0.0348	0.0788	0.0428	0.1180	0.0527	0.1908	0.0672
Electronics	Gemma-3-4B	**0.0321**	**0.0195**	0.0539	0.0267	**0.0911**	**0.0364**	0.1545	0.0493	Toys & Games	Gemma-3-4B	0.0373	0.0252	0.0556	0.0311	**0.0854**	0.0386	**0.1463**	**0.0506**
	Gemma-3-12B	0.0302	0.0186	0.0544	0.0265	0.0893	0.0357	**0.1553**	0.0492		Gemma-3-12B	0.0378	0.0252	0.0559	0.0311	0.0847	0.0384	0.1426	0.0499
	Gemma-3-27B	0.0315	0.0192	0.0528	0.0263	0.0880	0.0355	0.1523	0.0487		Gemma-3-27B	0.0371	0.0251	**0.0571**	**0.0315**	0.0851	0.0385	0.1451	0.0504
	Qwen2.5-VL-7B	0.0314	0.0192	0.0544	0.0268	0.0884	0.0357	0.1538	0.0491		Qwen2.5-VL-7B	0.0376	0.0253	0.0557	0.0312	0.0851	**0.0386**	0.1451	0.0504
	Qwen2.5-VL-32B	0.0315	0.0194	**0.0547**	**0.0270**	0.0904	0.0364	0.1535	0.0493		Qwen2.5-VL-32B	**0.0382**	**0.0255**	0.0553	0.0310	0.0843	0.0383	0.1448	0.0503
	Qwen2.5-VL-72B	0.0315	0.0193	0.0545	0.0269	0.0897	0.0361	0.1547	**0.0494**		Qwen2.5-VL-72B	0.0373	0.0253	0.0558	0.0312	0.0848	0.0385	0.1453	0.0505
	Ground Truth	0.0309	0.0191	0.0541	0.0267	0.0885	0.0358	0.1543	0.0492		Ground Truth	0.0376	0.0255	0.0559	0.0314	0.0851	0.0388	0.1452	0.0506
Home	Gemma-3-4B	0.0307	0.0215	0.0486	0.0273	0.0843	**0.0363**	0.1540	**0.0502**	Video Games	Gemma-3-4B	0.0621	0.0406	0.0990	0.0525	**0.1541**	**0.0666**	0.2519	0.0861
	Gemma-3-12B	**0.0327**	**0.0225**	**0.0505**	**0.0282**	0.0806	0.0358	0.1518	0.0501		Gemma-3-12B	0.0605	0.0392	0.0968	0.0510	0.1514	0.0648	0.2497	0.0844
	Gemma-3-27B	0.0319	0.0217	0.0503	0.0277	0.0825	0.0358	0.1526	0.0498		Gemma-3-27B	0.0623	0.0403	0.0991	0.0522	0.1521	0.0657	0.2532	0.0859
	Qwen2.5-VL-7B	0.0303	0.0210	0.0480	0.0268	0.0835	0.0358	0.1540	0.0499		Qwen2.5-VL-7B	0.0608	0.0394	0.0966	0.0510	0.1507	0.0647	0.2507	0.0847
	Qwen2.5-VL-32B	0.0309	0.0214	0.0476	0.0269	0.0832	0.0359	0.1537	0.0500		Qwen2.5-VL-32B	0.0620	0.0404	**0.1001**	**0.0528**	0.1539	0.0664	**0.2535**	**0.0863**
	Qwen2.5-VL-72B	0.0307	0.0213	0.0474	0.0267	**0.0844**	0.0361	**0.1546**	0.0501		Qwen2.5-VL-72B	**0.0630**	**0.0408**	0.0999	0.0527	0.1534	0.0663	0.2518	0.0859
	Ground Truth	0.0312	0.0215	0.0500	0.0276	0.0803	0.0353	0.1544	0.0501		Ground Truth	0.0623	0.0404	0.0985	0.0521	0.1523	0.0658	0.2508	0.0855
Instruments	Gemma-3-4B	0.0651	0.0423	0.1041	0.0548	0.1540	0.0675	0.2535	0.0873										
	Gemma-3-12B	0.0650	0.0423	**0.1052**	**0.0553**	0.1544	0.0677	0.2521	0.0872										
	Gemma-3-27B	**0.0659**	0.0422	0.1049	0.0548	0.1552	0.0675	**0.2551**	**0.0874**										
	Qwen2.5-VL-7B	0.0645	0.0419	0.1041	0.0548	**0.1552**	**0.0677**	0.2528	0.0871										
	Qwen2.5-VL-32B	0.0655	**0.0424**	0.1047	0.0550	0.1541	0.0675	0.2537	0.0874										
	Qwen2.5-VL-72B	0.0653	0.0423	0.1028	0.0544	0.1531	0.0672	0.2533	0.0872										
	Ground Truth	0.0651	0.0425	0.1031	0.0548	0.1539	0.0676	0.2544	0.0877										

Chapter 9

LLMPopcorn: Popularity-Oriented Micro-Video Generation

Publication Note

This chapter is adapted from: Junchen Fu, Xuri Ge, Kaiwen Zheng, Alexandros Karatzoglou, Ioannis Arapakis, Xin Xin, Yongxin Ni, and Joemon M. Jose. LLMPopcorn: Exploring LLMs as Assistants for Popular Micro-video Generation. ICASSP 2026.

Chapter Positioning

This chapter completes the thesis by moving from platform-side product completion to user/content-side generation. LLMPopcorn studies how LLMs can assist the creation of micro-videos that are more likely to attract engagement, making recommendation-oriented adaptation visible at the content-production end of the system.

9.1 Introduction

Micro-videos (or short videos) have emerged as a crucial element of the digital economy, representing a multi-billion-dollar industry (J. Guo et al., 2024). They have become an integral part of daily life for people worldwide, providing substantial commercial value for social media platforms and content creators. Popular content creators can receive significant revenue through their content (Ran et al., 2022), which underscores the ever-growing influence of micro-videos in modern society.

Despite their widespread popularity and financial impact, producing popular micro-video content remains a costly and labour-intensive process. Professional filming, scripting, and editing require significant time and resources, which not all creators or businesses can afford. Driven

by these challenges, we explore simplifying the micro-video creation process using state-of-the-art AI solutions, with a particular focus on generating popular micro-videos.

On one hand, the rapid advancement of Large Language Models (LLMs) like ChatGPT has unlocked new possibilities for content generation, demonstrating strong capabilities in tasks such as document summarisation (Jin, Zhang, Meng, Wang, & Tan, 2024), programming (Y. Cai et al., 2023), and recommendation (Fu, Ge, Xin, Karatzoglou, et al., 2025). On the other hand, video generation models powered by diffusion techniques and neural rendering are transforming creative content production, enabling high-quality synthesis for applications in film production and interactive media (Y. Liu, Zhang, Li, et al., 2024). Significant breakthroughs in LLMs and video generation models have prompted many studies (P. Zhou et al., 2024) to integrate LLMs to facilitate automatic video generation. For instance, VideoLLMs like VideoPoet (Kondratyuk et al., 2023) and GPT4Video (Z. Wang et al., 2024) leverage LLMs and video generation models for multimodal video understanding and generation.

Current studies on LLMs for video generation overlook their role in creating popular micro-videos, focusing instead on quality metrics like resolution and duration. Recent research, including models like those in (J. Lu et al., 2024) and datasets like MicroLens (Y. Ni et al., 2023), emphasises micro-video popularity prediction. However, the use of mainstream LLMs to generate highly popular micro-videos, leveraging their text generation and planning capabilities (W. Wu, Zhu, & Shou, 2025), remains underexplored.

Driven by the above insights, we present an initial exploration of *whether LLMs can assist in the **popular** micro-video creation.* To address this research question systematically, we propose and investigate the following three Research Questions (RQs):

RQ9.1: *How can LLMs be effectively utilised to assist popular micro-video generation?*
We present LLMPopcorn, a micro-video creation pipeline that uses LLMs to generate prompts for popular video production. To assess popularity, we integrate a state-of-the-art offline evaluator and compare three mainstream LLMs against human-created videos.

RQ9.2: *To what extent can prompt-based enhancements optimise the content generated by LLMs for achieving greater popularity?*
We propose a prompt enhancement method that combines Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) and chain-of-thought (CoT) prompting (J. Wei et al., 2022), inspired by human content creation.

RQ9.3: *How do various LLMs and video generators perform on popular micro-video generation?*
We compare five LLMs and three video generation models using the LLMPopcorn pipeline.

By investigating these research questions, this study represents an initial exploration of this emerging domain, providing key insights into AI-assisted micro-video creation.

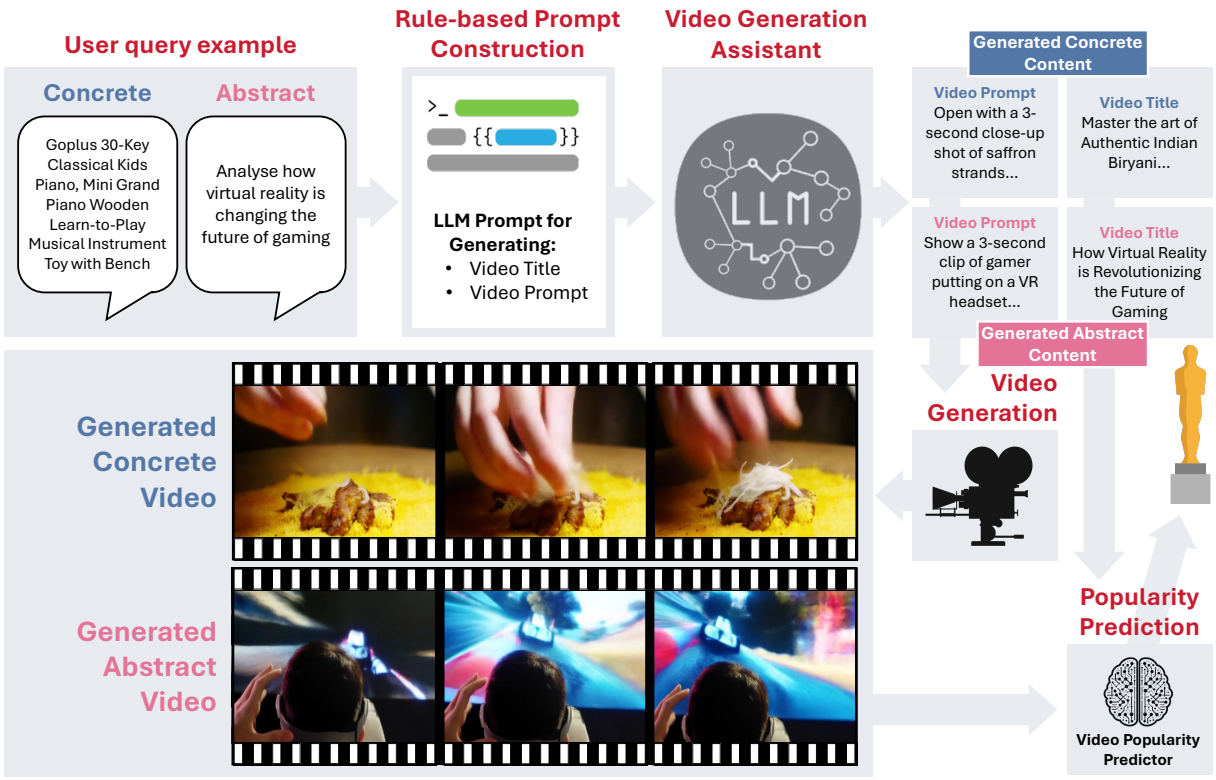


Figure 9.1: Overview of the LLMPopcorn pipeline. An LLM processes a user query, generating a video title and prompt for a video generator to create a micro-video. The video and title are then evaluated for predicted popularity.

