The ostensive model of developing information-needs

Iain Campbell
iain@iaincampbell.com

Submitted for the degree of Doctor of Philosophy
to the University of Glasgow
from the Department of Computing Science
September 2000

UNIVERSITY
of
GLASGOW

Acknowledgements

I'd like to thank the following:

Keith van Rijsbergen – this man has done many tangible things for me, and for this thesis. However, that pales in comparison to the *influence* that he has had on me. I am proud to regard him as my intellectual and scientific mentor.

Mutti & Dad – for their constant and unquestioning love and support; and for keeping under control their natural parental concern and frustration over my rather itinerate academic career – well, *almost* under control.

Denise Barrault – for love, affection, and support; and for an endless supply of top-notch French cooking during the write-up.

Jarle Vestrheim & **Guy Fletcher** – for being my two best friends; but definitely not for the goading and abuse that they have dealt me over the course of this work.

Mark Sanderson – for being a friend; for being inspirational; for constant encouragement; and for reading 10,000 crap papers on IR so that I only had to read the 10 good ones.

Peter Ingwersen – for breaking a stereotype of mine, by proving that Information Science really can offer both theoretical and operational advances in IR; and for taking such an active and inspiring interest in my work.

The Glasgow IR group – (and all the honorary members from around the world who have spent time with us) for a stimulating scientific and fun social environment.

Bruce Croft – for having the confidence in me to only bet £20 that I'd never submit this thesis.

David Harper[1] – for not having such confidence, and consequently owing me a tidy £500.

---

[1] The same frustration at my apparent lack of determination, that drove Bruce and David to offer those incentives, also drove Keith to bet me £5,000 on the same evening. I have never seen three men so keen to lose their money – it was genuinely touching. The only person to *truly* display lack of confidence was the idiot who accepted the £20, then the £500, but finally crumbled in the face of the big money.

Abstract

From intuitions and informal observations of searching behaviour, a formal model is developed of cognition during a searching session. The model is of the iterative updating of an information-need by exposure of a user to information during a session.

The model is path-based – using trends within the content of objects on a path to predict the current information-need. This provides contextual interpretation of objects based upon the path taken to an object. The model is ostensive in nature; however, instead of the active communicated evidence of traditional conceptions of ostension, it uses passive observational evidence. It produces a new notion of relevance: Ostensive Relevance – profiles of which are the key to the effective use of path information.

The integration of the Ostensive Model and the Binary Probabilistic Model is achieved by weakening of a conventional assumption in the estimation of a probabilistic parameter. This integration effects a novel combination of objective and subjective probabilities – commonly regarded as incompatible.

The Ostensive Model is instantiated in a combination of a networked IR server and a novel graphical user-interface. The interface presents a fish-eyed view of a growing multi-path browsing surface that hides internal representations and obviates querying. The hiding of internals, combined with the ability of the Ostensive Model to follow a developing information-need, makes the interface a truly media-neutral searching environment.

A new test collection of general interest images with four binary relevance assessments is constructed and used for an evaluation of three Ostensive Relevance Profiles. The results are analysed in the light of different interpretations of the multiple assessments of the test-collection. The evaluation method is itself analysed and concrete proposals made for its development. The results of the evaluation provide strong encouragement for the Ostensive approach.

Organisation of the thesis

After the introduction, the work is split into ten chapters. Each chapter stands more or less on its own – with an introduction describing the contents and a summary listing the major points. To provide higher structure, the chapters are grouped into five parts:

Part I: **The Conception** (Chapters 1, 2, & 3)

Intuitions, observations, and an analysis of the current approaches to supporting information seeking are presented. These culminate in a proposal for a new, path-based, approach.

Part II: **The Model** (Chapters 4 & 5)

The ideas presented in Part I are formalised into a new model of information-needs – the Ostensive Model. That model is then integrated with an existing operational model that has some intuitively appealing properties.

Part III: **The System** (Chapters 6 & 7)

Two components are described: a retrieval engine that implements the model of Part II, and a graphical user-interface that implements the ideas of Part I and the model of Part II.

Part IV: **The Evaluation** (Chapters 8 & 9)

The construction is described of an image test-collection that was built specifically to test the Ostensive Model. Using the system of Part III, and the new test collection, an evaluation is presented of core components of the Ostensive Model.

Part V: **The Conclusions** (Chapter 10)

The individual achievements are listed, as are the overall achievements, highlighting suggested improvements where appropriate. Finally, some directions for further work are presented.

Contents

Introduction

The destination

The goal of this work was to improve the experience of a person searching for information using an Information Retrieval system – with a focus on the tasks that the person must perform along the way. The change in the nature of those tasks can be summed up as:

A shift in effort from things 'procedural' to things 'functional'.

That is, a reduction in a user's cognition of, and manipulation of, artefacts of the retrieval system; replacing them with a corresponding concentration upon the quintessential task of knowledge discovery (and to a lesser extent, relevance assessment). To achieve that goal, the following sub-goals were identified:

- To gather evidence of an information-need from a user in a query-less and passive manner.
- To combine and utilise the gathered evidence in manner that would most effectively predict which information objects to offer the user.
- To design a searching environment that hides completely from the user any internal representation and retrieval techniques.

The route

The starting point of the work is the recognition that information-needs are developing, inaccessible, and observable only retrospectively through their external physical effects.

The idea that information-needs change over time has been recognised by many authors (e.g. [Belkin82], [Ingwersen92], [Ingwersen96]). Nevertheless, such ideas of change in an information-need have, to varying extents, been loaded with a judgemental notion of objective 'improvement' – a loading that is absent in the ideas presented in this thesis. For example, one could argue that the notion of information-need presented here is consistent with the range of "intrinsically ill-defined needs" presented those works; but the idea of being ill- or well-defined makes no sense if one accepts the proposition of this thesis that an information-need is inaccessible.

At the other end of the spectrum, work such as that of [Bates89] are free of such judgemental or objectivist notions. Nevertheless, they are loaded in a complementary manner, in that they lack any sense of inherent direction or intent in the changes.

Those loadings make both approaches difficult to apply in any operational sense. They are not consistent with the conception of information-needs made here, where there is a sense of 'development', but in a wholly neutral and user-subjective sense – in fact, 'evolution' might be a term that is appropriate. That is, an information-need changes in any manner that the user perceives, at that time, to most effectively lead to a natural end-point of either satisfaction, or redundancy. Nothing is assumed beyond that drive to an end-point – e.g. nothing regarding any environmental influences that might assert themselves upon the user.

This work makes explicit the notion that an information-need is inseparable from the overall mind state of the searcher, in fact, that they are one and the same. The work also makes explicit that an information-need only makes sense as a retrospective analytical notion. As Relevance is wholly a product of the information-need, it too is retrospective and analytic in nature. This view of Relevance is in sharp contrast to ideas common in, for example [Saracevic96], where Relevance is spoken of as if it were some real function that is actively and consciously applied by a searcher during the searching process.

The distanced perspective on both information-needs and Relevance, and the recognition of their neutral development, provide motivation for a number of terms of reference that are used to analyse existing searching approaches. The two classes of approach (Query-based and Browse-based) are comparatively analysed with respect to those terms.

Combination of the graphical interface of the browse approach (which hides representation issues) with the dynamic document space of the query approach (which supports the development of an information–need) is achieved by identifying paths of relevant-indicated objects as a physical manifestation of the development of the information-need. These paths are used to drive the support provided by a retrieval system. The use of path information as a surrogate of the information-need (in fact, as the physical manifestation of the development of the information-need) is formalised as the Ostensive Model.

The Ostensive model is a cognitive model that motivates the use of evidence from path-based passive relevance indications to predict objects that are most likely to be relevant to the user. In essence, it models the iterative process of: a user choosing a document; reading it; the change of the information-need as a result of that reading; and the modified choosing behaviour as a result of the changed information-need. It proposes why, operationally, one should consider a document recently marked as relevant as being more indicative of the *current* information-need than one marked some time before.

The classical conceptions of Ostension are based upon actively communicated exemplars – i.e. it is based upon actions, the intentions behind which are to communicate a notion or definition. The Ostensive Model broadens that classical conception to include passive observational evidence – i.e. where no such intention is assumed present.

The model provides a new conception of relevance: Ostensive Relevance. It is essentially an operational conception that directly recognises the distance at which Relevance is experienced by any observer (human or system). It also recognises that Relevance is an analytic notion whose purpose is purely to aid in the predictive support for user searching.

One intriguing aspect of the conception is that, although the definition is specific at both the conceptual and the operational levels, it can be thought of as subsuming all the members of the 'subjective' class of relevance conceptions as discussed in [Saracevic96] and [Cosijn2000]. This arises quite simply through its basis being a distanced and retrospective view of the information-need and through not attempting to deconstruct the 'nature' or 'origin' of the information-need.

There is a temporal discounting of Ostensive Relevance – i.e. older relevance observations are less indicative of the current state of the information-need than more recent observations, and hence less indicative of the objects (unseen by the user) that are likely to be regarded as relevant. The discounting of evidence results in Ostensive Relevance Profiles, which determine the precise relationship between the discounting of evidential weight and its age.

Discounting of evidence is in contrast to the assumption of equality of evidential weight made in the traditional conception of Relevance Feedback. The Binary Probabilistic implementation of Relevance Feedback is modified to remove that assumption and allow it to incorporate any range of evidential weights. This integration of the Ostensive Model and the Probabilistic Model permits various Ostensive Relevance Profiles to be applied to the evidence, and allows an operational retrieval system to be constructed.

The integrated retrieval model is implemented in a working retrieval system with a novel graphical browsing environment. The browser instantiates directly the path aspect of the Ostensive Model, and demonstrates the media-neutrality of the Ostensive approach through its complete hiding of internal object representations and retrieval algorithmics.

To evaluate the Ostensive Model using a non-textual medium, a test-collection of over six hundred images and thirty queries with relevance-assessments is constructed. The relevance assessments are novel in that four separate assessments were made for each query and image pair. Therefore, for each query, each image has an associated count from zero to four indicating the number of assessors who regarded it to be relevant. The multiple assessments are an attempt to capture some of the subjectivity (and error) that is inherent in relevance assessments – in particular, to capture the distribution of relevance for a particular target population.

Gathering multiple assessments require many more people than single assessments, so the approach used to obtain those assessments is designed to be low cost and high speed. In addition, the approach is shown to be extensible – allowing additional levels of assessment to be added later.

An evaluation is carried out of three Ostensive Relevance Profiles: a profile representing the antithesis of the Ostensive Model (i.e. the Relevance *increasing* with age); a flat profile corresponding to the traditional Relevance Feedback model; and the 'preferred' increasing-with-age Ostensive Model profile. Beyond the straightforward profile comparisons, the evaluation proved to be a rich source of information for the design of a larger, more specific experiment.

The evaluation consists of Recall-oriented analyses of the effectiveness of users when using the three test profiles. Recall is measured with respect to the test collection, with each image having a degree of relevance based upon its popularity amongst the test-collection assessors. The multiple assessments of the test collection require interpretation and transformation into some measure of Relevance before they can be used in such measurements.

Eight interpretation functions are used – four binary thresholds and four continuous functions. The binary thresholds set Relevance to be true or false depending upon the number of assessors that agreed on its relevance, whereas the continuous functions set the Relevance to be some value between zero and a maximum depending upon how many assessors agreed. Essentially, each function instantiates different intuitions on what the assessor-count means.

Three Recall analyses are presented: average Recall; counts of best Recall performance; and statistical significance on coordinated average Recall. Each tells a slightly different story and has different strengths – confirming that any one analysis is not sufficient.

The eight assessor-count-interpretation functions are shown to affect the relative effectiveness of the profiles, but not the rank ordering. Further, they are shown to affect the degree of statistical significance achieved. This confirms that any one of these functions is also not sufficient to perform an effective analysis.

The first two analyses show the Ostensive Model's discounting profile to be superior to the other profiles. The third analysis showed only weak statistical significance in its comparisons. Taken as a whole, the evaluation does not *conclusively* show that the

Ostensive Model profile is better than the traditional approach, but it does provide strong evidence in its favour.

Summary

Overall, this thesis charts the development of an idea. It starts with informal observations made whilst using an Information Retrieval system. The observations lead to a recognition of complementarity in existing approaches. The idea is formalised as a cognitive model using a classical notion of conceptual inference. That model is then integrated with an existing probabilistic model to give an operational one. A novel media-neutral browsing environment is built that instantiates the models as a hybrid of the existing approaches. Finally, an evaluation is carried out, using the new system, that shows the idea to be promising.

The Ostensive Model, the Ostensive Browsing Environment, and the use of Ostension in general, appear to have great potential in terms of both theoretical and operational development. This thesis concludes by presenting some of that potential.

Part I:

The Conception

The chapters of this part are developments of the work published in [Campbell95].

In this Part, I describe the characteristics of information-needs that I consider to be the most important for their effective modelling in Information Retrieval (IR). Using these characteristics as terms of reference, I highlight the inadequacies of current IR approaches. I propose a new approach that not only is a hybrid of the positive aspects of the existing approaches, but also embodies a significant and novel extension – the ostensive definition of developing information needs. The new approach embodies interesting properties, such as the contextual interpretation of information items. I outline these properties and explain why I feel it promises a more appropriate model upon which to base an environment for searching tasks. It has important media, language, and domain independence characteristics that make it particularly suitable for the next generation of information systems.

1 A view of information-needs

When first setting out to track down information, be it an answer to a specific question, or general information on a topic of interest, one often perceives of an 'information-need'. Such information-needs are combinations of ideas such as what the target information might look like, where it might be found, or how one might go about tracking it down – with the words *look*, *where*, and *how* used in a most general sense. It is difficult, both for an observer and for the person involved, to determine the information that constitutes those ideas, and further, to determine the various proportions of each that make up the currently perceived information-need.

This chapter discusses two ideas around information-needs and how they are at odds with assumptions that we appear to routinely make when interacting with IR systems. From those ideas, a characterisation of information-needs is presented that, although recognising the inaccessibility of an information-need, exposes structural and operational aspects. That characterisation will be used for analysis and comment throughout this thesis.

Contents of this chapter

Section 1.1 discusses the idea of information need description and our willingness to make such descriptions. Section 1.2 then presents the characterisation of information-needs that will form the basis for a comparative analysis in Chapter 2.

1.1 Describing information-needs

Attempting to describe an information-need effectively to another intelligent human being can be a difficult enough task to perform. Upon casual analysis, trying to do the same to a simplistic, uninformed, inexperienced computer system could be regarded as plainly ridiculous. Putting that slightly glib comment to one side, the thought has within it two ideas that are the subject of this chapter, and arguably key to understanding the nature of information seeking. The first is the whole idea of describing an information-need, and the second is our apparent willingness to make such descriptions to systems known to be so seriously flawed.

Before talking further about an information-need, perhaps it would be better to consider an information *lack*. It is proposed that it is the perception of a lack of information that provokes one to then develop a need for it. It is a simple fact that information one does not currently have is information that one cannot describe in its entirety. A full description of something cannot be *made* by anything; it can only *be* the item itself. The existence of something, whether conceptual or physical, *is* the only full description of that thing. We can attempt to describe it in many ways, for example: Physically – in terms of light reflected from it, sound emitted from it etc; Compositionally – in terms of its constituent parts; Functionally – in terms of what it can or cannot do; Procedurally – in terms of how it does things. Such descriptions can only be partial descriptions – i.e. reliant upon some restricted view of the world, and as such, laden with assumptions and simplifications.

All physical objects can be thought of as information – e.g. a particular arrangement of elementary particles and waves. Concepts can also be thought of as particular arrangements of other, perhaps elementary, concepts. To these ideas of arrangement, one can add the idea that a particular arrangement may change over time and still be regarded as the same thing. All these arrangements, and arrangements of arrangements are information. Everything can be regarded as information, and hence information can be thought of as a *thing* in the purest and most general sense. Finally, recognising that a particular arrangement can only *be* something if an observer regards it as such, the 'nature' of a thing can be recognised as a totally subjective concept and impossible to tie down.

From the above, it can be seen that information that is perceived to be lacked can neither be known nor described. Only the fact that something is missing can be recognised. We often make estimates as to the size of this lacking, but as no clear concept of a quantitative measure for such information has been developed or distilled from observation, such estimates have little meaning[1]. Asking someone to describe the information that they perceive as lacking can be a frustrating and exasperating experience for both the questioner and the questioned in all but the most trivial of situations. Given that the information lack is elusive and unquantifiable, it follows that the corresponding information-need is equally elusive. It would seem then, that the idea (popular in IR) of a clearly defined, or well understood, information-need is, at best, misleading.

This calls into question the idea that an agent can have an identifiable goal or target (in terms of information) to which its information seeking activities are aimed. Only full retrospective knowledge can identify the information that ultimately satisfied an information lack. That information can *only then* be regarded as the agent's original information-need, and thus the target of their previous information seeking activities.

The point being made is that it is an information-*lack* (and its perception) that is the originator and motivator of the searching activity. An information-need is that which is lacking, and that can be said loosely as being sought. This distinction is a subtle one. It was introduced purely to highlight the problems associated with describing something we do not have – where "not having" means not knowing what it is.

Having made the point, but accepting that talking of "information-needs" is traditional in the field of IR and accepting that it is an arguably less clumsy term, it shall be used from now on.

Let us imagine a cognitive agent capable of perceiving and acting upon information-needs. Let us imagine further that immediately after an information-need has been perceived, all flow of information to the perceiving agent stops and all flux of information within that agent ceases. It is obvious that, under such circumstances, the

---

[1] The informal and abstract notions of information made here are distinct from the concrete notions of Information Theory for which there are measures [Shannon49]

agent will never obtain the missing information. If the restriction on internal flux is lifted, then it is possible that by generating new information internally the agent may derive what was perceived as missing (this ignores the issue of whether, in the absence of external confirmation, the agent would be able to recognise the information as that was originally missing). Although possible, such purely internal situations are not of particular interest to us as we are concerned primarily with the flow of information into and out of the agent. We wish to optimise that flow such that it leads to effective (in terms of speed and accuracy) resolution of information-needs.

With no flow of information into or out of an agent, it will not be able to obtain the information that it requires to resolve its information-need. As soon as information begins to flow into the information-seeking agent, it will be processed. This processing, whatever its form, will naturally be considered in the resolution of the information-need. Two clear applications can be seen for the incoming information – firstly, it will be evaluated to see if it is, or is part of, the information being sought, and secondly, it will be incorporated into the general store of information that makes up the agent's view of the world.

The general store of information is a result of all the information that has previously flowed into the agent – i.e. the agent's previous experiences. Given a different store, the agent would perceive and act differently as a result of a given input of information. With every experience, the store of information will change – even 'identical' experiences (if such a thing were possible) would provoke minimally a reinforcement of a previous pattern. Therefore, the instantaneous state of a store can be regarded as a specification or definition of the agent at that particular time. This will be referred to as simply the current 'state' of the agent.

An agent's perception of an information-need will be derived from its state. As this state will be changing as it is exposed to information during its information seeking activities, its perception of the information-need will change also.

During seeking activity, an agent will be exposed to much information – a high proportion of which will be similar to, or related to, the information being sought.

This proportion will be significantly higher than would be the case during some other arbitrary activity.

An agent (acting reasonably) will actively be trying to improve its chances of locating the desired information, and so will be most receptive to any such similar or related information encountered along the way. Being 'receptive' means making great effort to incorporate encountered information into its store and using it to both reappraise/redefine its information-need, and to process any subsequently encountered information. As a result, we can expect the agent's perception of its information-need to be affected to a greater extent by the changes of its state provoked by these exposures, than by those provoked by other experiences.