9.2 The Proposed *LLMPopcorn*

9.2.1 Problem Formulation

We consider a problem in which a user provides an input prompt $x \in \mathscr{X}$. An LLM processes this prompt and generates both a video title and a video prompt, which we denote by $(t, p) = f(x)$, where t belongs to the set of titles $\mathscr{T}$ and p belongs to the set of video prompts $\mathscr{P}$. Next, a pre-trained video generation model g uses the generated prompt to produce an AI-created micro-video $v \in \mathscr{V}$, that is, $v = g(p)$. An offline video popularity predictor h then assigns a popularity score $s \in \mathbb{R}$ to the generated video, so that $s = h(v,t)$. Overall, the entire pipeline can be represented as the composite function:

$$(t,p) = f(x), \qquad v = g(p), \qquad F(x) = h(v,t) = h(g(p),t).$$

The objective is to optimise the LLM function f (with g and h fixed) to maximise the expected popularity score of the generated videos. Formally, we wish to achieve:

$$\max_{f \in \mathscr{F}} \mathbb{E}_{x \sim \mathscr{X}} \left[h\big(g(p_f(x)), t_f(x)\big) \right],$$

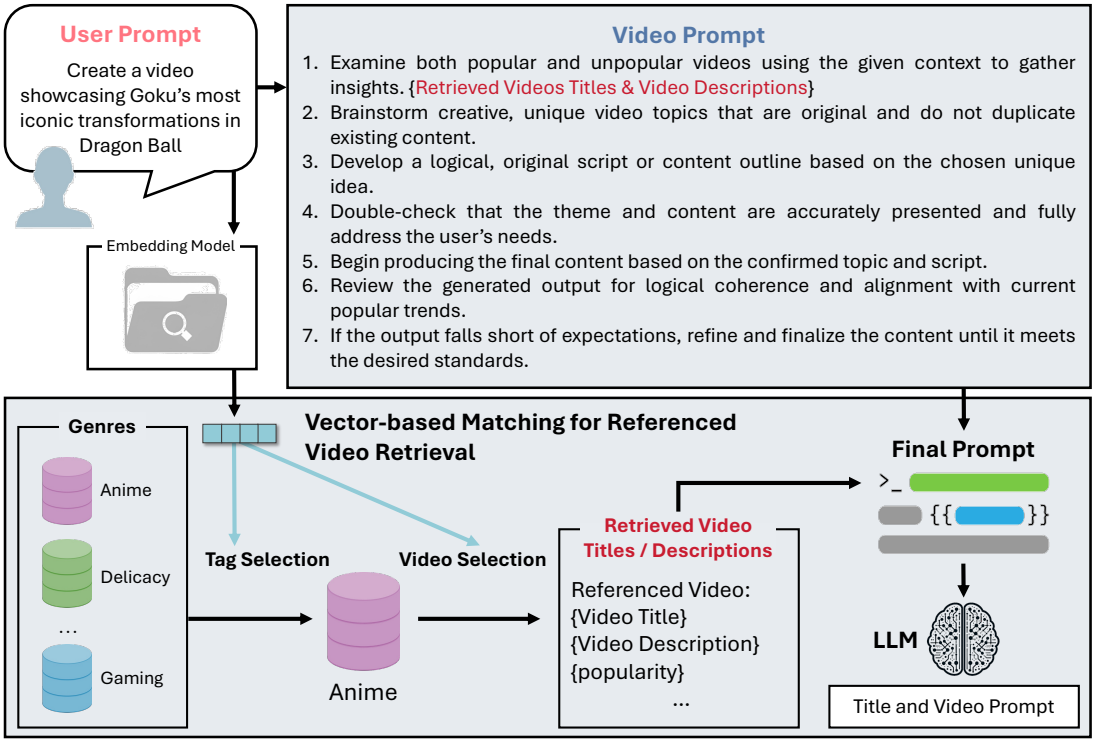


Figure 9.2: An overview of the Prompt Enhancement (PE) process. PE enables the LLM to review relevant micro-videos from the database and engage in chain-of-thought reasoning.

where $\mathscr{F}$ denotes the set of all possible configurations of the LLM. This formulation captures the aim of leveraging LLMs to enhance the creation of popular AI-generated micro-videos.

9.2.2 Pipeline

Based on the problem formulation, we propose a new **LLMPopcorn** pipeline in Figure 9.1

that begins by constructing user prompts, categorised into concrete and abstract prompts, which are constructed by rules into inputs for an LLM. The LLM serves as a video generation assistant, producing *video titles* and *video generation prompts* based on these inputs. These video generation prompts are then fed into a video generator to create corresponding videos. The generated videos are combined with their titles to form final short videos, which are subsequently evaluated using a pre-trained video popularity prediction model to assess their potential impact. This pipeline streamlines the process of AI-driven video generation and evaluation, enabling automatic popular micro-video content creation.

9.2.3 Prompt Enhancement

We demonstrate the Prompt Enhancement process in Figure 9.2, which enhances the input prompt for the LLM using a process inspired by human content creators. Formally, let $x \in \mathscr{X}$ be the user prompt (e.g., "Create a video showcasing Goku's most iconic transformations in

Dragon Ball."). An embedding function $f_{\text{embed}} : \mathscr{X} \to \mathbb{R}^d$ maps x to its vector representation:

$$\mathbf{x} = f_{\text{embed}}(x).$$

Given a predefined set of topic tags $\mathscr{T}$ (e.g., Food, Anime, Movie) with corresponding embeddings, a matching algorithm based on cosine similarity is used to select the most relevant tags:

$$\mathscr{T}_x = \arg\max_{t \in \mathscr{T}} sim\big(\mathbf{x}, f_{\text{embed}}(t)\big),$$

where $sim(\cdot,\cdot)$ denotes the cosine similarity. For each selected tag $t \in \mathscr{T}_x$, the system retrieves Top-K associated video samples from the test set of the MicroLens dataset[1] and partitions them into positive examples S_t^+ and negative examples S_t^- based on popularity. Finally, an enhancement function constructs the new prompt:

$$x' = \text{Enhance}\Big(x, \bigcup_{t \in \mathscr{T}_x} \{S_t^+, S_t^-\}\Big),$$

which is then used to guide the video generation.

9.3 Experiment Setup

Dataset. To enhance comprehensiveness, we designed two types of user prompts as inputs for the LLM: **Concrete** and **Abstract**. The user prompts are primarily derived from five key categories—Anime, Delicacy, Daily Sharing, Film & Television, and Gaming. We reference the MicroLens dataset (Y. Ni et al., 2023), as it is the largest available micro-video dataset. Concrete prompts include specific elements that provide clear details, while abstract prompts describe videos in a more general and high-level manner. Specifically, we utilised GPT-4o (Hurst et al., 2024) to generate two corresponding user prompt datasets, each containing *one hundred* prompts, based on instructions to include either concrete elements or general content.[2]

These datasets are further divided into the aforementioned five categories, with each category comprising twenty prompts. This approach ensures a balanced and diverse set of prompts across all categories, catering to both specific and broad input styles for the LLM.

Evaluation. In this study, we assess the popularity of generated micro-videos using the MMRA model (Zhong et al., 2024), defining popularity as the median number of comments for robustness, as per (Zhong et al., 2024). Higher values indicate greater median engagement, suggesting videos with higher engagement potential. In model-wise comparisons (e.g., Section 9.5 and Section 9.6), we use Win-Rate (Y. Gao, Xu, Wang, & Cohan, 2024) to evaluate model perfor-

[1] Since MMRA is trained on the training set of the MicroLens dataset, we exclusively use the test set to prevent data leakage.

[2] Due to the high cost of video generation, a total of 200 prompts is comparable in size to existing video-generation datasets (Chivileva et al., 2024).

Table 9.1: Comparison of popularity across different categories and LLMs.

Category	Abstract			Concrete		
	Llama	DeepSeek	GPT	Llama	DeepSeek	GPT
Anime	0.54	**0.92**	0.63	0.37	**0.45**	0.39
Daily Sharing	**0.59**	0.36	0.23	0.28	**0.69**	0.61
Delicacy	0.48	**1.12**	0.64	**0.66**	0.59	1.24
Film	0.68	**1.02**	0.82	0.46	**0.66**	0.35
Game	0.38	0.49	**0.55**	0.35	0.30	**0.56**

Table 9.2: Comparison of popularity medians for video generation tasks (using CogVideoX-5b as the video generator). Human-created videos are included for reference. The popularity predictor provides predicted values of Human Crafted video for comparison. Concrete-M and Abstract-M represent the human videos from the MicroLens dataset that are most closely matched to the same user prompt in the concrete and abstract datasets, respectively.

LLMPopcorn	**Concrete**	**Abstract**
Llama-3.3-70B	0.41	0.57
DeepSeek-V3	**0.56**	**0.73**
GPT-4o	0.52	0.49
Human Video	**Concrete-M**	**Abstract-M**
MicroLens	0.44	0.71

mance in predicting popularity, comparing micro-videos generated by Model A and Model B for a given user prompt. This pairwise approach is common in the literature (Y. Gao et al., 2024). Additionally, we conducted a user study following (Z. Cheng et al., 2024), involving five expert volunteers with master's degrees and over five years of experience in micro-video consumption and creation.

The thesis author, assisted by a co-author of the published study, manually checked all generated videos.

Implementation details In this thesis, we utilised five state-of-the-art LLMs, including two open-source models, Llama-3.3-70B and Qwen-2.5-72B, and three API-based models, DeepSeek-V3, DeepSeek-R1, and GPT-4o. For efficiency, we apply 4-bit quantisation to the open-source LLMs. Additionally, we incorporated three widely-used open-source video generation models: CogVideoX-5B, LTX-Video, and HunyuanVideo. All model weights were obtained from the HuggingFace platform[3] to ensure consistency and reproducibility. Notably, all open-source LLMs used in this study were instruction-tuned versions. We ensured reproducibility by fixing the random seed for open-source models, allowing for consistent results. We adopt Faiss[4] as a library for matching and indexing. For the prompt enhancement related to RQ9.2, we search the retrieval-augmented generation (RAG) size over $\{10, 50, 100\}$.

The video generation prompt contains existing video titles and video captions paired with

[3] https://huggingface.co/

[4] https://github.com/facebookresearch/faiss

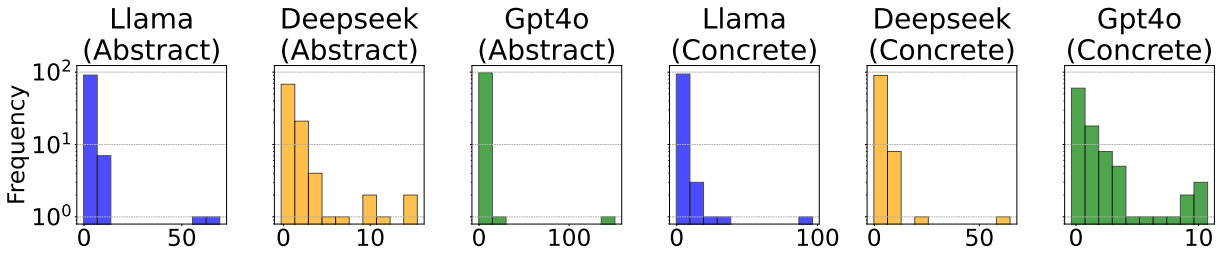


Figure 9.3: Predicted popularity distributions for different LLMs. Y-axis: frequency. X-axis: predicted popularity.

Table 9.3: Comparison of popularity for Prompt Enhancement's win rates across different LLMs.

LLM	Task	Win Rate (%)
Llama-3.3-70B	Concrete	**56**
	Abstract	**52**
DeepSeek-V3	Concrete	**55**
	Abstract	49
GPT-4o	Concrete	**51**
	Abstract	**53**

their popularity. All experiments are conducted on one H100 80G GPU.

Table 9.4: User study for Prompt Enhancement vs. Basic LLMPopcorn prompt.

LLM	Task	Win Rate (%)
User Evaluation	Concrete	**66.67**
	Abstract	**62.50**

9.4 Popular Micro-videos Generation (RQ9.1)

We evaluate three LLMs—Llama-3.3-70B, GPT-4o, and DeepSeek-V3—with CogVideoX as the default video generation model, using concrete and abstract prompt datasets from Section 9.3. We compare LLMPopcorn-generated video popularity to human-created videos from the MicroLens dataset (Y. Ni et al., 2023) by matching prompts to semantically similar video titles using an embedding model to reduce thematic bias. Table 9.2 shows that DeepSeek-V3-generated videos achieve median popularity scores of 0.56 (concrete) and 0.73 (abstract), rivaling MicroLens' 0.44 (concrete) and 0.71 (abstract). However, absolute scores remain low, likely due to limitations in open-source video models (Y. Liu, Zhang, Li, et al., 2024). Figure 9.3 reveals DeepSeek-V3's smoother popularity distribution for abstract prompts, with more videos in mid-to-high ranges, while Llama-3.3-70B and GPT-4o show concentrated low scores with rare outliers. For concrete prompts, model differences are minimal. Table 9.1 indicates

Table 9.5: Different LLMs' Win Rate comparisons across concrete and abstract datasets. BP denotes the basic prompt, WR represents the win rate, and PE refers to prompt enhancement.