From the above talk of 'perceived' information-needs, it might be thought that a disembodied, abstract information-need exists, and that an agent merely constructs a subjective, dynamic view of it. This would be to misunderstand the nature of the thing. The information-need is only present relative to, and contained within, the state of the agent. If the perception of an information-need were to somehow spontaneously disappear from the state of an agent, then the information-need itself would disappear also. The information-need and its perception are one and the same.

Given that the perception of an information-need *is* the need, it can be seen that its current and future form is dictated by the perceiving agent's state. Therefore, we could consider the information-need as part of that state. That state will change as a result of internal activity and through the agent's perceptions of the outside world. Therefore, to capture the information-need fully we would have to include the whole of the agent's state. This leads us to the idea that the agent's state and the information-need being one and the same. Therefore, if we talk of changes to the agent's state we imply changes to the information-need – and vice versa.

The agent's behaviour (i.e. responses to external situations) is determined by its state. It seems reasonable to assume that those parts of the state most related to the current activity are those that would most influence the behaviour. It would then be consistent that the information-need would affect behaviour related to itself (e.g.

information seeking behaviour) more than other behaviours, and further that it would have the primary effect for that particular behaviour.

As any cognitive activity on the part of the agent will result in changes in its state, any attempt by the agent to describe its currently perceived information-need will result in a change of the state of the agent. This change is most likely to be in the areas relating to the information-need, as those would be the 'thoughts' that would be primarily processed and analysed in the description attempt. The identity between the information-need and the agent's state means that the information-need itself will be modified by this attempt. This implies that a description attempt will result in a description (regardless of its completeness) that is both inaccurate and immediately outdated.

Willingness to describe information-needs

If one accepts the idea presented above that an information-need is a constantly changing, inaccessible phenomena present only in the mind of the searching agent, it becomes clear that it is something that we cannot, and could never, capture. Why is it then, that we are so willing to attempt to describe such things? Further, why do we build computer-based searching systems that rely upon such clearly inaccurate descriptions? The answers to these two questions are directly linked and mutually supporting. The reliance of the vast majority of searching systems upon descriptive queries fuels the expectation and reinforces the belief that this traditional approach is a 'natural' approach. It is natural only in the sense of it being simple and immediately understandable. This simplicity and understandability is a procedural one – describing (effectively, asking for) something that we want is a process that is familiar to us in our daily lives. Is the naturalness of this act misleading?

In our day-to-day lives, we ask for things by communicating a partial description of the desired item to others. We might originally have done this as cave men by pointing and grunting if the item in question was in sight. Requesting items that were not directly in sight or that were more abstract in nature would have been extremely difficult, if not impossible. Languages developed from the simple grunting and pointing to support such sophisticated communication needs. Nowadays we find it trivial to request things that are not directly accessible, or things that are purely abstract, by communicating a descriptive request to another person.

Communication can be viewed as a tool that is applied only when deemed necessary or appropriate – i.e. when it is not possible to obtain what is desired directly. For example, perceiving a need for a pencil and quickly identifying one upon a nearby desktop will result in the need being satisfied without communication. Communication would be resorted to only when a pencil could not be identified immediately (e.g. as a result of a clutter of other items), and the cost (in terms of effort, time, and likeliness of failure) of performing a search of the desktop were perceived to be greater than that of formulating an appropriate request and communicating it to an appropriate person (e.g. turning round and asking an office-mate for one).

Although we may not realise it, we automatically accept and deal with the fact that there are costs involved in descriptive communication. These costs determine our confidence in obtaining the information we wish. The costs involved when dealing with other humans are relatively low as a result of the degree of common experience and common intention. The basis for this has been described as 'Shared Cognitive Structures' by de May [deMay80] – i.e. the similar experiences, similar driving forces, similar manner of arranging those things, and finally, similar manners of cognition (i.e. computation) of them. Descriptive communication between such similar individuals is part of our everyday existence. As a result, it is possible that we fail to identify the difference between knowing something and describing it, similarly, we fail to identify the degree of similarity between us and the people whom we ask. That similarity cannot be said to exist between us and the computing systems that we use.

1.2 A characterisation of information-needs

Given the ethereal and intangible nature of information-needs presented in Section 1.1, useful descriptions of them can only be achieved from a distance. This section presents a characterisation that tries to retain such a distance from details and talk only of things that can actually be observed (to varying extents) from the 'outside'.

It is similar in approach to that of Bates when she talks of 'Berrypicking' [Bates89], where the searching user is seen to (apparently randomly) move from information-source to information-source, gathering snippets of information as they go, until the information-need is ultimately satisfied. What is missing, arguably, from such accounts is a suggestion as to some regularity or predictability in the behaviour, or in things associated with the behaviour, that might be harnessed in an operational sense. The characterisations presented here are attempts to do just that, by presenting changes in information-needs that might be caused by, or be the cause of, the user's behaviour.

The characterisation is considered within the environment of performing a searching task. They are not different 'types', but characteristics of, information-needs.

Three structural characteristics are identified (i.e. *development*, *multiplicity*, and *tangentiality*), along with two operational ones (i.e. *embedding* and *threading*):

Information-needs are developing

The first characteristic is that of the 'developing' or 'evolving' nature of an information-need during a search. This was motivated in the previous section, where it was described that an information-need changes constantly as a result of the exposure of its perceiving agent to information. In particular, that happens to information closely related to that being sought (or more specifically: thought, by the agent, to be close).

As an agent progresses with its search, its internal representation of the information-need (which, as outlined already, is equivalent to the need itself) is augmented by the encountered information. It could be 'augmented' in the sense of the provision of evidence to support or deny beliefs in various aspects of the need – such as the need's importance to the agent and its higher goals, likelihood of satisfaction in general, or in the likelihood of satisfaction within the current environment, etc. It could be augmented simply by the reduction in its 'size' through partial and piecemeal satisfaction. This process continues until the information-need reaches its most developed state – i.e. precisely at the instant when it disappears as a result of either satisfaction, or of it no longer having a perceived importance to the agent.

The point is that the information-need is not merely changing in a random or arbitrary manner, it is developing in an improving and refining sense – heading inexorably towards a end-point of either satisfaction or redundancy. It is perhaps worth pointing out that this end-point is simply the final state that the information-need happens to reach, and it is not in any sense an expected, predictable, or predefined 'final goal' for its existence. Actual end-points can only be discussed in the light of full retrospective knowledge.

In the following discussions, diagrams are used to assist in the description of the individual characteristics. In them, an information-need is presented as a line segment – running from the information-need's inception, through its development, to its end-point of satisfaction or redundancy (Fig 1.1).

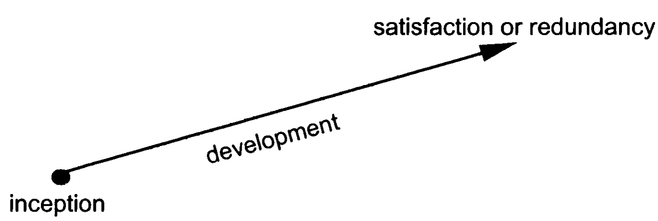


Fig 1.1 A developing information-need.

It is not possible to determine absolutely a 'direction' for an information-need; therefore, this characterisation does not attempt to describe one. Direction will only used in the weakest relative sense – i.e. to recognise that a *difference* in direction exists.

Information-needs are multiple

Within the context of a particular searching session, it is unlikely that an agent will have a single information-need that will be followed exclusively from beginning to end. It is likely that in the course of following a need, other needs will be provoked as a result of exposure to information encountered along the way (Fig 1.2).

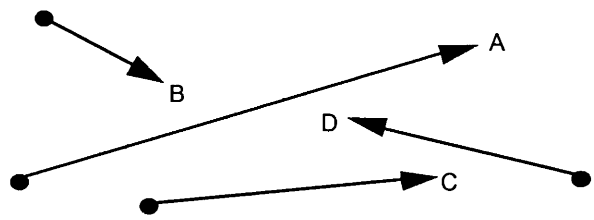


Fig 1.2 Multiple information-needs. (A-D indicate end-points)

In some circumstances these needs may actually be perceived as sub-needs – i.e. information-needs whose satisfaction will contribute to the ultimate satisfaction of one of the other information-needs. For my purposes, it is not necessary to recognise explicitly such subordination – the *effects* of it are implicit in the following three characteristics.

Just as absolute direction is not presented in this characterisation, neither is there any pretence to describing 'lengths' or 'sizes' of information-needs. The presented lengths in the diagrams show only that a difference might exist.

Information-needs are tangential

The themes of the additional needs provoked during a searching session are unlikely to be random. Although it is possible that some of the spontaneously provoked needs will not be directly related to the original one, they are most likely to be closely related to that of the current need at the time of their inception. In fact they are most likely to be extremely subtle variations of the information-need current at the time. This splitting apart is described as 'tangential' to highlight two things: firstly, that the two information-needs share a common history; and secondly, that they 'peel' apart, gently at first, with accelerating divergence (Fig 1.3). This gentle peeling apart applies only at the time of the split – very soon afterwards, the two needs could be quite distinct. It is intended to highlight that the user may not necessarily perceive the splitting at the point at which it occurs – this may happen only later when the difference has become large enough to become apparent to him.

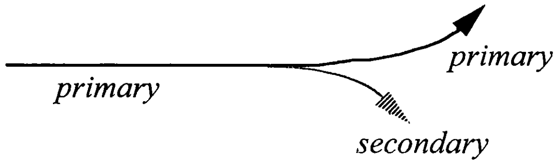


Fig 1.3. One information-need splitting tangentially into two information-needs.

Incorporating tangential splitting into the visual representation gives a tree-like structure of multiple diverging information-needs all effectively rooted at the same starting point (Fig 1.4).