LLM	Model Comparison	Concrete		Abstract	
		BP	PE	BP	PE
Llama-3.3-70B	vs. DeepSeek-V3	48	48	43	50
	vs. GPT-4o	49	51	49	53
	vs. DeepSeek-R1	46	40	45	53
	vs. Qwen-2.5-72B	51	56	50	55
DeepSeek-V3	vs. Llama-3.3-70B	52	52	57	50
	vs. GPT-4o	52	55	52	60
	vs. DeepSeek-R1	51	44	50	58
	vs. Qwen-2.5-72B	50	64	55	49
GPT-4o	vs. Llama-3.3-70B	51	49	51	46
	vs. DeepSeek-V3	48	45	48	40
	vs. DeepSeek-R1	47	43	46	48
	vs. Qwen-2.5-72B	52	54	49	44
DeepSeek-R1	vs. Llama-3.3-70B	54	60	55	47
	vs. DeepSeek-V3	49	56	50	42
	vs. GPT-4o	53	57	54	52
	vs. Qwen-2.5-72B	56	66	55	51
Qwen-2.5-72B	vs. Llama-3.3-70B	49	44	50	45
	vs. DeepSeek-V3	50	36	45	51
	vs. GPT-4o	48	46	51	56
	vs. DeepSeek-R1	44	34	45	49

DeepSeek-V3 leads in several abstract-prompt categories like Anime, Film, and Delicacy, with less dominance in Daily Sharing and Game. **(Answer to RQ9.1)** LLMs enhance micro-video creation by structuring descriptions, with DeepSeek-V3 achieving popularity comparable to or surpassing human-created videos, highlighting AI's potential for engaging short video generation.

9.5 Popularity Enhancement (RQ9.2)

Table 9.3 compares Prompt Enhancement (PE) to the basic LLMPopcorn prompt using win rates from pairwise comparisons across models. Llama-3.3-70B achieves 56% (concrete) and

Table 9.6: Different video generation models' Win Rate comparisons across concrete and abstract Datasets.

Video Generators	Comparison	Concrete		Abstract	
		BP	PE	BP	PE
CogVideoX-5b	vs. LTX-Video	49	38	42	39
	vs. HunyuanVideo	44	38	47	50
LTX-Video	vs. CogVideoX-5b	51	62	58	61
	vs. HunyuanVideo	48	52	49	60
HunyuanVideo	vs. CogVideoX-5b	56	62	53	50
	vs. LTX-Video	52	48	51	40

52% (abstract) win rates, outperforming LLMPopcorn. DeepSeek-V3 scores 55% (concrete) and 49% (abstract), while GPT-4o maintains 53% (abstract), showing PE's creative compatibility. A user study with DeepSeek-V3 evaluated PE against LLMPopcorn. Participants blindly compared five video pairs per prompt type, selecting the more popular video. Table 9.4 shows PE achieved over 60% win rates for both prompt types. **(Answer to RQ9.2)** PE significantly enhances prompt effectiveness, validated by offline metrics and user studies.

9.6 Benchmarking Models (RQ9.3)

We evaluated five large language models (LLMs): Llama-3.3-70B (Touvron et al., 2023), Qwen-2.5-72B (A. Yang et al., 2024), GPT-4o (Hurst et al., 2024), DeepSeek-V3, and DeepSeek-R1 (D. Guo et al., 2025), using their API versions. For video generation, we tested CogVideoX-5B (Z. Yang et al., 2024), LTX-Video (HaCohen et al., 2024), and HunyuanVideo (Kong et al., 2024), with instruction-tuned versions for open-source LLMs.

Table 9.5 shows DeepSeek-V3 and DeepSeek-R1 leading, with DeepSeek-R1 achieving a peak BP WR of 56% and PE WR of 66%. Llama-3.3-70B performs strongly in PE WR for abstract tasks (56%), while Qwen-2.5-72B lags with a PE WR of 34% against DeepSeek-R1. Table 9.6 indicates LTX-Video and HunyuanVideo outperform CogVideoX-5B, both exceeding 50% win rates in pairwise comparisons, with a peak PE WR of 62% on the Concrete dataset. **(Answer to RQ9.3)** DeepSeek-V3 and DeepSeek-R1 excel among LLMs, consistently leading in Concrete and Abstract scenarios. LTX-Video and HunyuanVideo show comparable, reliable performance in video generation across diverse prompts.

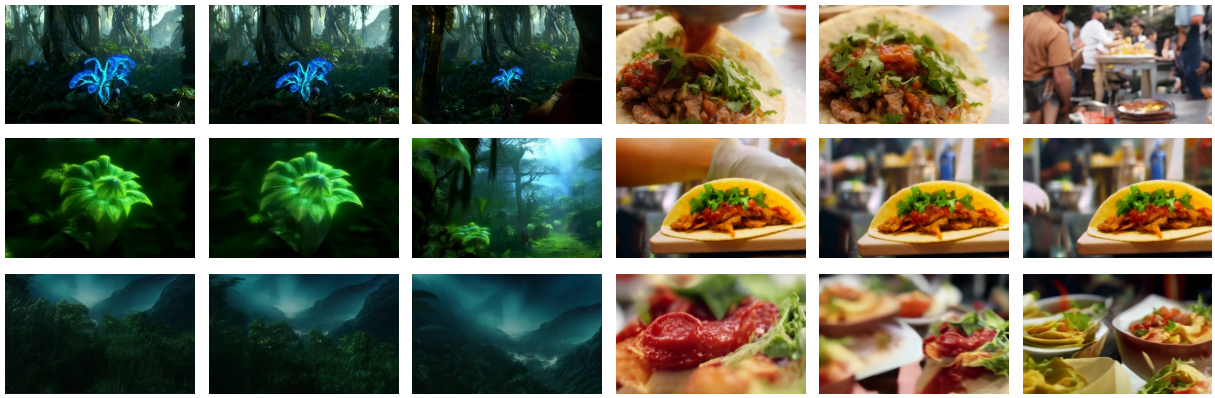


Figure 9.4: Video examples from LLMPopcorn, shown from left to right: abstract user queries (1–3) followed by concrete user queries (4–6).

9.7 Visualisation

To compare different video generation models, we present visualisations generated using two prompts across three models, as shown in Figures 9.4 and 9.5. Each figure includes three rows: videos in the first row are generated by **CogVideoX-5b**, those in the second row by **Hunyuan-Video**, and those in the third row by **LTX-Video**. The first prompt is abstract: *"Create a video about the history and evolution of a popular street food."* The second prompt is a concrete prompt: *"Create a video explaining the world-building in Avatar (2009)."* These examples help illustrate how each model handles abstract versus concrete prompts. In all figures, the first three columns represent abstract data, while the last three represent concrete data. From the two figures, Prompt Enhancement may boost popularity by enriching queries to produce dynamic, narrative-driven content. This is shown in its ability to transform abstract prompts into a more diverse range of imaginative scenes (whereas the baseline results are more uniform), while also depicting concrete subjects through engaging actions, like a chef's hands at work, instead of as static objects.

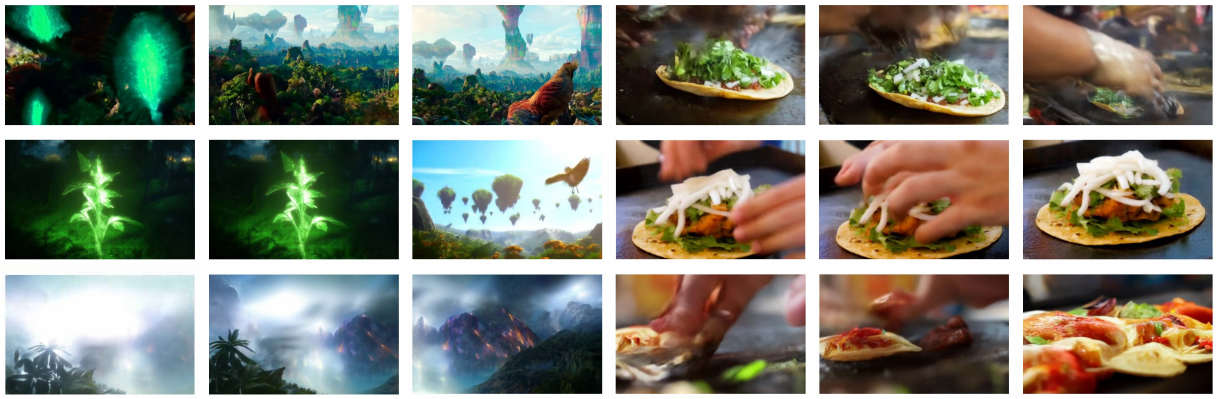


Figure 9.5: Video examples from LLMPopcorn with Prompt Enhancement, shown from left to right: abstract user queries (1–3) and concrete user queries (4–6).

9.8 Conclusion

This empirical study demonstrates that the proposed LLMPopcorn for micro-video generation assisted by state-of-the-art LLMs achieves slightly higher predicted popularity than the matched evaluated human-created videos as assessed by the offline popularity predictor. These results highlight the potential of this approach. To further enhance the LLMPopcorn pipeline, we introduced prompt enhancement (PE), which exhibited strong adaptability across multiple LLMs and video-generation models. Among the tested models, DeepSeek-V3 and DeepSeek-R1 consistently performed best as LLMs, while LTX-Video and HunyuanVideo produced reliable outputs across diverse prompts. In future work, we plan to explore reinforcement learning and fine-tuning techniques to better align prompts with contextual cues, ultimately improving video quality and engagement. This study pioneers popularity-driven micro-video generation, establishing a strong foundation for future research and innovation.

Chapter 10

Conclusion and Future Work

10.1 Conclusion

This thesis investigated how the broad representation and generation capabilities of multimodal foundation models (MFMs) can be adapted for recommender systems. The central finding is that adaptation should be defined by the role that the MFM plays in the recommendation pipeline: representation providers should be aligned with behavioural prediction and ranking objectives, whereas generators need to be evaluated and adapted according to the platform function served by their outputs. To make the relationship between the thesis argument and its evidence explicit, this section answers the seven research questions introduced in Chapter 1 in the same order.

10.1.1 RQ1: Efficient Continuous Representation Adaptation

Question. How can multimodal foundation models be efficiently adapted through end-to-end joint optimisation with a recommender in the continuous representation setting?

Answer. Chapter 3 shows that MFM-based representation learning in the continuous setting can be adapted efficiently by decoupling lightweight trainable side networks from frozen multimodal backbones and caching the backbone hidden states. IISAN preserves recommendation supervision for the adaptation modules while removing the backbone from the backward graph. The results demonstrate that practical efficiency should be assessed through training time, GPU memory, cacheability, and trainable parameters together, rather than through parameter count alone. Effective adaptation also requires both intra-modal refinement and inter-modal interaction.

10.1.2 RQ2: Scaling to Larger and Asymmetric Encoders

Question. How can continuous representation adaptation be scaled from paired encoders to larger and asymmetric foundation encoders?

Answer. Chapter 4 demonstrates that cached side adaptation can scale to encoders with different depths, dimensions, and capacities through dimension transformation and structured layer selection. IISAN-Versa supports settings in which a large LLM-based text encoder is paired with a smaller vision encoder, and it reveals a clear text-encoder scaling effect when the architectural mismatch is handled explicitly. Scalable MFM adaptation therefore requires architecture-aware alignment in addition to computational efficiency.

10.1.3 RQ3: Adapting Multiple Modality-Specific Foundation Models

Question. How can multiple modality-specific foundation models be jointly adapted with a recommender in the continuous representation setting?

Answer. Chapter 5 shows that multiple modality-specific MFMs can be adapted through cross-modal side networks that exchange information at corresponding depths, followed by item-specific mixture-of-modality expert fusion. CROSSAN improves as text, image, video, and audio encoders are added, whereas a simple extension of independent side adapters does not scale as consistently. Additional modalities become useful when the adaptation mechanism explicitly controls cross-modal interaction, routing, and selective fusion.

10.1.4 RQ4: Deep Adaptation of Unified Single-Tower Models

Question. How can deep unified single-tower multimodal embedding models be adapted for recommendation?

Answer. Chapter 6 shows that deep side adaptation requires explicit preservation and refinement of representation streams. Stresa combines Stream-aware Hidden-Adapter Fusion (SHAF) with the Residual Stream Adapter (ReSA) so that earlier information is not overwritten as more hidden layers are adapted. Unlike a standard side adapter, Stresa remains stable when adaptation extends to the full backbone depth. Deep unified models should therefore be adapted as progressive memory-refinement systems rather than as sequences of independent shallow residual updates.