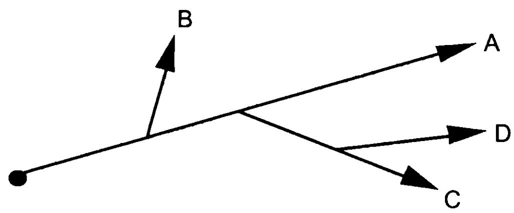


Fig 1.4 Multiple information-needs sharing a common root splitting tangentially from one another.

Information-needs are embedded

If one accepts that an information-need may fragment into multiple sub-needs, or may provoke a secondary related need at any point during its existence, then the same can happen to those new information-needs. This indicates that information-needs are naturally embedded within one another. The degree to which they are embedded might be related to the individual agent's ability to deal with the information necessary to manage the stacking of, and returning to, the individual needs embedded within each other.

Embedding is shown in the visual representation by means of fragmented lines, with each segment numbered to indicate the order in which it was followed by the agent. For example, Fig 1.5 shows that the agent followed *A* for a short distance before (for example, being distracted and) following *B* to completion before returning to follow *A* again. Segments *2* to *4* show the agent stacking *A* in order to follow *C*, and then stacking *C* in order to follow *D*, before returning to complete *C*, and then finally returning to complete the original information-need *A*.

At this point it is perhaps worth reiterating the 'common history' aspect of the tangential relationship between the information-needs. In Fig 1.5, the section marked *1* is the complete development of *B*, but approximately the first half of that is also the early development of *A*, *C*, and *D*.

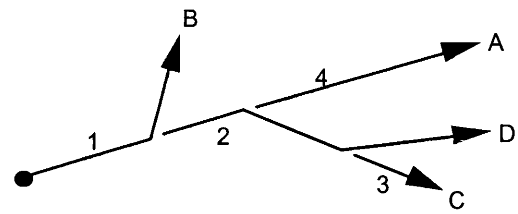


Fig 1.5 Embedded information-needs. (1-4 indicate order of execution)

Information-needs are threaded

We develop (and strive to follow) disciplines in most areas of our life in an effort to gain a certain amount of control and to encourage consistency and reliability in our actions. Nevertheless, spontaneous behaviour is both natural and automatic and is, by its very nature, neither methodical nor disciplined. Expecting a searching agent to manage meticulously the numerous and various embedded information-needs that crop-up during a search session is at best unreasonable, and at worst unworkable.

What is much more likely is that the agent will follow one need partially, then switch to another, follow that partially, switch back to the original, follow that partially, before switching once again, etc.

These switches of context and sub-task can be provoked by many factors. One such reason might be that of information encountered whilst following up one need provoking a change in a 'stacked' previous need of such a magnitude as to warrant an immediate return to it. Additionally, the reappraisal of a sub- or secondary-need, previously regarded as having been followed to satisfaction, may be provoked by newly encountered information. This reappraisal in the light of new information may result in the older, satisfied, need being perceived differently and hence regarded, once again, as unsatisfied. That would prompt a return to it in order to develop it further.

The term 'threading' is used to indicate that the execution of each of the information-needs is threaded through the execution of the others. It can be thought of as a kind of course-grained parallel execution of all the needs from their common route to their individual end points.

Fig 1.6 indicates that the agent followed *A* before moving off to follow *B* for a short distance, then returned to following *A*, moved off to follow *C* partly, before jumping back to complete *B*.

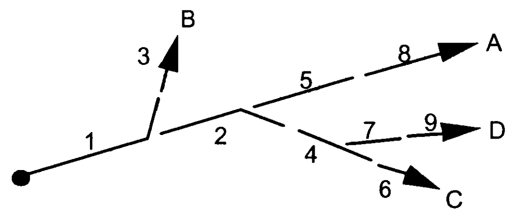


Fig 1.6 Threaded information-needs.

The causes of such switches in attention for each case would be difficult or impossible to determine. Therefore, this characterisation does not assume their availability – it merely accepts that it *can* happen.

1.3 Summary

This section has presented the idea that describing information-needs is an ill founded and ineffective endeavour; that information-needs change constantly and that this change is most effected by related information encountered during information seeking activities; that the behaviour of an agent during searching activities will be greatly influenced by the current state of the information-need.

Three characteristics of the structure of information-needs during a searching session were introduced: the constant change of an information-need is a development towards an end point of either satisfaction or redundancy; the original information-need may split into several sub- or secondary-needs, each of which is a different development of the original need leading up to the split. Care was taken to not say anything about how the formation of the structural characteristics happens, simply that it does.

Finally, two related characteristics of the execution of the information-needs by the searching agent were introduced: the development of an individual information-need may be embedded within the development of another; the embedding might not be a clean completion of individual information-needs, but may result in a threading of the development of the information-needs through each other. These are essentially a recognition of the operational characteristics of information-needs and that these characteristics are managed by the agent.

2 Current information retrieval systems

Within interactive approaches to IR systems, two main classes can be seen. Both classes represent a fundamentally different approach to the exploration of an information space by the user – that of query-based, and that of browse-based. Systems employing the query-based approach rely upon some kind of description being provided and refined by the user. Systems employing the browse-based approach rely upon the user selecting or pointing to information objects presented to them by the system. This distinction is key to understanding the motivation and ideas of this thesis.

Only interactive IR systems are considered. Batch systems are so hopelessly far removed in their mode of operation from the realities of the searching task, that analysis, of the sort envisaged here, and the resulting suggestions for improvement become so subtle as to render them practically meaningless.

Contents of this chapter

Section 2.1 presents the seven 'terms of reference' that will be used for the comparative analysis. The terms come directly from the ideas presented in Chapter 1.

Using those terms of reference, systems that typically result from both approaches are reviewed: Section 2.2 reviews the support offered by the query-based approach; and then Section 2.3 reviews that of the browse-based approach.

Finally, Section 2.4 summarises across both approaches, highlighting what will become important later in the thesis – i.e. the complementary nature of the support that they provide.

2.1 Terms of reference

Listed here are the seven terms of reference that will be used to frame the discussions in the following sections.

The first term comes directly from the ideas of Section 1.1:

1. **Avoidance of information-need description**

From Section 1.2, there are terms based on the three structural characteristics:

2. **Support for Development**

3. **Support for Tangentiality**

4. **Support for Multiplicity**

Also from Section 1.2, there are terms based on the two operational characteristics:

5. **Support for Embedding**

6. **Support for Threading**

The wider context within which this work is being carried out is that of multimedia IR, and the goal is the development of a new framework for IR that will facilitate effective retrieval of information expressed in the new media of sound, image, video, etc. Its significance will become apparent later. For the moment it is simply added as a term of reference:

7. **Support or suitability for non-text media**

2.2 The query-based approach

A query-based system is regarded as any one that requires of the user the production and refinement of a description of their information-need. Broadly speaking, this description encompasses the vast majority of current IR systems – both commercial and research.

The most prominent examples of query-based systems are the Boolean systems offered by commercial dial-up online information providers – e.g. *STAIRS* [Ibm72], and its derivative *BRS-Search*. Other Boolean systems available for most personal workstations and bundled with information bases on CD-ROM are: *Topic* [Lehman94], *Personal Librarian*, and the rather dismal searching interfaces provided with almost every non-IR computer system. The most modern of the query-based systems are the weighted systems that were generally restricted to the research world: *ConQuest* [Nelson94], *NRT* [Sanderson91], *WAIS* [Stein91], *Inquiry* [Callan92], *Smart* [Salton71], and *Okapi* [Robertson93]. With the development of the public Internet, hybrid weighted and Boolean retrieval systems form the core of most of the Web searching services (e.g. AltaVista.com, Lycos.com, HotBot.com, etc)

What all these systems have in common is that their operation centres on a query – i.e. a description of the user's information-need, presented in terms of some system-supported language. These languages are direct derivatives of the internal representations used by the system for the stored information objects. For example, the presence or absence of individual words in the representation of documents, along with Boolean operators such as 'AND' and 'OR', make up their query languages.

In practice, the user generates an initial query, the query is evaluated by the system, a list of resulting documents is presented to the user, the user inspects these documents, refines his query, the new query is evaluated by the system, and the process iterates Fig 2.1.

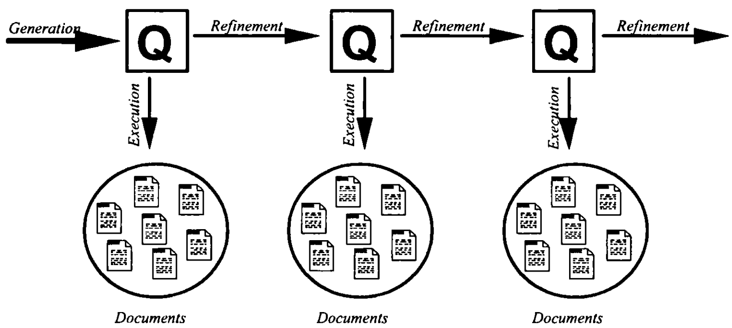


Fig 2.1 The iterative process of query-based systems.

Term 1 – "Avoidance of description"

Users must describe the documents that they believe will satisfy their information-needs by providing the pattern of presence or absence of words within them. This must be done without having seen the documents in advance. As pointed out already, expecting a user to be able to describe something he doesn't have, does not appear to be a very promising approach. To then place the restriction that such descriptions must be made in terms of an artificial language is a further hindrance to the user satisfying his needs. These languages are chosen mainly for their computational simplicity and correspondence to the internal representation schemes and not for their suitability for expression of information-needs.

Weighted retrieval systems have slightly different internal representations and significantly different retrieval algorithms to that of the Boolean systems. This results in querying languages with a very similar basis (i.e. word occurrences), but with very different interpretations. Here, the 'hard' nature of Boolean operators on word occurrences is replaced by a 'soft' accumulation of weights attached to candidate words, giving more informal and less error-sensitive languages in which to express information-needs. Nevertheless, the direct correspondence between the internal representations and the query languages remain.