10.1.5 RQ5: Joint Optimisation of Discrete Representations

Question. How can semantic-ID construction and a generative recommender be effectively aligned through joint optimisation?

Answer. Chapter 7 demonstrates that semantic IDs can be optimised jointly with a generative recommender when discrete assignment is made differentiable and codebook exploration is controlled. DIGER uses Gumbel-based exploratory learning, soft codebook updates, and uncertainty decay to prevent the premature collapse observed with a naive straight-through estimator. It improves next-item generation while maintaining balanced code usage and sta-

ble train–inference assignments. Discrete adaptation therefore requires both a gradient pathway from the recommendation objective and an explicit exploration-to-exploitation mechanism. By addressing non-differentiable code assignments and codebook collapse, these results lay a methodological foundation for a future unified large recommendation model with end-to-end joint optimisation of MFM-based content encoding, semantic-ID construction and generative recommendation.

10.1.6 RQ6: Platform-Side Missing-Modality Completion

Question. How should multimodal foundation models be evaluated and adapted for platform-side missing-modality completion, supporting both item presentation and recommendation-model training?

Answer. Chapter 8 shows that platform-side completion should be evaluated at two levels: the fidelity and usability of the generated content, and its downstream recommendation utility. MMPCBench combines CQBench and RecBench to measure both roles. Current MLLMs capture high-level product semantics but remain weaker at precise word-, pixel-, and attribute-level reconstruction; nevertheless, their completed modalities often retain recommendation performance close to ground-truth content. GRPO improves Image-to-Text completion but not Text-to-Image completion, showing that generation adaptation is task-dependent and requires direction-specific reward design.

10.1.7 RQ7: User-Side Popularity-Oriented Generation

Question. How can foundation-model-assisted content generation be studied and improved on the user side when the generated content is intended to gain popularity?

Answer. Chapter 9 shows that user-side generation can be studied as a content-creation pipeline in which LLM-generated titles and video prompts are evaluated together with the resulting micro-videos. LLMPopcorn demonstrates that model choice and prompt enhancement materially affect predicted popularity, and that retrieval- and reasoning-based prompt enhancement can improve both offline win rates and human preference. Because popularity prediction is learned from historical exposure and interaction data, these results should be interpreted as evidence of engagement potential rather than as a complete measure of long-term user or platform value.

10.1.8 Overall Answer to the Thesis Question

Taken together, the seven answers establish a unified view of recommendation-oriented MFM adaptation. For representation, the adaptation mechanism should match the representation interface and architecture: cached side networks support efficient continuous representation adapta-

tion, architecture-aware alignment handles asymmetric encoders, cross-modal routing supports multiple MFMs, stream-aware memory supports deep unified models, and exploratory differentiable quantisation supports discrete semantic IDs. For generation, the objective and evidence should match the role of the output: platform-side completion requires faithfulness and downstream utility, whereas user-side creation requires engagement-oriented evaluation together with safeguards against exposure bias and narrow optimisation.

The broader implication is that recommender systems provide an ecosystem-level test for foundation models. Representations affect ranking, ranking affects exposure, exposure affects user feedback, and feedback shapes future training data; generated product content and generated videos can alter the catalogue and candidate supply inside the same loop. Recommendation-oriented adaptation therefore requires explicit alignment among capability, interface, architecture, objective, efficiency, and evaluation. Across the seven technical studies, this thesis provides methods, benchmarks, and empirical evidence for designing that alignment.

10.2 Limitations

The first limitation concerns evaluation. Most experiments in this thesis use offline ranking metrics, benchmarked content-quality metrics, human judgments, or predicted popularity scores. These protocols provide reproducible and controlled evidence, and they are necessary for comparing methods across datasets and models. Real recommendation platforms also involve exposure policies, long-term user satisfaction, creator responses, feedback loops, and changing catalogues. This gap matters especially for the generation chapters. MMPCBench evaluates generated product content through benchmarked content quality and downstream recommendation utility, while LLMPopcorn uses predicted popularity and user preference as offline evidence for engagement-oriented generation. Deployment studies with live users and real exposure dynamics would provide stronger evidence for long-term effects.

The second limitation concerns domain and dataset coverage. The thesis studies product recommendation and micro-video recommendation, which are important content-rich scenarios. Other domains, including news, music, education, healthcare, tourism, and local services, may involve different modality structures, user intents, safety requirements, and feedback patterns. A method designed for product images and descriptions may need different assumptions for long-form video, audio-first items, document-heavy content, or safety-critical recommendation. The thesis establishes general adaptation principles and concrete methods, while broader domain validation remains an important next step.

The third limitation concerns backbone and generator dependence. The empirical conclusions depend on the selected foundation encoders, multimodal embedding models, video generators, and multimodal large language models. These models differ in pretraining data, architecture, instruction tuning, modality coverage, and generation behaviour. MMPCBench and

LLMPopcorn therefore evaluate recommendation-oriented formulations together with the capabilities of their underlying generators. As stronger MFMs appear, the relative contribution of the adaptation method and the backbone may change. Future studies should separate backbone effects, prompt effects, decoding effects, and recommendation-oriented adaptation effects more systematically.

The fourth limitation concerns the current separation between representation and generation pipelines. The thesis treats representation adaptation and generation adaptation as connected perspectives, and most chapters still study them in separate experimental settings. Representation chapters produce adapted embeddings or semantic IDs for recommendation. Generation chapters produce completed product content or generated micro-videos and then evaluate their recommendation relevance. Real systems can connect these processes more tightly. Generated descriptions may become inputs to representation adaptation; adapted representations may guide content generation; user feedback may update both encoders and generators. Fully joint representation-generation adaptation remains a promising extension of the thesis.

The fifth limitation concerns deployment lifecycle. Cached side adaptation improves practical efficiency during training, while production recommendation systems also require decisions about refresh frequency, new-item handling, representation drift, adapter versioning, and serving latency. Discrete semantic IDs raise related deployment questions: item codes may need to evolve as catalogues and user behaviour change, while abrupt ID changes can affect generative retrieval stability. The thesis addresses the modelling and efficiency foundations for these issues, and a complete production lifecycle for continual MFM adaptation remains future work.

10.3 Future Work

10.3.1 Unified Representation and Generation Adaptation

Within representation adaptation, the longer-term goal is a unified large recommendation model that jointly optimises MFM-based content encoding, semantic-ID construction and generative recommendation under one recommendation objective. DIGER contributes methodological foundations for this goal by addressing non-differentiable code assignments and codebook collapse. Future work should extend stable recommendation-driven gradients to the upstream trainable MFM components and evaluate the resulting model as a complete end-to-end system. One possible efficient implementation would combine DIGER with the side-tuning methods developed in Chapters 3–6, jointly training the side adapters, semantic-ID construction and generative recommender while retaining the frozen-backbone design. This combination remains an avenue for exploration within the broader unified-model objective.

A natural next direction is to build unified pipelines where representation adaptation and generation adaptation directly inform each other. Generated product descriptions, images, captions,

or videos can become additional modalities for representation learning. Adapted recommendation representations can guide generation by selecting content attributes, prompts, titles, or visual concepts that match user interests and catalogue structure. A platform-side system could generate a missing product modality, evaluate the generated content through both fidelity and downstream ranking utility, and use this feedback to update the generator or adaptation module. A user-side system could use recommendation-oriented embeddings to guide micro-video prompt generation while preserving diversity and user control.

This direction requires careful objective design. Joint optimisation can create feedback loops when generated content is rewarded only by short-term ranking gains or predicted popularity. Future frameworks should combine recommendation utility with content faithfulness, uncertainty estimation, diversity control, and user-centred evaluation. Such systems would connect the two thesis perspectives more directly and turn MFM adaptation into a closed-loop process between content generation, representation learning, and recommendation feedback.

10.3.2 Online and Long-Term Evaluation

Future work should extend offline evidence with online and long-term evaluation. Offline metrics support reproducibility, while online studies can measure user satisfaction, trust, novelty, diversity, retention, and behavioural change. For representation adaptation, online evaluation can show how adapted MFM embeddings influence exposure patterns and user experience. For product content completion, online evaluation can test whether generated descriptions or images improve user understanding and purchase decisions. For popularity-oriented generation, online evaluation can examine whether generated micro-videos attract meaningful engagement and whether they affect creator incentives or content diversity.

Long-term evaluation is particularly important because MFM-generated content can change the item supply and the future training data of a recommender. A generated modality may improve short-term ranking while shifting catalogue representation. A popularity-oriented generator may improve immediate engagement while increasing stylistic repetition. Future studies should therefore track delayed effects, repeated exposure, diversity, fairness, creator behaviour, and user satisfaction over time.

10.3.3 Continual and Cache-Aware Adaptation

The representation chapters show that cacheability is central to efficient MFM adaptation in recommendation. Future work should develop continual and cache-aware adaptation methods for dynamic catalogues. New items arrive, existing item content changes, user preferences drift, and platform objectives evolve. A practical system should decide which cached hidden states require refresh, which adapter layers should be updated, and how to balance freshness with computation cost. Layer-level caching, selective refresh, adapter scheduling, and drift-aware

update policies are promising directions.

Continual adaptation also matters for discrete semantic IDs. When semantic IDs are learned under recommendation supervision, the item-token space may need to expand or reorganise as new items enter the catalogue. Future methods should study stable ID updates, backward-compatible tokenisation, incremental codebook learning, and retrieval robustness under changing item codes. This would extend DIGER from joint learning of discrete representations to continual semantic-ID management.

10.3.4 Broader Multimodal Architectures and Modalities

Future recommender systems will likely combine more diverse foundation-model components, including multimodal large language models, unified embedding models, video-language models, audio-language models, graph-enhanced encoders, and generator-retriever hybrids. They will also process richer item signals, such as structured attributes, reviews, prices, metadata, knowledge graphs, long videos, and user-generated comments. Future work should evaluate the adaptation principles from this thesis across these broader model families and modality combinations.

This direction calls for flexible adaptation mechanisms. A recommender may need to route different item types through different encoders, activate adapters according to available modalities, combine frozen and trainable components under resource budgets, and support both retrieval and generation. The methods developed in this thesis provide building blocks for such systems: cached decoupled adaptation, asymmetric layer alignment, cross-modal gated side adaptation, stream-aware residual adaptation, and differentiable semantic IDs. Combining these components into modular adaptation systems is a promising path toward general MFM-based recommendation.

10.3.5 Trustworthy Recommendation-Oriented Generation

Generated content in recommender systems raises issues of faithfulness, provenance, controllability, and platform responsibility. Product descriptions and images influence user decisions, so completion systems should preserve factual product attributes and avoid unsupported details. Micro-video generators can influence what users watch and what creators imitate, so popularity-oriented generation should consider diversity, safety, authenticity, and user well-being. Future work should develop mechanisms for generated-content disclosure, source tracing, hallucination detection, uncertainty estimation, and controllable generation under recommendation objectives.

Trustworthy generation also needs role-specific evaluation. Platform-side product completion should be assessed through attribute fidelity, content quality, human judgment, and recommendation utility. User-side micro-video generation should be assessed through engagement potential, diversity, safety, creator control, and long-term platform effects. Future benchmarks

should connect these dimensions and make their assumptions explicit, so that recommendation-oriented generation improves platform utility while preserving user trust.

10.3.6 Towards Recommendation-Oriented Foundation Models

The long-term direction suggested by this thesis is the development of foundation models designed for recommendation-oriented use. Such models would encode multimodal content, behavioural sequences, catalogue structure, user intent, and platform constraints within compatible representation and generation interfaces. They would support efficient adaptation, controllable generation, interpretable evaluation, continual updates, and deployment-aware caching. They would also distinguish semantic similarity from recommendation utility, so that item representations and generated content directly reflect user choice and recommendation outcomes.

This direction points to a pragmatic future for MFM-based recommendation. General-purpose MFMs supply powerful representation and generation capabilities, and recommender systems define the objectives, constraints, and feedback loops that make those capabilities useful. The core challenge is to design interfaces between these two sides. Across continuous representation adaptation and discrete representation adaptation, platform-side and user-side generation, and seven technical studies, this thesis shows that recommendation-oriented adaptation provides a practical route for bringing MFMs into recommender systems. It offers methods and empirical insights for adapting MFM representation and generation capabilities today, and it lays the groundwork for future recommendation-oriented foundation models.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., ... others (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Adams, R. P., & Zemel, R. S. (2011). Ranking via sinkhorn propagation. *arXiv preprint arXiv:1106.1925*.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., ... others (2022). Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, *35*, 23716–23736.

Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, *33*, 12449–12460.