These representations are also used by the systems to 'explain' particular retrieval decisions. The words that caused the retrieval of particular documents are often highlighted in some way, thus providing feedback to the user on the extent to which each of the query words is involved in the retrieval of documents. This explanation can be crucial to the effective exploitation of such systems. It is with this information that users make their decisions upon how to refine their query and thus continue their search.

Term 2 – "Development"

During the query-refinement/retrieve/document-inspection iterations, the user's information-need will be changing as a result of his exposure to documents. One could imagine that the user would be able to reflect that changing information-need through a corresponding modification of the query. Unfortunately, two things prevent this from happening.

The first factor is that as an information-need changes, the changes could be sufficiently subtle that the user is not consciously aware of each change despite that it is already affecting his relevance judgments. Changes in our beliefs or attitudes are often not clear to us until we notice that they are having significant effects upon our actions. A user will only become aware of a change in his information-need after it has reached a level great enough to provoke a reappraisal of that need. Further, time will have passed between the changes taking effect and the user becoming aware and then acting upon them. The making of relevance judgments and the user's general satisfaction with the progress of the search will naturally be out of synchronisation with the way that the search is being driven.

The second factor is that all information-needs perceived by the user must be described. This requires that he translate each new version of the information-need into the terms of the query language. This task is not simple and the feedback provided by the results of execution of a particular query is the only way in which a user can appraise the appropriateness of a particular query formulation. This means that several iterations of query-refinement and query-execution are likely to be required before the user will be happy with, or confident in, the query as a representation of his information-need. Sadly, in the meantime, the very information-need that he is endeavouring to represent will have changed 'beneath his feet' due to his continuing exposure to documents.

Query-based systems do not *support* developing information-needs. These systems *allow* those changes to happen and the user to track that development. The requirement for description serves only to hinder that tracking. What they do support the development of, are *queries* – but as indicated above, queries are a poor surrogate for the development of information-needs themselves.

Terms 3 & 4 – "Tangentiality and Multiplicity"

The presence of multiple information-needs during a search session implies that the user might want to switch between those needs at any point, any number of times. Query-based systems, with their emphasis on the progressive development of a query, naturally hinder this process. As outlined above, proper development of a query can only happen with the query-refinement/retrieve/document-appraisal cycle. This means that either the process of effective query development is disturbed, or that a new information-need cannot be followed up as and when necessary.

An additional problem is that the user might not even notice that has had a new, or subordinate, information-need provoked until he has already followed it for some time. This is identical to that outlined above in the discussion of the support for development, where changes in an information-need are not consciously perceived until they have become large. Query-based systems with their iterative query-refinement do not support directly a convenient 'unwinding' of the previous actions to allow the user to return to the point of splitting for reappraisal.

Histories (or saved states) of all queries and their resulting document-lists can go some way to supporting this. Nevertheless, histories would typically store only those queries submitted for execution – they would not store each of the various edited versions that were present between query-executions. If that level of detail were stored, it would result in inordinately long and complex histories – histories that would only show the results of the attempts to represent instantaneous information-needs. That would be rather indirect. One can only speculate that the lack of history-management systems to help with such problems results from the additional complexity that such a system would pile on top of the existing complexity.

Terms 5 & 6: "Embedding and Threading"

The fact that multiple information-needs are embedded and threaded is the result of the manner in which the user spontaneously perceives of, and manages the pursuance of, those needs. With the lack of support for multiple information-needs outlined above, the more subtle support that embedding and threading require is ruled out.

Term 7: "non-text media"

The query-based approach has been the major thrust of IR research, development, and application for many years – it looks set to remain so. The concept of a query has become central to our ideas of IR systems. Despite the success of these systems in the past, their effectiveness is threatened by non-text media.

The hardware and software tools to support the generation and storage of multimedia information have become widespread. Large collections of such information are gathering at an increasing pace, and the desire to retrieve that information is growing in parallel.

Information stored in a non-textual medium, by its very nature, requires different representation techniques to that of text. Examples such as that of speech-only audio information being transcribed into text, and then that text being used for retrieval of the original audio are, by virtue of their triviality, not very instructive and are not representative of the general situation. Music, photographs, video, etc cannot be given anything but the most trivial of textual 'transcriptions', and because of the human effort required, these transcriptions are expensive to generate. By necessity, new representation techniques for non-textual media are being developed. These representations, if they are to be effective, are unlikely to be accessible to humans. One approach demonstrating that point is that of neural networks, where any representation would consist of minute probability modifications spread across many thousands of cells each of which has no 'real' meaning to us.

As a result of such demand and such representational developments, the reliance of query-based systems upon the user's exposure to, and manipulation of, internal representations, rules out the descriptive approach as the basis of the next generation of multimedia IR systems.

Relevance Feedback

Weighted-term systems can supplement queries as a means of obtaining information-need specifications, by incorporating Relevance Feedback. Relevance Feedback is the technique where the user indicates to the system documents that he regards as relevant and the system then uses that information, in place of a query, to perform a search. The idea is that the contents of such relevant-marked documents will essentially be a richer source of information for choosing terms and setting weights. The presence of Relevance Feedback (still relatively uncommon in the systems used by the majority of people – e.g. the Web search engines) can reduce significantly the problems identified here with pure query-based systems. Nevertheless, Relevance Feedback changes the mode of operation and interaction of a query-based system with the user, presents its own problems, and does not invalidate the statements made here – it will be returned to in Chapter 3.

2.3 The browse-based approach

In a browse-based system, there is no query and no explicit description of the user's information-need. The user furthers his search purely through his selection of information objects (usually documents) presented to him by the system. Such systems almost exclusively employ a graphical interface presenting documents as objects with links between them (Fig 2.2).

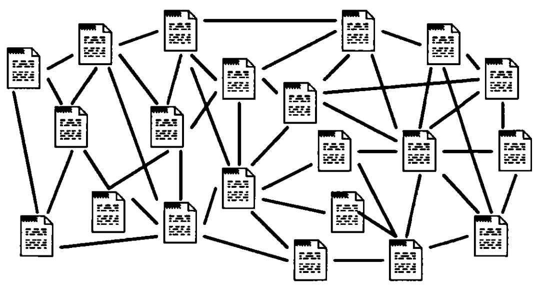


Fig 2.2 A network of information objects, with accessibility relationships between them

The classic browse-based IR system is I^3R [Thompson89] – where documents and groups of documents are presented as icons with accessibility links between them. *Transient-Hypergraphs* [Shepherd91] exemplify an early application of the browse-based approach to a traditional strongly-typed database. *TACHIR* [Agosti94] is a system that allows the user to browse across a plane of documents and their accessibility relationships, across a similar plane of terms, and across the accessibility relationships between the different planes. *Galaxy of News* [Rennison94], and *Bead* [Chalmers92], present documents in roughly 2-D spaces that have been dimensionally reduced from original higher dimensional spaces. The distance-metric is again that of an accessibility relationship or similarity measure.

Not hypertext systems

The browse-based systems considered here are distinguished from their cousin 'hypertext' systems. Although both present objects to the user with links between them, the distinction is based upon the origin and nature of those links. In hypertext systems, the accessibility links are either human generated, or derived directly from the structure/syntax of the objects concerned (e.g. for a book in hypertext from there will be links such as 'next paragraph', 'previous paragraph', and links to other pages directly referenced in the contents). The links commonly span a tiny subset of the object space (i.e. the degree of mutual interconnection is very low), and typically, the links are of a binary nature – i.e. present or absent. Finally, the links commonly represent a 'recommended' or 'preferred' reading path through the collection of objects.

In browse-based IR systems, the links are computed, and that computation is based upon the documents' semantic content (i.e. not just their syntax). The links are commonly present between every object and every other object in the space. The strength of the links (i.e. the degree of accessibility) is represented by continuous values. Finally, the links do not reflect direct references from object to object; they are derived from indirect associations formed by the external similarity measure.

Despite having made clear that hypertexts are not the concern of this analysis, some of the following criticisms (both positive and negative remarks) do also apply to hypertexts.

Strongest links

In order to move around within the information space, the user makes selections from a list of alternatives presented to him by the system. These alternatives are objects that the system has computed to be the most appropriate next steps. These next-steps are computed relative to the current location (i.e. the current information object), and the degree of appropriateness is the accessibility relationship defined for the space.

This accessibility relationship may be anything from a simple mutual set membership, to a sophisticated measure of semantic similarity between two objects. Usually it is a pseudo-semantic similarity measure based upon one of the classical retrieval models. Although it is possible for every other information object to have some degree of accessibility from a given object, it is not practical, in all but the most trivially small collections of objects, to present them all to the user. Only a small number of the strongest links is presented at each object.

Static non-adaptive views

The information spaces (i.e. the arrangement of information objects and their accessibility relationships) are static. They are set up once for a particular collection of information objects, and remain the same for all searching sessions, and for all users. The selections made by the user only allow them to move from one statically linked object to the next. The static nature of the information space is in contrast to that of the adaptive and dynamic space of the query-based systems where the space changes in response to individual users and their instantaneous needs – albeit via the clumsy mechanism of queries.

Term 1 – "Avoidance of description"

During a session, the user simply moves from one object to the next as the feeling takes him. He need only identify something of interest in the locality of his current object and investigate. There is no need for the user to attempt to translate that interest into a description of why he now wishes to move in that direction. More importantly, there is no need for the user to even think in which direction he may possibly wish to move – he simply sees where he wishes to go, and goes there. The manner by which information-needs are followed is both spontaneous and immediate. The lack of description also means that the user is completely hidden from any internal representations that were used to form the accessibility relationship.

Term 2 – "Development"

As the information space is static, the paths through it can only be over the predetermined links between information objects. The small number of statically defined links available at each node restricts the possible next steps to those that seem appropriate from the point of view of the current node alone. The 'view' from that object is the same for all information-needs (i.e. for all users, all tasks, at all times, in all sessions), regardless of their state or rate of development.