Banerjee, S., & Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization* (pp. 65–72).

Bao, K., Zhang, J., Wang, W., Zhang, Y., Yang, Z., Luo, Y., ... Tian, Q. (2025). A bi-step grounding paradigm for large language models in recommendation systems. *ACM Transactions on Recommender Systems*, *3*(4), 1–27.

Bao, K., Zhang, J., Zhang, Y., Wang, W., Feng, F., & He, X. (2023). Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th acm conference on recommender systems* (pp. 1007–1014).

Bengio, Y., Léonard, N., & Courville, A. (2013). Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Brown, T., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.

Cai, H., Gan, C., Zhu, L., & Han, S. (2020). Tinytl: Reduce memory, not parameters for efficient on-device learning. *Advances in Neural Information Processing Systems*, *33*, 11285–11297.

Cai, Y., Mao, S., Wu, W., Wang, Z., Liang, Y., Ge, T., ... others (2023). Low-code llm: Visual programming over llms. *arXiv preprint arXiv:2304.08103*, *2*.

Chen, L., Msigwa, G., Yang, M., Osman, A. I., Fawzy, S., Rooney, D. W., & Yap, P.-S. (2022). Strategies to achieve a carbon neutral society: a review. *Environmental Chemistry Letters*,

20(4), 2277–2310.

Chen, Y., Liu, Y., Zhao, J., & Zhu, Q. (2020). Mobile edge cache strategy based on neural collaborative filtering. *IEEE Access*, *8*, 18475–18482.

Chen, Z., Duan, Y., Wang, W., He, J., Lu, T., Dai, J., & Qiao, Y. (2022). Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*.

Chen, Z., Zhang, F., Zhang, Y., Zhu, J., Li, Q., & Liu, Y. (2025). Adaptive latent disease state learning for multimodal alzheimer's disease biomarker detection with missing modalities. *Pattern Recognition*, 112389.

Chen, Z., Zhang, Y., Fang, Y., Geng, Y., Guo, L., Chen, X., ... others (2024). Knowledge graphs meet multi-modal learning: A comprehensive survey. *arXiv preprint arXiv:2402.05391*.

Cheng, Y., Pan, Y., Zhang, J., Ni, Y., Sun, A., & Yuan, F. (2023). An image dataset for benchmarking recommender systems with raw pixels. *arXiv preprint arXiv:2309.06789*.

Cheng, Z., Cheng, Z.-Q., He, J.-Y., Sun, J., Wang, K., Lin, Y., ... Hauptmann, A. G. (2024). Emotion-llama: Multimodal emotion recognition and reasoning with instruction tuning. *Advances in Neural Information Processing Systems*, *37*, 110805–110853.

Chivileva, I., Lynch, P., Ward, T. E., & Smeaton, A. F. (2024). A dataset of text prompts, videos and video quality metrics from generative text-to-video ai models. *Data in Brief*, *54*, 110514.

Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for youtube recommendations. In *Proceedings of the 10th acm conference on recommender systems* (pp. 191–198).

Cui, X., Lu, W., Tong, Y., Li, Y., & Zhao, Z. (2025). Multi-modal multi-behavior sequential recommendation with conditional diffusion-based feature denoising. In *Proceedings of the 48th international acm sigir conference on research and development in information retrieval* (pp. 1593–1602).

Cui, Z., Ma, J., Zhou, C., Zhou, J., & Yang, H. (2022). M6-rec: Generative pretrained language models are open-ended recommender systems. *arXiv preprint arXiv:2205.08084*.

Dai, D., Deng, C., Zhao, C., Xu, R., Gao, H., Chen, D., ... others (2024). Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*.

Dai, J., Fang, Q., Hu, J., Cai, D., Yang, Y., & Zhao, C. (2026). Cross-modal attention network with dual graph learning in multimodal recommendation. *ACM Transactions on Multimedia Computing, Communications and Applications*.

Dai, W., Hou, L., Shang, L., Jiang, X., Liu, Q., & Fung, P. (2022). Enabling multimodal generation on clip via vision-language knowledge distillation. *arXiv preprint arXiv:2203.06386*.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., ... others (2023). Scaling vision transformers to 22 billion parameters. In *International conference on machine learning* (pp. 7480–7512).

Deng, J., Wang, S., Cai, K., Ren, L., Hu, Q., Ding, W., ... Zhou, G. (2025). OneRec:

Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*. Retrieved from `https://arxiv.org/abs/2502.18965` doi: 10.48550/arXiv.2502.18965

Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171–4186).

Dong, Z., Liu, X., Chen, B., Polak, P., & Zhang, P. (2024). Musechat: A conversational music recommendation system for videos. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 12775–12785).

Dosovitskiy, A., Beyer, L., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Elsayed, S., Brinkmeyer, L., & Schmidt-Thieme, L. (2022). End-to-end image-based fashion recommendation. *arXiv preprint arXiv:2205.02923*.

Fan, A., Grave, E., & Joulin, A. (2019). Reducing transformer depth on demand with structured dropout. *arXiv preprint arXiv:1909.11556*.

Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., & Feichtenhofer, C. (2021). Multiscale vision transformers. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 6824–6835).

Fei, N., Lu, Z., Gao, Y., Yang, G., Huo, Y., Wen, J., ... others (2022). Towards artificial general intelligence via a multimodal foundation model. *Nature Communications*, *13*(1), 3094.

Feichtenhofer, C., Fan, H., Malik, J., & He, K. (2019). Slowfast networks for video recognition. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 6202–6211).

Fu, J., Deng, W., Zheng, K., Arapakis, I., Ye, Y., Ni, Y., ... Ge, X. (2026). Benchmarking multimodal large language models for missing modality completion in product catalogues. *Pattern Recognition*, 114020.

Fu, J., Ge, X., Karatzoglou, A., Arapakis, I., Verberne, S., Jose, J. M., & Ren, Z. (2026). Differentiable semantic id for generative recommendation. *arXiv preprint arXiv:2601.19711*.

Fu, J., Ge, X., Xin, X., Karatzoglou, A., Arapakis, I., Wang, J., & Jose, J. M. (2024). Iisan: Efficiently adapting multimodal representation for sequential recommendation with decoupled peft. In *Proceedings of the 47th international acm sigir conference on research and development in information retrieval* (pp. 687–697).

Fu, J., Ge, X., Xin, X., Karatzoglou, A., Arapakis, I., Zheng, K., ... Joemon, J. M. J. (2025).

Efficient and effective adaptation of multimodal foundation models in sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Fu, J., Ge, X., Xin, X., Yu, H., Feng, Y., Karatzoglou, A., ... Jose, J. (2025). The 1st erel@ mir workshop on efficient representation learning for multimodal information retrieval. In *Companion proceedings of the acm on web conference 2025* (pp. 2149–2152).

Fu, J., Ge, X., Zheng, K., Karatzoglou, A., Arapakis, I., Xin, X., ... Jose, J. M. (2026). Llm-popcorn: Exploring llms as assistants for popular micro-video generation. In *Icassp 2026-2026 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 12007–12011).

Fu, J., Ni, Y., Jose, J. M., Arapakis, I., Zheng, K., Li, Y., & Ge, X. (2025). Crossan: Towards efficient and effective adaptation of multiple multimodal foundation models for sequential recommendation. *arXiv preprint arXiv:2504.10307*.

Fu, J., Yuan, F., Song, Y., Yuan, Z., Cheng, M., Cheng, S., ... Pan, Y. (2024). Exploring adapter-based transfer learning for recommender systems: Empirical studies and practical insights. In *Proceedings of the 17th acm international conference on web search and data mining* (pp. 208–217).

Fu, J., Zheng, K., Arapakis, I., Deng, W., Xin, X., Jose, J. M., & Ge, X. (2026). Stream-aware side adaptation for large pre-trained multimodal embedding models in sequential recommendation. *arXiv preprint arXiv:2607.10909*.

Gao, L., Zhang, Y., Han, J., & Callan, J. (2021). Scaling deep contrastive learning batch size under memory limited setup. *arXiv preprint arXiv:2101.06983*.

Gao, P., Han, J., Zhang, R., Lin, Z., Geng, S., Zhou, A., ... others (2023). Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*.

Gao, Y., Xu, G., Wang, D. Z., & Cohan, A. (2024). Bayesian calibration of win rate estimation with llm evaluators. In *Proceedings of the 2024 conference on empirical methods in natural language processing* (pp. 4757–4769).

Gardner Jr, E. S. (2006). Exponential smoothing: The state of the art—part ii. *International journal of forecasting*, *22*(4), 637–666.

Ge, X., Chen, F., Jose, J. M., Ji, Z., Wu, Z., & Liu, X. (2021). Structured multi-modal feature embedding and alignment for image-sentence retrieval. In *Proceedings of the 29th acm international conference on multimedia* (pp. 5185–5193).

Ge, X., Chen, F., Shen, C., & Ji, R. (2019). Colloquial image captioning. In *2019 ieee international conference on multimedia and expo (icme)* (pp. 356–361).

Ge, X., Chen, F., Xu, S., Tao, F., & Jose, J. M. (2023). Cross-modal semantic enhanced interaction for image-sentence retrieval. In *Proceedings of the ieee/cvf winter conference on applications of computer vision* (pp. 1022–1031).

Ge, X., Fu, J., Chen, F., An, S., Sebe, N., & Jose, J. M. (2024). Towards end-to-end explainable facial action unit recognition via vision-language joint learning. In *Proceedings of the*

32nd acm international conference on multimedia (pp. 8189–8198).

Ge, X., Xu, S., Chen, F., Wang, J., Wang, G., An, S., & Jose, J. M. (2024). 3shnet: Boosting image–sentence retrieval via visual semantic–spatial self-highlighting. *Information Processing & Management*, *61*(4), 103716.

Geng, S., Liu, S., Fu, Z., Ge, Y., & Zhang, Y. (2022). Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). *arXiv preprint arXiv:2203.13366*.

Geng, S., Tan, J., Liu, S., Fu, Z., & Zhang, Y. (2023). Vip5: Towards multimodal foundation models for recommendation. In *Findings of the association for computational linguistics: Emnlp 2023* (pp. 9606–9620).

Ghalandari, D., Hokamp, C., & Ifrim, G. (2022). Efficient unsupervised sentence compression by fine-tuning transformers with reinforcement learning. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1267–1280).

Gomez, A. N., Ren, M., Urtasun, R., & Grosse, R. B. (2017). The reversible residual network: Backpropagation without storing activations. *Advances in neural information processing systems*, *30*.

Gómez-Chova, L., Tuia, D., Moser, G., & Camps-Valls, G. (2015). Multimodal classification of remote sensing images: A review and future directions. *Proceedings of the IEEE*, *103*(9), 1560–1584.

Gong, Y., Chung, Y.-A., & Glass, J. (2021). Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*.

Graves, A. (2011). Practical variational inference for neural networks. *Advances in neural information processing systems*, *24*.

Gu, C., Bu, J., Zhou, X., Yao, C., Ma, D., Yu, Z., & Yan, X. (2022). Cross-modal image retrieval with deep mutual information maximization. *Neurocomputing*, *496*, 166–177.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., ... others (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Guo, J., Gong, R., Ma, Y., Liu, M., Xi, X., Nie, X., & Yin, Y. (2024). A survey of micro-video analysis. *Multimedia Tools and Applications*, *83*(11), 32191–32212.

Guo, W., Huang, H., Kong, X., & He, R. (2019). Learning disentangled representation for cross-modal retrieval with deep mutual information estimation. In *Proceedings of the 27th acm international conference on multimedia* (pp. 1712–1720).

Guo, X., Zhang, T., Wang, F., Wang, X., Zhang, X., Liu, X., & Cui, Z. (2025). Mmhcl: Multi-modal hypergraph contrastive learning for recommendation. *ACM Transactions on Multimedia Computing, Communications and Applications*, *21*(10), 1–23.

Guo, X., Zhang, T., Xue, Y., Wang, C., Wang, F., & Cui, Z. (2025). M 3 rec: Selective state

space models with mixture-of-modality experts for multi-modal sequential recommendation. In *Icassp 2025-2025 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 1–5).

Gupta, A. K., Smith, K. G., & Shalley, C. E. (2006). The interplay between exploration and exploitation. *Academy of management journal*, *49*(4), 693–706.

HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., ... Bibi, O. (2024). Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*.

Han, R., Yin, B., Chen, S., Jiang, H., Jiang, F., Li, X., ... others (2025). Mtgr: Industrial-scale generative recommendation framework in meituan. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 5731–5738).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).

He, P., Gao, J., & Chen, W. (2021). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

He, R., & McAuley, J. (2016). Vbpr: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 30).

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd international acm sigir conference on research and development in information retrieval* (pp. 639–648).

He, X., Li, C., Zhang, P., Yang, J., & Wang, X. E. (2022). *Parameter-efficient fine-tuning for vision transformers.* arXiv. doi: 10.48550/ARXIV.2203.16329

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T.-S. (2017). Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web* (pp. 173–182).

He, Y., Fu, J., Zheng, K., Xu, S., Chen, F., Li, J., ... Ge, X. (2025). Double-filter: Efficient fine-tuning of pre-trained vision-language models via patch&layer filtering. In *Forty-second international conference on machine learning*.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, *30*.

Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Hore, A., & Ziou, D. (2010). Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition* (pp. 2366–2369).

Hou, M., Wu, L., Liao, Y., Yang, Y., Zhang, Z., Zheng, C., … Hong, R. (2025). A survey on generative recommendation: Data, model, and tasks. *arXiv preprint arXiv:2510.27157*.

Hou, Y., He, Z., McAuley, J., & Zhao, W. X. (2022). Learning vector-quantized item representation for transferable sequential recommenders. *arXiv preprint arXiv:2210.12316*.

Hou, Y., Li, J., Shin, A., Jeon, J., Santhanam, A., Shao, W., … McAuley, J. (2025). Generating long semantic ids in parallel for recommendation. In *Proceedings of the 31st acm sigkdd conference on knowledge discovery and data mining v. 2* (pp. 956–966).

Hou, Y., Mu, S., Zhao, W. X., Li, Y., Ding, B., & Wen, J.-R. (2022). Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th acm sigkdd conference on knowledge discovery and data mining* (pp. 585–593).

Hou, Y., Zhang, A., Sheng, L., Yang, Z., Wang, X., Chua, T.-S., & McAuley, J. (2025). Generative recommendation models: Progress and directions. In *Companion proceedings of the acm on web conference 2025* (pp. 13–16).

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., … Gelly, S. (2019). Parameter-efficient transfer learning for nlp. In *International conference on machine learning* (pp. 2790–2799).

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., … Chen, W. (2021). Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., … others (2022). Lora: Low-rank adaptation of large language models. *Iclr*, *1*(2), 3.

Hu, H., Guo, W., Liu, Y., & Kan, M.-Y. (2023). Adaptive multi-modalities fusion in sequential recommendation systems. In *Proceedings of the 32nd acm international conference on information and knowledge management* (pp. 843–853).

Hu, H., Liu, Q., Li, C., & Kan, M.-Y. (2024). Lightweight modality adaptation to sequential recommendation via correlation supervision. *arXiv preprint arXiv:2401.07257*.

Hu, J., Gao, J., Zhao, X., Hu, Y., Liang, Y., Wang, Y., … Yin, H. (2024). Bivrec: Bidirectional view-based multimodal sequential recommendation. *arXiv preprint arXiv:2402.17334*.

Hu, J., Liu, Y., Zhao, J., & Jin, Q. (2021). Mmgcn: Multimodal fusion via deep graph convolution network for emotion recognition in conversation. *arXiv preprint arXiv:2107.06779*.

Hua, W., Xu, S., Ge, Y., & Zhang, Y. (2023). How to index item ids for recommendation foundation models. In *Proceedings of the annual international acm sigir conference on research and development in information retrieval in the asia pacific region* (pp. 195–204).

Huan, Z., Ding, K., Li, A., Zhang, X., Min, X., He, Y., … others (2024). Exploring multi-scenario multi-modal ctr prediction with a large scale dataset. In *Proceedings of the 47th international acm sigir conference on research and development in information retrieval* (pp. 1232–1241).

Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., & Liu, W. (2019). Ccnet: Criss-cross at-

tention for semantic segmentation. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 603–612).

Huh, M., Cheung, B., Agrawal, P., & Isola, P. (2023). Straightening out the straight-through estimator: Overcoming optimization challenges in vector quantized networks. In *International conference on machine learning* (pp. 14096–14113).

Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., ... others (2024). Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jain, S. M. (2022). Hugging face. In *Introduction to transformers for nlp: With the hugging face library and models to solve problems* (pp. 51–67). Springer.

Jang, E., Gu, S., & Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Ji, W., Liu, X., Zhang, A., Wei, Y., Ni, Y., & Wang, X. (2023). Online distillation-enhanced multi-modal transformer for sequential recommendation. In *Proceedings of the 31st acm international conference on multimedia* (pp. 955–965).

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., ... Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning* (pp. 4904–4916).

Jin, H., Zhang, Y., Meng, D., Wang, J., & Tan, J. (2024). A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*.

Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. In *2018 ieee international conference on data mining (icdm)* (pp. 197–206).

Karedla, R., Love, J. S., & Wherry, B. G. (1994). Caching strategies to improve disk system performance. *Computer*, *27*(3), 38–46.

Karimi Mahabadi, R., Henderson, J., & Ruder, S. (2021). Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in Neural Information Processing Systems*, *34*, 1022–1035.

Ke, G., He, S., Wang, X., Wang, B., Chao, G., Zhang, Y., ... Su, H. (2025). Knowledge bridger: Towards training-free missing modality completion. In *Proceedings of the computer vision and pattern recognition conference* (pp. 25864–25873).

Khan, N., & Sisodia, D. S. (2025). Exploiting diffusion-based structured learning for item interactions representations in multimodal recommender systems. *Information Processing & Management*, *62*(3), 104075.

Kim, J., Kang, H., Kim, S., Kim, K., & Park, C. (2025). Disentangling and generating modalities for recommendation in missing modality scenarios. In *Proceedings of the 48th international acm sigir conference on research and development in information retrieval* (pp. 1820–1829).

Kim, S., Kang, H., Kim, K., Kim, J., Kim, D., Yang, M., ... Park, C. (2025). Lost in sequence:

Do large language models understand sequential recommendation? In *Proceedings of the 31st acm sigkdd conference on knowledge discovery and data mining v. 2* (pp. 1160–1171).

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Schindler, G., ... others (2023). Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*.

Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., ... others (2024). Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, *42*(8), 30–37.

Krichene, W., & Rendle, S. (2020). On sampled metrics for item recommendation. In *Kdd.*

Larochelle, H., Bengio, Y., Louradour, J., & Lamblin, P. (2009). Exploring strategies for training deep neural networks. *Journal of machine learning research*, *10*(1).

Lee, D., Kim, C., Kim, S., Cho, M., & Han, W.-S. (2022). Autoregressive image generation using residual quantization. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 11523–11532).

Lee, Y.-L., Tsai, Y.-H., Chiu, W.-C., & Lee, C.-Y. (2023). Multimodal prompting with missing modalities for visual recognition. In *Cvpr* (pp. 14943–14952).

Lester, B., Al-Rfou, R., & Constant, N. (2021). The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 3045–3059).

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... others (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, *33*, 9459–9474.

Li, C., Gan, Z., Yang, Z., Yang, J., Li, L., Wang, L., & Gao, J. (2024). Multimodal foundation models: From specialists to general-purpose assistants. *Foundations and Trends in Computer Graphics and Vision*, *16*(1-2), 1–214.

Li, H., Yuan, P., Xu, S., Wu, Y., He, X., & Zhou, B. (2020). Aspect-aware multimodal summarization for chinese e-commerce products. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 34, pp. 8188–8195).

Li, J., Li, D., Savarese, S., & Hoi, S. (2023). Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning* (pp. 19730–19742).

Li, M., Zhang, Y., Long, D., Chen, K., Song, S., Bai, S., ... others (2026). Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and

ranking. *arXiv preprint arXiv:2601.04720*.

Li, R., Deng, W., Cheng, Y., Yuan, Z., Zhang, J., & Yuan, F. (2025). Exploring the upper limits of text-based collaborative filtering using large language models: Discoveries and insights. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 1643–1653).

Li, W., Huang, R., Zhao, H., Liu, C., Zheng, K., Liu, Q., ... others (2025). Dimerec: a unified framework for enhanced sequential recommendation via generative diffusion models. In *Proceedings of the eighteenth acm international conference on web search and data mining* (pp. 726–734).

Li, X., Chen, B., She, J., Cao, S., Wang, Y., Jia, Q., ... others (2025). A survey of generative recommendation from a tri-decoupled perspective: Tokenization, architecture, and optimization. *Preprints*.

Li, Y., Du, H., Ni, Y., He, Y., Fu, J., Liu, X., & Guo, Q. (2024). An empirical study of training id-agnostic multi-modal sequential recommenders. *arXiv preprint arXiv:2403.17372*.

Li, Y., Du, H., Ni, Y., Zhao, P., Guo, Q., Yuan, F., & Zhou, X. (2023). Multi-modality is all you need for transferable recommender systems. *arXiv preprint arXiv:2312.09602*.

Li, Z., Zhu, J., Huang, C., Chen, Z., Guo, H., Ma, G., & Ling, J. (2026). Capturing dynamic user interests under modality imbalance for multimodal sequential recommendation. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 40, pp. 15198–15206).

Liang, D., Krishnan, R. G., Hoffman, M. D., & Jebara, T. (2018). Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference* (pp. 689–698).

Liang, J., Zhao, X., Li, M., Zhang, Z., Wang, W., Liu, H., & Liu, Z. (2023). Mmmlp: Multimodal multilayer perceptron for sequential recommendations. In *Proceedings of the acm web conference 2023* (pp. 1109–1117).

Liao, R., Moyer, D., Cha, M., Quigley, K., Berkowitz, S., Horng, S., ... Wells, W. M. (2021). Multimodal representation learning via maximization of local mutual information. In *Medical image computing and computer assisted intervention–miccai 2021: 24th international conference, strasbourg, france, september 27–october 1, 2021, proceedings, part ii 24* (pp. 273–283).

Lin, J., Dai, X., Xi, Y., Liu, W., Chen, B., Li, X., ... others (2023). How can recommender systems benefit from large language models: A survey. *arXiv preprint arXiv:2306.05817*.

Lin, X., Shi, H., Wang, W., Feng, F., Wang, Q., Ng, S.-K., & Chua, T.-S. (2025). Order-agnostic identifier for large language model-based generative recommendation. In *Proceedings of the 48th international acm sigir conference on research and development in information retrieval* (pp. 1923–1933).

Lin, Y.-B., Sung, Y.-L., Lei, J., Bansal, M., & Bertasius, G. (2023). Vision transformers are parameter-efficient audio-visual learners. In *Proceedings of the ieee/cvf conference on*

computer vision and pattern recognition (pp. 2299–2309).

Liu, C., Bai, Y., Zhao, X., Zhang, Y., Feng, F., & Rong, W. (2025). Discrec: Disentangled semantic-collaborative modeling for generative recommendation. *arXiv preprint arXiv:2506.15576*.

Liu, E., Zheng, B., Ling, C., Hu, L., Li, H., & Zhao, W. X. (2024). Generative recommender with end-to-end learnable item tokenization. *arXiv preprint arXiv:2409.05546*.

Liu, F., Cheng, Z., Zhu, L., Liu, C., & Nie, L. (2022). An attribute-aware attentive graph convolutional network for attribute missing in recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *34*(9), 4077–4088.

Liu, Q., Chen, N., Sakai, T., & Wu, X.-M. (2024). Once: Boosting content-based recommendation with both open-and closed-source large language models. In *Proceedings of the 17th acm international conference on web search and data mining* (pp. 452–461).

Liu, Q., Hu, J., Xiao, Y., Gao, J., & Zhao, X. (2023). Multimodal recommender systems: A survey. *arXiv preprint arXiv:2302.03883*.

Liu, Q., Zhu, J., Yang, Y., Dai, Q., Du, Z., Wu, X.-M., … Dong, Z. (2024). Multimodal pretraining, adaptation, and generation for recommendation: A survey. In *Proceedings of the 30th acm sigkdd conference on knowledge discovery and data mining* (pp. 6566–6576).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., … Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y., Yang, E., Dang, Y., Guo, G., Liu, Q., Liang, Y., … Wang, X. (2023). Id embedding as subtle features of content and structure for multimodal recommendation. *arXiv preprint arXiv:2311.05956*.

Liu, Y., Zhang, K., Li, Y., Yan, Z., Gao, C., Chen, R., … others (2024). Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*.

Liu, Y., Zhang, K., Ren, X., Huang, Y., Jin, J., Qin, Y., … Zhang, W. (2024). Alignrec: Aligning and training in multimodal recommendations. In *Proceedings of the 33rd acm international conference on information and knowledge management* (pp. 1503–1512).