To assist in the appreciation of this point, let us imagine two users, identical in every respect and therefore with an identical initial information-need. If they were to approach a particular object by different routes, they would have been exposed to different information (or minimally the same information, but in a different order). This would imply that they and their information-needs, at the time they reached the object in question, would be different. In that case, their 'view' of the world would be different and the effect on their information-needs of their exposure to the object would be different. Nevertheless, a browse-based system would show the same next-steps to both users. Therefore, browsing systems do not support developing information-needs.

Terms 3 & 4 – "Tangentiality and Multiplicity"

The almost exclusively graphical user-interfaces of browsing systems provide the benefits of visibility, immediacy, and ease of management of the objects being presented. The graphical presentation means that no matter what route is taken through the object space, that route is visible and can be backtracked or retraced at will. Additionally, it is the actual objects that are presented in these 'histories' – not user-descriptions of information-needs, as is the case in query-based systems. As it was those objects that provoked the development of the information-need in the first place, it seems reasonable to expect them to be more effective at reorienting the user if they were returned to at a later stage.

Depending upon the particular system, various amounts of the information space will be visible at any one time. The more of the space that is visible, the more, in general, the user will be able to simply jump to a previous location without actually having to backtrack through every intermediate step.

This visibility of information objects and the orientation provided by the links between them provide an environment that supports the user in the management of multiple information-needs. The directness that should be expected from the user's return to an object (rather than a query) suggests that determining where a small deviation in the information-need led to a new one being followed should be easier. Therefore, browse-based systems can be said to support the tangentiality of multiple information-needs – certainly to a greater extent than query-based systems.

Terms 5 & 6: "Embedding and Threading"

As outlined above, switching between paths through the information space is supported by the graphical presentation. This allows users to follow an information-need, and then return to a previous one by simply backtracking until they identify the information object at which they feel they began to diverge from the primary and follow a secondary information-need. They can then return to following the original primary need. Embedded needs are therefore supported.

Taking this one step further, it is easy to see that it is not necessary for the user to follow one particular information-need to completion before backtracking to continue a previous need. They can switch at will between several concurrent information-needs, thus threading the development of them.

Although there is support for multiple, embedded, and threaded information-needs, it must be remembered that the user's information-needs, however they are developed, arranged, or managed, must fit into the static restricted view of the information spaces presented by these systems.

Term 7: "non-text media"

In browse-based systems, the user is not exposed to the internal representations used to store information objects or compute accessibility relationships between them. The user simply views the objects in their native format, be it text, sound, graphics, etc. This is attractive because it permits the most inaccessible, most complex, and most contrived of mechanisms or combinations of mechanisms to be employed to structure the information space. This is particularly attractive when considering the development of a new generation of multimedia IR systems.

2.4 The approaches are complementary

From the critique of the two main approaches to IR it can be seen that their support, with respect to the terms of reference, is rather patchy. Nevertheless, there is a pattern in that support. The strength of the query-based approach is the highly adaptive nature of the views of the information space that enable them to follow the changing information-needs. Their main flaw is the difficulty in driving this adaptability effectively using descriptive queries. The strength of the browse-based approach is the highly direct and immediate manner in which a user can act upon observed information and changes in his information-need. Its drawback is the non-adaptive unsympathetic static views that restrict the extent to which the directness and immediacy of the graphical presentation can be exploited.

Graphical presentation of the information space

The graphical environment of information objects and links is the tool provided by browse-based systems that supports the multiple, embedded, and threaded nature of information-needs. The network of objects and links provides an overview of the current search path. This overview provides orientation-, coordination-, and memory-aides that allow the user to follow his information-needs in an unpredictable and undisciplined manner. For example, if the user decides that his current direction has become unsatisfactory, he can either iteratively backtrack until he is on firmer ground and head off in another direction, alternatively, he can immediately jump back to a point far back where he is certain he was doing better, and continue more carefully.

Adaptive views of the information space

Query-based systems restructure the information space on each query execution. This is a method by which the view presented to a user situated at any point in the space can be made to correspond more closely to, or at least change in a manner sympathetic to, the user's current information-need. This means that as the user's information-need changes, be it subtly or greatly, the system can restructure the space in a manner that brings closer, and thus makes more accessible, those information objects it considers to be most likely to be relevant to the user's information-need at that time. Adaptive views, therefore, support the developing nature of information-needs. Nevertheless, as stated already, the success of such an approach would be contingent upon the system obtaining an accurate representation of the information-need – i.e. one superior to that obtained by the query-based systems.

In the browse-based approach, there is a clear sense of a user always being situated at one or other information object in the information space. In the query-based approach the user is never at an actual object in the space. He is at a point, the position of which, is defined by the query (which can be regarded as a virtual document or object). That point may happen to correspond to the position of an information object, but the user is never considered to actually be at the object.

No support for developing information-needs

The one area, for which both approaches failed to provide support (as opposed to just allowing it), is that of the developing nature of information-needs. Development is one of the most fundamental characteristics of information-needs during a searching task – it underpins the ideas and arguments of this thesis.

Exposure to internal representations

One of the arguments of this thesis is that the key to supporting complex media objects is to remove all traces of internal representation methods from all interactions with the user. The upshot of taking that position is to effectively rule out all forms of descriptive querying. It is difficult to imagine a language that would allow a direct description of something to be formulated without resort to some representation technique. Even when we attempt to describe abstract concepts or visions to other people we must employ a basic representational vocabulary. Such a vocabulary must often be negotiated by the communicating participating parties in advance of the description being effected.

The query-based approach is fundamentally flawed in its exposure of representations, but the browse-based approach is able to allow users to explore the document collection without any hint of there even being an internal representation.

2.5 Summary

This chapter presented seven terms of reference for the comparative analysis of approaches to interactive IR. The seven terms were the five characteristics of information-needs presented in Chapter 1, with the addition of "Avoidance of descriptive querying", and "Support for non-text media".

A comparison of query-based and browse-based approaches using the terms of reference highlighted a number of things – amongst them were:

In order to drive the desired view-adaptation by space restructuring, one must obtain some representation of the user's current information-need. However, in order to support the new media we cannot expect direct descriptions of the information-need from the user. A new way of obtaining the required information in an effective manner is required.

In order to support multiple, embedded, and threaded information-needs there is the graphical presentation of an information space. However, this is at odds with the desire to provide an adaptive, continually restructuring, space. A method of providing a graphical presentation of a constantly changing information space is required.

Essentially, the comparison of query-based and browse-based approaches has identified their characteristics as complementary with respect to the seven terms of reference. The following chapter proposes a solution that exploits that complementarity and addresses the other identified problems.

3 Proposing a path-based hybrid approach

Taking the ideas of Chapter 1 with the analysis presented in Chapter 2, and combining them with informal observations of searching behaviour, this chapter presents the idea of using 'paths' as a mechanism to drive a dynamic searching environment. Paths offer the possibility of an IR system following the development of an information-need without resorting to user-generated descriptions. Moreover, they allow the environment to be graphical without the exposure of internal representations of objects. Paths have the novel characteristic of providing a basis for dynamic contextual interpretation of relevant-indicated objects.

Much of this chapter is relatively informal and speculative, echoing the way in which the ideas of exploiting paths were conceived. It forms a bridge between the ideas of Part I and the ideas of Part II.

Contents of this chapter

Section 3.1 details observations made whilst searching. Section 3.2 describes the idea of paths being the key to query-free support of developing information needs. Section 3.3 discusses the contextual interpretation that results from using paths to direct an IR system's searching. Finally, Section 3.4 proposes a look-and-feel of a system that would support the path-based contextual browsing envisaged in this chapter.

3.1 Some observations using Relevance Feedback

The following observations were made whilst building and experimenting with an early version of the IR engine to be presented in Chapter 6. Connected to it was an experimental graphical interface (somewhat like that of NRT [Sanderson91]). It presented a query as a list of terms, each term having a slider representation of its associated weight. The user could add and remove terms, and could adjust the weights of individual terms before carrying out a search using that query. In addition, the interface had an area into which the user could move retrieved documents. The idea was that a user would move documents he regarded as relevant into that area, and the system would then perform a standard binary-probabilistic Relevance Feedback calculation using them (following the techniques of Robertson & Spärck-Jones [Robertson76], and of Croft & Harper [Croft79]). The resulting query would be displayed to the user, and he could then choose to add or delete terms in that query, or (more commonly) simply perform a search using it.

The following are my personal observations – they are not intended to be representative of anything other than my experience. They are presented here to ground, and to aide the explanation of, the thoughts behind this thesis.

Query-modification versus relevance feedback

Starting a session with a simple query would have variable, but generally poor, results. There would rarely be properly relevant documents in the result set, but usually there would be one or two documents of very weak relevance. The alternative of crafting a more complex query (which usually meant more terms or synonyms for existing terms) seemed unattractive because of the effort involved, and did not appear to produce results that were much, if any, better. Experience seemed to suggest that it was more effective (and definitely less effort) to use the simple initial query, then identify any marginally relevant documents, and mark them as relevant.

Taking a query-modification route would often take several searches before any 'good' documents were retrieved. The relevance-feedback route would, on each successive search, produce documents that were increasingly relevant. This could be explained by the marginally relevant documents being richer sources of information for the system – e.g. more terms, and, if there were more than one document marked relevant, then there would be some relevance-frequency information for those terms.

With query-modification I felt that I was groping around until I suddenly hit on an effective query, whereas, with each iteration of relevance-feedback there was a sense of at least moving *towards* the documents for which I was looking.

Improving a query requires a degree of knowledge of the subject matter in question in order that effective words can be either thought of, or identified in retrieved documents. For most weighted-term retrieval systems, 'effective' means words that are infrequent in the collection as a whole, but frequent in the targeted documents. So, not only must a user identify words that are associated with the topic for which he is searching, but he must also have some idea of their relative frequencies of occurrence in the particular collection of documents being searched.

By contrast, using relevance feedback to drive the search process requires only that the user recognise documents that are, to some extent, relevant. Each additional relevant-indicated document gave the system a richer representation of the information-need.