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., … Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 10012–10022).

Liu, Z., Wang, S., Wang, X., Zhang, R., Deng, J., Bao, H., … others (2025). Onerec-think: In-text reasoning for generative recommendation. *arXiv preprint arXiv:2510.11639*.

Lu, J., Xiao, M., Wang, W., Du, Y., Cui, Y., Zhao, J., & Hua, C. (2024). Mufm: A mamba-enhanced feedback model for micro video popularity prediction. *arXiv preprint arXiv:2411.15455*.

Lu, W., & Yin, L. (2025). Dmmd4sr: Diffusion model-based multi-level multimodal denoising

for sequential recommendation. In *Proceedings of the 33rd acm international conference on multimedia* (pp. 6363–6372).

Lu, Z., Ie, E., & Sha, F. (2020). Mean-field approximation to gaussian-softmax integral with application to uncertainty estimation. *arXiv preprint arXiv:2006.07584*.

Ma, C., Kang, P., & Liu, X. (2019). Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining* (pp. 825–833).

Ma, M., Ren, J., Zhao, L., Testuggine, D., & Peng, X. (2022). Are multimodal transformers robust to missing modality? In *Cvpr* (pp. 18156–18165).

Ma, M., Ren, J., Zhao, L., Tulyakov, S., Wu, C., & Peng, X. (2021). SMIL: Multimodal learning with severely missing modality. In *Proceedings of the 35th aaai conference on artificial intelligence* (pp. 2302–2310).

Malitesta, D., Rossi, E., Pomo, C., Di Noia, T., & Malliaros, F. D. (2024). Do we really need to drop items with missing modalities in multimodal recommendation? In *Proceedings of the 33rd acm international conference on information and knowledge management* (pp. 3943–3948).

Meng, F., Meng, Z., Jin, R., Chen, Y., Lin, R., & Wu, B. (2025). Tamer: Interest tree augmented modality graph recommender for multimodal recommendation. In *Proceedings of the 33rd acm international conference on multimedia* (pp. 5998–6006).

Meng, L., Feng, F., He, X., Gao, X., & Chua, T.-S. (2020). Heterogeneous fusion of semantic and collaborative information for visually-aware food recommendation. In *Proceedings of the 28th acm international conference on multimedia* (pp. 3460–3468).

Mu, S., Hou, Y., Zhao, W. X., Li, Y., & Ding, B. (2022). Id-agnostic user behavior pre-training for sequential recommendation. In *China conference on information retrieval* (pp. 16–27).

Ni, J., Li, J., & McAuley, J. (2019). Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 188–197).

Ni, Y., Cheng, Y., Liu, X., Fu, J., Li, Y., He, X., … Yuan, F. (2023). A content-driven micro-video recommendation dataset at scale. *arXiv preprint arXiv:2309.15379*.

Ni, Y., Cheng, Y., Liu, X., Fu, J., Li, Y., He, X., … Yuan, F. (2025). A content-driven micro-video recommendation dataset at scale. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 6486–6491).

Paninski, L. (2003). Estimation of entropy and mutual information. *Neural computation*, *15*(6), 1191–1253.

Peng, B., Burns, B., Chen, Z., Parthasarathy, S., & Ning, X. (2023). Towards efficient and effective adaptation of large language models for sequential recommendation. *arXiv preprint*

arXiv:2310.01612.

Pfeiffer, J., Rücklé, A., Poth, C., Kamath, A., Vulić, I., Ruder, S., … Gurevych, I. (2020). Adapterhub: A framework for adapting transformers. *arXiv preprint arXiv:2007.07779*.

Qiu, X., Wei, Y., Tan, X., Xu, W., Wang, H., Ma, J., … Fang, Z. (2025). Mimar-osa: Enhancing obstructive sleep apnea diagnosis through multimodal data integration and missing modality reconstruction. *Pattern Recognition*, 111917.

Qu, Y., Qu, L., Chen, T., Nguyen, Q. V. H., & Yin, H. (2025). Efficient multimodal streaming recommendation via expandable side mixture-of-experts. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 2460–2470).

Qu, Z., Xie, R., Xiao, C., Yao, Y., Liu, Z., Lian, F., … Zhou, J. (2023). Thoroughly modeling multi-domain pre-trained recommendation as language. *arXiv preprint arXiv:2310.13540*.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., … Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the 38th international conference on machine learning* (pp. 8748–8763). PMLR.

Radford, A., Kim, J. W., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763).

Rafailidis, D., Manolopoulou, S., & Daras, P. (2013). A unified framework for multimodal retrieval. *Pattern Recognition*, *46*(12), 3358–3370.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., … Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, *21*(1), 5485–5551.

Rajput, S., Mehta, N., Singh, A., Keshavan, R. H., Vu, T., Heldt, L., … others (2023). Recommender systems with generative retrieval. *arXiv preprint arXiv:2305.05065*.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, *1*(2), 3.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., … Sutskever, I. (2021). Zero-shot text-to-image generation. In *International conference on machine learning* (pp. 8821–8831).

Ran, D., Zheng, W., Li, Y., Bian, K., Zhang, J., & Deng, X. (2022). Revenue and user traffic maximization in mobile short-video advertising. In *Aamas'22: Proceedings of the 21st international conference on autonomous agents and multiagent systems* (pp. 1092–1100).

Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 3982–3992).

Rosenfeld, J. S. (2021). Scaling laws for deep learning. *arXiv preprint arXiv:2108.07686*.

Safayenikoo, P., & Akturk, I. (2021). Weight update skipping: Reducing training time for

artificial neural networks. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, *11*(4), 563–574.

Sagar, A. D., & Najam, A. (1998). The human development index: a critical review. *Ecological economics*, *25*(3), 249–264.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., ... others (2024). Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Shi, S., Zhang, M., Yu, X., Zhang, Y., Hao, B., Liu, Y., & Ma, S. (2019). Adaptive feature sampling for recommendation with missing content feature values. In *Cikm* (pp. 1451–1460).

Skiena, S. S. (1998). *The algorithm design manual* (Vol. 2). Springer.

Song, T., Qi, L., Liu, W., Wang, F., Xu, X., Hu, H., ... Beheshti, A. (2025). Boosting guided diffusion with large language models for multimodal sequential recommendation. In *Proceedings of the 33rd acm international conference on multimedia* (pp. 6203–6212).

Stekhoven, D. J., & Bühlmann, P. (2012). Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, *28*(1), 112–118. doi: 10.1093/bioinformatics/btr597

Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. (2019). Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th acm international conference on information and knowledge management* (pp. 1441–1450).

Sun, Q., Yu, Q., Cui, Y., Zhang, F., Zhang, X., Wang, Y., ... Wang, X. (2023). Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*.

Sun, R., Cao, X., Zhao, Y., Wan, J., Zhou, K., Zhang, F., ... Zheng, K. (2020). Multi-modal knowledge graphs for recommender systems. In *Proceedings of the 29th acm international conference on information & knowledge management* (pp. 1405–1414).

Sun, W., Kong, K., Ma, X., Wang, S., Yin, D., de Rijke, M., ... Yang, Y. (2026). ZeroGR: A generalizable and scalable framework for zero-shot generative retrieval. In *The fourteenth international conference on learning representations.* Retrieved from `https://openreview.net/forum?id=RBoAwiQl5L`

Sung, Y.-L., Cho, J., & Bansal, M. (2022). Lst: Ladder side-tuning for parameter and memory efficient transfer learning. *Advances in Neural Information Processing Systems*, *35*, 12991–13005.

Takida, Y., Shibuya, T., Liao, W., Lai, C.-H., Ohmura, J., Uesaka, T., ... Mitsufuji, Y. (2022). Sq-vae: Variational bayes on discrete representation with self-annealed stochastic quantization. *arXiv preprint arXiv:2205.07547*.

Tan, H. L., Li, Z., Tan, Y. H., Rahardja, S., & Yeo, C. (2013). A perceptually relevant mse-based image quality metric. *IEEE Transactions on Image Processing*, *22*(11), 4447–4459.

Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional

sequence embedding. In *Proceedings of the eleventh acm international conference on web search and data mining* (pp. 565–573).

Tang, Z., Huan, Z., Li, Z., Zhang, X., Hu, J., Fu, C., … Li, C. (2023). One model for all: Large language models are domain-agnostic recommendation systems. *arXiv preprint arXiv:2310.14304*.

Tay, Y., Tran, V., Dehghani, M., Ni, J., Bahri, D., Mehta, H., … others (2022). Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, *35*, 21831–21843.

Tong, Z., Song, Y., Wang, J., & Wang, L. (2022). Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, *35*, 10078–10093.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., … others (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, *45*(3), 1–67. doi: 10.18637/jss.v045.i03

Van der Lee, C., Gatt, A., Van Miltenburg, E., & Krahmer, E. (2021). Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language*, *67*, 101151.

Vaze, S., Carion, N., & Misra, I. (2023). Genecis: A benchmark for general conditional image similarity. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 6862–6872).

Wang, C., Niepert, M., & Li, H. (2018). Lrmm: Learning to recommend with missing modalities. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 3360–3370).

Wang, D., Huang, Y., Gao, S., Wang, Y., Huang, C., & Shang, S. (2025). Generative next poi recommendation with semantic id. In *Proceedings of the 31st acm sigkdd conference on knowledge discovery and data mining v. 2* (pp. 2904–2914).

Wang, H., Guo, W., Zhang, L., Chin, J. Y., Ye, Y., Guo, H., … Chen, E. (2025). Generative large recommendation models: Emerging trends in llms for recommendation. In *Companion proceedings of the acm on web conference 2025* (pp. 49–52).

Wang, H., Liu, X., Fan, W., Zhao, X., Kini, V., Yadav, D. P., … Liu, H. (2025). Rethinking large language model architectures for sequential recommendations. In *Proceedings of the 14th international joint conference on natural language processing and the 4th conference of the asia-pacific chapter of the association for computational linguistics* (pp. 3376–3391).

Wang, J., Yuan, F., Cheng, M., Jose, J. M., Yu, C., Kong, B., … Li, Z. (2022). Transrec: Learning transferable recommendation from mixture-of-modality feedback. *arXiv preprint arXiv:2206.06190*.

Wang, J., Zeng, Z., Wang, Y., Wang, Y., Lu, X., Li, T., … Xia, S.-T. (2023). Missrec: Pre-training and transferring multi-modal interest-aware sequence representation for recommendation. In *Proceedings of the 31st acm international conference on multimedia* (pp. 6548–6557).

Wang, K., Shao, J., Zhang, T., Chen, Q., & Huo, C. (2023). Mpkgac: Multimodal product attribute completion in e-commerce. In *Companion proceedings of the acm web conference 2023* (pp. 336–340). doi: 10.1145/3543873.3584623

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., … Wei, F. (2022). Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Wang, R., Tang, D., Duan, N., Wei, Z., Huang, X., Cao, G., … others (2020). K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.

Wang, W., Bao, H., Lin, X., Zhang, J., Li, Y., Feng, F., … Chua, T.-S. (2024). Learnable item tokenization for generative recommendation. In *Proceedings of the 33rd acm international conference on information and knowledge management* (pp. 2400–2409).

Wang, X., Chen, H., Pan, Z., Zhou, Y., Guan, C., Sun, L., & Zhu, W. (2025). Automated disentangled sequential recommendation with large language models. *ACM Transactions on Information Systems*.

Wang, Y., Pan, J., Li, X., Wang, M., Wang, Y., Liu, Y., … Zhao, X. (2025). Empowering large language model for sequential recommendation via multimodal embeddings and semantic ids. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 3209–3219).

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, *13*(4), 600–612.

Wang, Z., Liu, B., Huang, W., Hao, T., Zhou, H., & Guo, Y. (2025). Leveraging multimodal large language model for multimodal sequential recommendation. *Scientific Reports*, *15*(1), 28960.

Wang, Z., Wan, Z., & Wan, X. (2020). Transmodality: An end2end fusion method with transformer for multimodal sentiment analysis. In *Proceedings of the web conference 2020* (pp. 2514–2520).

Wang, Z., Wang, L., Zhao, Z., Wu, M., Lyu, C., Li, H., … Tu, Z. (2024). Gpt4video: A unified multimodal large language model for lnstruction-followed understanding and safety-aware generation. In *Proceedings of the 32nd acm international conference on multimedia* (pp. 3907–3916).

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., … others (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, *35*, 24824–24837.

Wei, W., Huang, C., Xia, L., & Zhang, C. (2023). Multi-modal self-supervised learning for

recommendation. In *Proceedings of the acm web conference 2023* (pp. 790–800).