Changing relevance

I noticed that, as the results of successive Relevance Feedback produced documents of higher relevance, the lesser relevance of the earlier finds became apparent. This seems reasonable, and to be expected as inherent in any search process. Removing them from the 'relevant set' and performing a search with the smaller relevant set often improved the results further.

The suggestion is that this iterative replacement of early relevant-marked documents with recent, more relevant, finds, although reducing the number of relevant documents, was also allowing the search to develop. This is consistent with the way in which Relevance Feedback works – i.e. it uses the relevant set as a statistical sample of what relevant documents look like (in terms of the underlying representations). If the relevant set is made up of only documents regarded as highly relevant by the user, then that should be a better sample than a larger one with less relevant documents in it. An effective set of relevant documents is clearly not only a matter of quantity, it is also one of quality.

A further point to note is that simple accretion of new relevant documents would follow the 'law of diminishing returns' – i.e. each successive document would have less and less effect. This is because as the set of relevant-marked documents grows, the percentage change effected by a single addition becomes less. This is trivially observable with any traditional Relevance Feedback system.

Changing information-need

There was another factor motivating the changes to the relevant set – changes to my information-need. I felt that the changes to my information-need were led by two things: an increasing awareness of the subject in question, and an increasing understanding of what documents were to be found in that particular collection.

The increase in the perceived awareness/understanding of the subject area around my information-need was a direct result of my exposure to documents in the collection. Even documents that were only marginally relevant changed my understandings enough that previously relevant-marked documents would appear less relevant. More striking was that documents that I regarded as highly relevant would change my understanding of the area to such an extent that they themselves would soon after appear as less relevant.

Functional tasks versus procedural tasks

Making relevance judgements on information is the most basic and immutable part of the process of information discovery – regardless of what tools or methods are used to perform it. Any operational requirements beyond that is interference, and serves only to decrease the effectiveness of the process or its accessibility to those not practised in its use.

With relevance feedback, the user is reasoning and driving the process from a perspective that is more direct and functional than that of query-manipulation. That contrasts with driving things from the indirect and procedural perspective of terms, combinations of terms, and their frequencies of occurrences.

Returning, as promised, to the analysis of query-based systems in Section 2.2, the above discussions show that even when queries are transcended by Relevance Feedback, there are still significant management issues around the user providing an effective representation of the information-need with which the system can perform searches. Such problems are magnified by the multiple, developing, tangential, embedded, and threaded nature of information-needs just as they are in a query-based system – although not to the same extent.

Summary

Gathering the thoughts and suggestions of this section together:

- **Emphasise the functional** – i.e. a system should concentrate its support on the key task of the user identifying relevant documents.
- **De-emphasise the procedural** – i.e. a system should minimise or remove a need for the user to manage queries, terms, etc.
- **Managing relevant sets is procedural** – i.e. just as managing query terms and their weights is procedural load, so too is the effort of maintaining a representative set of relevant documents whilst the information-need changes.
- **Documents change information-needs** – i.e. during a session, a user may change his ideas on what is and is not relevant as a result of his exposure to information.
- **Relevance indications are subject to change** – i.e. relevance indications made by a user might be withdrawn, particularly those made early on in a session.

The following section takes these ideas and combines them into the basis of a new approach.

3.2 Supporting developing information-needs

An information-need is a changing need. The change is a development towards an endpoint of satisfaction or redundancy. This development results from the searching agent being exposed to information during the search. This information adds to and refines the knowledge and view of the world that is held by the agent. It is that knowledge that drives the assessment of information with respect to the information-need. Therefore, as the information upon which such relevance assessments are made improves, it seems reasonable to assume that the assessments themselves improve. That is, relevance assessments and relevance-related decisions improve during the progress of a search.

The passage of experience

The change in the knowledge and view of the world of the agent, resulting from the exposure to information, can be described as being "in the light of experience"[1] [Jeffrey65].

If relevance assessments improve in the light of experience, then this implies that at any point in a search, any previously made assessments are less good – or more precisely, have retrospectively *become* less good. The corollary of assessments improving continually as the search progresses is that they would be seen to continually degrade if one were to look back in time from any point.

It is now clear why browse-based systems do not support developing information-needs. Decisions made at the start of any search path restrict the portion of the information space that is directly accessible from then on. Therefore, at any later point in the search, the available space is still being restricted by those early and now less-appropriate assessments.

[1] Jeffrey developed a probabilistic mechanism and philosophical framework for the revision of probabilities in a consistent manner in the light of new evidence. In doing so, it imposes a mathematical restriction that probabilities cannot be allowed to reach either zero or one if they are to be revisable. The restriction is also motivated philosophically – attaching a temporal meaning to probabilities.

Undoing the effects of early decisions

One obvious solution to the problems of early and now less-appropriate assessments would be to simply start all over again. Starting again seems a rather drastic action; it would seem more useful if the constraining effects of the earlier decisions could somehow be removed from the current 'view' of the information space. One could consider an approach such as a 'sell-by-date' that would simply remove the effect of all decisions made over a certain amount of time in the past. This age-related switch-off can be presented as a binary weighting function where the 'weight' of a decision switches from 1 to 0 after its ages reaches a certain value (Fig 3.1).

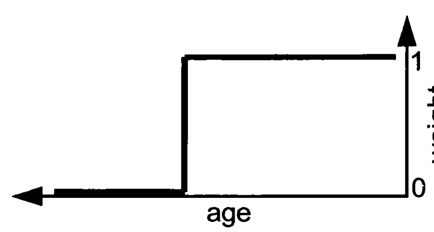


Fig. 3.1 A binary weighting function, switching-off the effect of decisions that reach a certain age.

Completely switching-off old decisions/assessments after some time interval might be in itself a rather drastic approach. It might not accurately reflect their change in appropriateness with respect to the information-need. All decisions and assessments might have some degree of relevance to the information-need, however small. Rather than removing them completely, a gradual reduction of their effect over time might correspond more closely to their changing appropriateness (Fig 3.2).

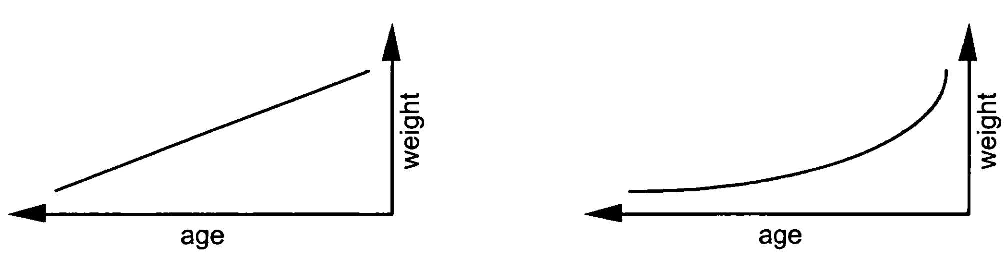


Fig 3.2 Two examples of functions to reduce the weight of relevance decisions with their age.

If the effect of decisions is to be reduced gradually over time, a manner of determining the 'age' of particular decisions with respect to each other and with respect to the particular information-needs is required. The idea of 'time' would not necessarily be measured in minutes and seconds, it could be an "information-need time", where time progresses with respect to changes in the information-need.

Following the ideas of the previous chapters, such changes could be the result of the exposure of the searching agent to information objects. Therefore, a reasonable 'tick' in this time might be an individual presentation of an information object to the searching agent. This suggests that the ordering of objects in path leading to the current point be used for ageing. The path would be from the point representing the start of the information-need, and would pass through all information objects to which the agent was exposed, and that contributed to the state of the current information-need.

The result of such an approach would be to essentially treat the path as an instantiation (or concrete manifestation) of the abstract information-need.

An evidential approach

So far the discussions have been relatively informal, talking of things such as "undoing the effects of earlier decisions". Formalisation of those arguments begins now and will result in a model presented in Chapter 4.

The first step is to distinguish the motivation for an action from the action itself. Therefore, *evidence* collected on a path, will be distinguished from its *combination*, and from any *use* to which it is put. There are many sources of evidence – for example, a user-indication of relevance can be regarded as evidence that the document in question is indicative of the current information-need. Then, that evidence can be combined in many ways with other evidence and models of the environment where that evidence was observed. Finally, the results of the combination can be used, again in a number of ways, to effect a change to the document-space around the user in many ways.

Talking of 'evidence' accepts that there might be many, varied, and interrelated sources of influence. For example, as suggested above, the time at which a relevance-indication was made can be considered as evidence that can be used to help interpret the evidence within the relevance-indication itself. Additionally, individual user-exposures to documents can be regarded as evidence that 'time' has passed (i.e. in the sense suggested above).

As the discussions of interactions with retrieval systems have been moving rapidly from the purely abstract to the concrete, the term 'searching agent' will now be dropped. Its purpose was to put distance between the abstract idea and our manifold and potentially interfering experiences of real people searching – this is no longer necessary. The altogether simpler term 'user' will be used exclusively.

Evidence combination

To form an effective representation of the information-need, the system could gather all evidence available along the path. Such evidence might include the documents to which the user was exposed, documents regarded as relevant by the user, and decisions or selections made by the user. The most visible evidence would be documents indicated (whether directly or indirectly) by the user to be relevant.

The evidence from the multiple sources along the path could be combined in a manner that recognises the lessening of their expected appropriateness (as described previously, and shown in Fig 3.2). This would amount to a 'weighting of evidence' where older evidence receives a lower weight than recent evidence.

A representation of the user's information-need that is collected by the system could be used to restructure the information space around the point at which the user is currently located. That is, the view presented to the user at any point will be calculated using the evidence gathered from the path followed by the user. This dynamic view is similar to the query-based approach, except for the source of the evidence – i.e. observation of the user, rather than a description generated by the user. The dynamic path-influenced view is different from that of the browse-based approach in that any particular object will have a different view onwards depending upon the route taken to reach it. The next section develops that idea.