Wei, Y., Wang, X., Nie, L., He, X., & Chua, T.-S. (2020). Graph-refined convolutional network for multimedia recommendation with implicit feedback. In *Proceedings of the 28th acm international conference on multimedia* (pp. 3541–3549).

Wei, Y., Wang, X., Nie, L., He, X., Hong, R., & Chua, T.-S. (2019). Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th acm international conference on multimedia* (pp. 1437–1445).

Wei, Y., Xu, Y., Zhu, L., Ma, J., & Huang, J. (2024). Fummer: A fine-grained self-supervised momentum distillation framework for multimodal recommendation. *Information Processing & Management*, *61*(5), 103776.

Williams, J. H., Jones, R. A., Haley, B., Kwok, G., Hargreaves, J., Farbes, J., & Torn, M. S. (2021). Carbon-neutral pathways for the united states. *AGU advances*, *2*(1), e2020AV000284.

Wu, C., Wu, F., Qi, T., & Huang, Y. (2021a). Empowering news recommendation with pre-trained language models. In *Proceedings of the 44th international acm sigir conference on research and development in information retrieval* (pp. 1652–1656).

Wu, C., Wu, F., Qi, T., & Huang, Y. (2021b). Mm-rec: multimodal news recommendation. *arXiv preprint arXiv:2104.07407*.

Wu, H., Zhang, X., Tian, J., Wu, S., Dou, C., Sun, Y., & Feng, Z. (2021). A mutual information-based disentanglement framework for cross-modal retrieval. In *Neural information processing: 28th international conference, iconip 2021, sanur, bali, indonesia, december 8–12, 2021, proceedings, part iv 28* (pp. 585–596).

Wu, J., Yu, X., He, K., Gao, Z., & Gong, T. (2024). Promise: A pre-trained knowledge-infused multimodal representation learning framework for medication recommendation. *Information Processing & Management*, *61*(4), 103758.

Wu, L., Yang, Y., Chen, L., Lian, D., Hong, R., & Wang, M. (2020). Learning to transfer graph embeddings for inductive graph based recommendation. In *Proceedings of the 43rd international acm sigir conference on research and development in information retrieval* (pp. 1211–1220).

Wu, W., Zhu, Z., & Shou, M. Z. (2025). Automated movie generation via multi-agent cot planning. *arXiv preprint arXiv:2503.07314*.

Xie, Z., Wei, Y., Cao, H., Zhao, C., Deng, C., Li, J., … others (2025). mhc: Manifold-constrained hyper-connections. *arXiv preprint arXiv:2512.24880*.

Xu, J., Chen, Z., Yang, S., Li, J., Wang, H., & Ngai, E. C. (2025). Mentor: multi-level self-supervised learning for multimodal recommendation. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 39, pp. 12908–12917).

Xu, M., Zhang, Z., Wei, F., Hu, H., & Bai, X. (2023a). San: Side adapter network for open-vocabulary semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine*

Intelligence.

Xu, M., Zhang, Z., Wei, F., Hu, H., & Bai, X. (2023b). Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 2945–2954).

Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., ... others (2024). Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Yang, D., Chen, K., Rao, J., Guo, X., Zhang, Y., Yang, J., & Zhang, Y. (2023). Tackling vision language tasks through learning inner monologues. *arXiv preprint arXiv:2308.09970*.

Yang, J., Duan, J., Tran, S., Xu, Y., Chanda, S., Chen, L., ... Huang, J. (2022). Vision-language pre-training with triple contrastive learning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 15671–15680).

Yang, W., Zhong, R., Chen, Y., Li, S., Ping, H., Lu, C., & Jiang, P. (2025). Fitmm: Adaptive frequency-aware multimodal recommendation via information-theoretic representation learning. In *Proceedings of the 33rd acm international conference on multimedia* (pp. 6193–6202).

Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., ... others (2024). Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*.

Ye, Y., Fu, J., Song, Y., Zheng, K., & Jose, J. M. (2026). Are multimodal embeddings truly beneficial for recommendation? a deep dive into whole vs. individual modalities. In *European conference on information retrieval* (pp. 66–81).

Ye, Y., Zheng, Z., Shen, Y., Wang, T., Zhang, H., Zhu, P., ... Xiong, H. (2025). Harnessing multimodal large language models for multimodal sequential recommendation. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 39, pp. 13069–13077).

Yi, X., Yang, J., Hong, L., Cheng, D. Z., Heldt, L., Kumthekar, A., ... Chi, E. (2019). Sampling-bias-corrected neural modeling for large corpus item recommendations. In *Proceedings of the 13th acm conference on recommender systems* (pp. 269–277).

Yin, D., Yang, Y., Wang, Z., Yu, H., Wei, K., & Sun, X. (2023). 1% vs 100%: Parameter-efficient low rank adapter for dense predictions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 20116–20126).

Yuan, Z., Yuan, F., Song, Y., Li, Y., Fu, J., Yang, F., ... Ni, Y. (2023). Where to go next for recommender systems? id-vs. modality-based recommender models revisited. In *Proceedings of the 46th international acm sigir conference on research and development in information retrieval* (pp. 2639–2649).

Zaken, E. B., Ravfogel, S., & Goldberg, Y. (2021). Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.

Zhai, P., Yuan, Y., Di, F., Li, J., Liu, Y., Li, C., ... Li, X. (2025). A simple contrastive framework of item tokenization for generative recommendation. *arXiv preprint arXiv:2506.16683*.

Zhang, J., Cheng, Y., Ni, Y., Pan, Y., Yuan, Z., Fu, J., ... Yuan, F. (2023). Ninerec:

A benchmark dataset suite for evaluating transferable recommendation. *arXiv preprint arXiv:2309.07705*.

Zhang, L., Song, K., Lee, Y. Q., Guo, W., Wang, H., Li, Y., … Chen, E. (2025). Killing two birds with one stone: Unifying retrieval and ranking with a single generative recommendation model. In *Proceedings of the 48th international acm sigir conference on research and development in information retrieval* (pp. 2224–2234).

Zhang, L., Zhou, X., & Shen, Z. (2023). Multimodal pre-training framework for sequential recommendation via contrastive learning. *arXiv preprint arXiv:2303.11879*.

Zhang, L., Zhou, X., Zeng, Z., & Shen, Z. (2024). Multimodal pre-training for sequential recommendation via contrastive learning. *ACM Transactions on Recommender Systems*, *3*(1), 1–23.

Zhang, R., Fang, R., Zhang, W., Gao, P., Li, K., Dai, J., … Li, H. (2021). Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 586–595).

Zhang, S., Chen, L., Shen, D., Wang, C., & Xiong, H. (2025). Hierarchical time-aware mixture of experts for multi-modal sequential recommendation. In *Proceedings of the acm on web conference 2025* (pp. 3672–3682).

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zhang, X., Xu, B., Ma, F., Li, C., Yang, L., & Lin, H. (2023). Beyond co-occurrence: Multi-modal session-based recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Zhang, Y., Feng, F., Wang, C., He, X., Wang, M., Li, Y., & Zhang, Y. (2020). How to retrain recommender system? a sequential meta-learning method. In *Proceedings of the 43rd international acm sigir conference on research and development in information retrieval* (pp. 1479–1488).

Zhang, Y., Zhang, R., Guo, J., Peng, W., Li, S., Lv, F., & Cheng, X. (2025). C2t-id: Converting semantic codebooks to textual document identifiers for generative search. In *Proceedings of the 2025 annual international acm sigir conference on research and development in information retrieval in the asia pacific region* (pp. 331–336).

Zhang, Z., Cao, S., & Wang, Y.-X. (2024). Tamm: Triadapter multi-modal learning for 3d shape understanding. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 21413–21423).

Zhang, Z., Jia, Z., Li, C., Zhou, Y., Liu, X., Min, X., & Zhai, G. (2026). Towards versatile multimedia quality assessment for visual communications. *Science China Information Sciences*, *69*(2), 122306.

Zhang, Z., Wang, J., Guo, Y., Wen, F., Chen, Z., Wang, H., ... others (2025). Aibench: towards trustworthy evaluation under the 45° law. *Displays*, 103255.

Zhang, Z., Wang, J., Wen, F., Guo, Y., Zhao, X., Fang, X., ... others (2025). Large multimodal models evaluation: a survey. *Science China Information Sciences*, *68*(12), 221301.

Zhang, Z., Zhao, D., Miao, X., Oliaro, G., Zhang, Z., Li, Q., ... Jia, Z. (2024). Quantized side tuning: Fast and memory-efficient tuning of quantized large language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1–17).

Zhao, B., Zhang, W., & Zou, Z. (2025). Mce: Towards a general framework for handling missing modalities under imbalanced missing rates. *Pattern Recognition*, 112591.

Zhao, J., Wang, W., Xu, C., Chen, X., Ren, Z., & Verberne, S. (2025). Unifying search and recommendation: A generative paradigm inspired by information theory. *arXiv preprint arXiv:2504.06714*.

Zhao, J., Wang, Z., Pan, S., Verberne, S., & Ren, Z. (2026). Unifying search and recommendation in llms via gradient multi-subspace tuning. *arXiv preprint arXiv:2601.09496*.

Zhao, W., Peyrard, M., Liu, F., Gao, Y., Meyer, C. M., & Eger, S. (2019). Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 563–578).

Zheng, B., Liu, E., Chen, Z., Ma, Z., Wang, Y., Zhao, W. X., & Wen, J.-R. (2025). Pretraining generative recommender with multi-identifier item tokenization. *arXiv preprint arXiv:2504.04400*.

Zhong, T., Lang, J., Zhang, Y., Cheng, Z., Zhang, K., & Zhou, F. (2024). Predicting micro-video popularity via multi-modal retrieval augmentation. In *Proceedings of the 47th international acm sigir conference on research and development in information retrieval* (pp. 2579–2583).

Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., ... Wen, J.-R. (2020). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th acm international conference on information & knowledge management* (pp. 1893–1902).

Zhou, P., Liu, C., Ren, J., Zhou, X., Xie, Y., Cao, M., ... others (2025). When large vision language models meet multimodal sequential recommendation: An empirical study. In *Proceedings of the acm on web conference 2025* (pp. 275–292).

Zhou, P., Wang, L., Liu, Z., Hao, Y., Hui, P., Tarkoma, S., & Kangasharju, J. (2024). A survey on generative ai and llm for video generation, understanding, and streaming. *arXiv preprint arXiv:2404.16038*.

Zhou, X. (2023). Mmrec: Simplifying multimodal recommendation. In *Proceedings of the 5th*

acm international conference on multimedia in asia workshops (pp. 1–2).

Zhou, X., & Shen, Z. (2023). A tale of two graphs: Freezing and denoising graph structures for multimodal recommendation. In *Proceedings of the 31st acm international conference on multimedia* (pp. 935–943).

Zhou, X., Zhou, H., Liu, Y., Zeng, Z., Miao, C., Wang, P., ... Jiang, F. (2023). Bootstrap latent representations for multi-modal recommendation. In *Proceedings of the acm web conference 2023* (pp. 845–854).

Zhou, Y., Lei, T., Liu, H., Du, N., Huang, Y., Zhao, V., ... others (2022). Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, *35*, 7103–7114.

Zhou, Y., Ren, T., Zhu, C., Sun, X., Liu, J., Ding, X., ... Ji, R. (2021). Trar: Routing the attention spans in transformer for visual question answering. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 2074–2084).

Zhou, Z., Meng, X., & Zhang, Y. (2025). Rethinking convolutional neural network in multimodal sequential recommendation. *ACM Transactions on Information Systems*, *44*(2), 1–35.

Zhu, D., Huang, H., Huang, Z., Zeng, Y., Mao, Y., Wu, B., ... Zhou, X. (2024). Hyper-connections. *arXiv preprint arXiv:2409.19606*.

Zhu, J., Ju, M., Liu, Y., Koutra, D., Shah, N., & Zhao, T. (2025). Beyond unimodal boundaries: Generative recommendation with multimodal semantics. *arXiv preprint arXiv:2503.23333*.

Zhu, Y., Zhao, H., Zhang, W., Ye, G., Chen, H., Zhang, N., & Chen, H. (2021). Knowledge perceived multi-modal pretraining in e-commerce. In *Proceedings of the 29th acm international conference on multimedia* (pp. 2744–2752).

Zhuang, Z., Li, H., Fu, J., Liu, J., Jose, J. M., Li, Y., & Ni, Y. (2025). Frequency-decoupled distillation for efficient multimodal recommendation. In *Proceedings of the 34th acm international conference on information and knowledge management* (pp. 4571–4581).