3.3 Contextual interpretation

The view presented at any point in the information space will consist of other information objects. These other objects would ideally be those that the user would most likely want to investigate after exposure to the current object. The browse-based approach presents objects that are most similar to the current one. The 'similarity' is with respect to some calculation based upon measurable characteristics of the objects. This calculation is a pair-wise comparison of the current object with all other objects in the collection.

The objects in a collection are being 'interpreted' by the similarity measure when they are considered for display. That interpretation is being performed within the context of the current object. The only other information influencing this interpretation (and therefore forming part of the context) would be summary information on the distribution of individual characteristics across the collection as a whole. This localised and static context, within which the system makes interpretations, does not reflect the nature of the context within which the user will make his interpretations.

It is difficult to imagine any information, however 'simple', that could not be interpreted differently in different contexts. Information objects will have many (potentially infinite) interpretations and so restricting the view presented at any particular object to one static selection of next steps does not seem appropriate.

In the approach proposed here, the view presented at any object will not simply depend upon the object itself, but will depend also upon the path taken by the user to reach that point. That is, the interpretation of objects will take place within a context that includes not only the current object, but also those on the path. The idea is that if the user is following up a particular theme, that theme will be apparent in the path in the form of such things as the selected information objects and their order of being visited. The proposal is that the path is the best available evidence of the user's current context – after all, each of the objects on the path were chosen to be followed-up by the user. It is within that current context that the user will make his interpretations – e.g. of the relevance of objects presented as next-steps.

If one accepts the arguments that the path is a better representation of the current context than the single current object, then a representation of the context derived from the path could be used to influence the similarity measure. That would in turn influence the objects that were presented to the user as next-steps. If the path were a better representation of the developing information-need then it would be reasonable to expect the choice of next-steps to be more in tune with the changing information-need in the user's head.

The effectiveness of this approach is crucially dependent upon the manner in which the evidence collected along the path is used to restructure the information space. It is impossible for the system to have a full and accurate copy of the context used by the user to make his interpretations. Nevertheless, it is hoped that with appropriate methods of evidence identification, and combination, a system-context can be obtained from a path that is sympathetic to the user-context. Here, sympathetic is taken to mean that it approximates the general themes and directions taken by the user.

Different paths, different views

Accepting the idea of contextual interpretation of an object based upon the path taken to reach it, means that the intention/expectation of a user will be different depending upon that path. This means that users reaching the same object by different paths would want to see different objects around it (Fig 3.3).

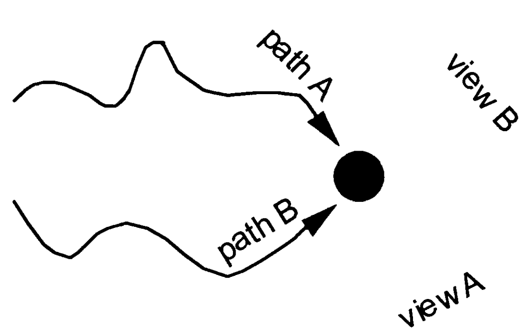


Fig 3.3 Two different paths to the same object and their different views of the next steps.

A user could reach an object via different paths for two reasons: because different paths lead to the same object; or because an object may be present at more than one point on the same path:

- **Different paths to the same object.** The appearance of an object on multiple paths might occur because the different paths represent different themes, with the object being appropriate to both. For example, imagine a user searching for information on nuclear waste dumping. He might follow a path that had documents relating to the technology of waste disposal (e.g. glass-impregnation, steel containers, monitoring systems) and encounter the same document that he encountered whilst following another path on the politics of the subject (e.g. government inquiries, local opposition to a dump, industry lobbying). The document in question may well talk of both aspects, and therefore be appropriate to both, but its interpretation by the user is likely to be different in the two contexts.

  When the user is following technical themes, the object will be interpreted with a technical bias. When the user is following political themes, the object will be interpreted with that political bias. These biases should be evident in objects of the respective paths (Fig 3.4).

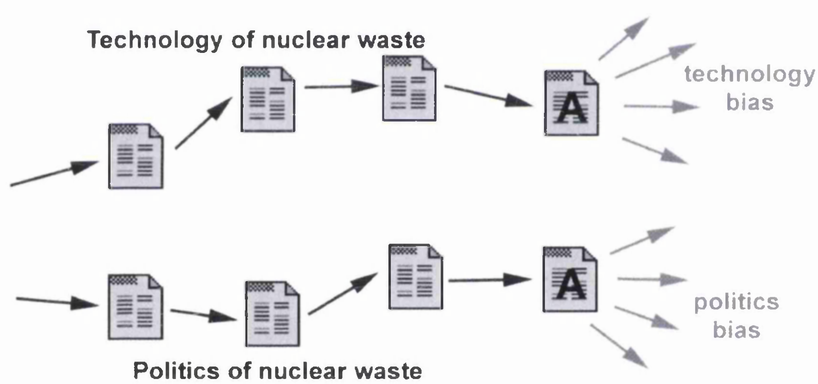


Fig 3.4 Different biases evident in different paths to the same object

- **The same object, different points in a path.** The appearance of an object at more than one point in a path could occur if the system thought that the document was again appropriate to be presented to the user as a next-step. The view on from the second (or further) occurrence of a particular object should be different from the previous occurrence(s) because the path leading to it is different.

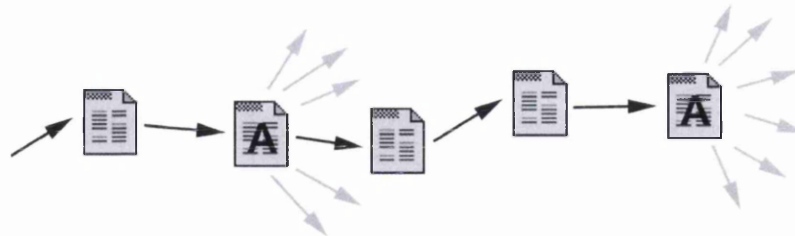


Fig 3.5 The same object at different points in a path

That second instance raises the issue of how soon, after an object has been the current object, it can be appropriate for it to appear again in the same path as a next-step. One simple 'fix' would be not to include the current object in its own list of next-steps. Similarly, the last few documents on the path are also candidates for exclusion. The question might become one of: After how many objects would the nature of the path have changed sufficiently for a repeat appearance of an object to drive the path forward into new and more relevant documents? As the evidence combination will be done using a concrete algorithm, the answer might well be an empirical one.

An interpretation-space

The multiple paths by which a particular object may be reached, and the different views from it in each case, may suggest that the object is present in multiple locations in the space. This seems inconsistent with the common understanding of a space – i.e. that points it are unique, fixed, and have the same view around it. Therefore, a user re-encountering an object would be quite reasonable in thinking that he had moved in a circle.

In the proposed approach, each object is the source of multiple interpretations – each of which is a result of the same object, but interpreted in different contexts. In fact, these interpretations make up the points in the space, not the objects. The user is therefore moving in an 'interpretation-space'. The interpretations are unique and thus the apparent inconsistencies are no longer present. As multiple interpretations of a particular object may appear on the same path, this interpretation-space is potentially infinite.

Such an interpretation space is novel – novelty brings in itself problems. It is possible that users will have difficulty in separating the idea of a document from the idea of one of its interpretation. The different interpretations are the next-steps that are presented around an object, not the objects themselves. A user looking at an object will not see any difference in it from other instances (there will be none to see). Therefore, they might not realise the difference in the next-steps from a previous one. As a result, they may be confused or disoriented by the apparent reappearance of the same document in several places in the space. Experimental evaluation may well be the only way to quantify this effect. The space might require techniques to be developed to encourage and reinforce the idea of browsing over interpretations and to make clear the secondary nature of simple object identity.

3.4 A proposed look-and-feel

This section, envisages a system that could present to the user support for developing information-needs. The ideas of the previous section and this section grew simultaneously and provoked the development of each other. Presenting them separately is an attempt at clearer organisation of the thoughts.

The description here is an idealised one, without the restrains of actual programming schedules, available libraries etc. It describes how it was thought a system using a path-based approach would look and feel – and is intended to provide a more clear idea of how the path-based approach will work in practice and what it has to offer in general. It will assist in the understanding of the model to be presented in Chapters 4 and 5.

3.4.1 Objects and links

The presentation will be that of a pane with objects and links placed upon it. Objects will be presented as icons, perhaps indicating some aspect of the information or its medium. For the purposes of this discussion, they will be simple document icons. Links will be presented as arrows. A path is a sequence of objects and links, with the arrows indicating the direction of the growth of a path. Path growth will generally be from left to right.

There will be the concept of the Current Object. A small number of candidate next-steps will be shown around the current object. Candidates will only be shown for that object – thus reducing considerably the visual clutter, with only those objects actually visited being shown on a path (Fig 3.5).

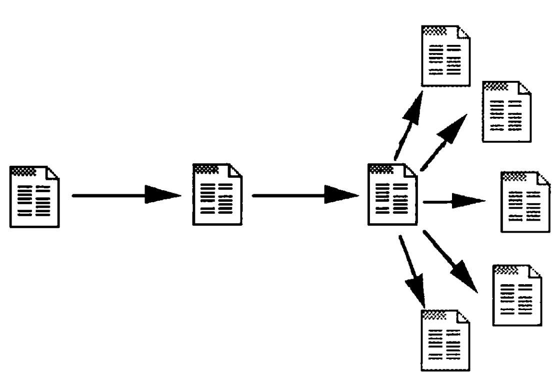


Fig 3.5 A three-document path, with five candidate next-steps surrounding the Current Object

The candidates presented will be selected based upon evidence collected from the path – i.e. they represent the computed view at that point. Only a small number will be presented to minimise clutter.

Backtracking to a previously visited object (i.e. making it the Current Object again) will show its next-steps again. This would occur, if at some point, more than one interesting object was seen and the user wanted to investigate them (Fig 3.6). This is how new, secondary, information-needs would be followed up, or alternatively how a user would backtrack to return to the primary information-need from which he had been diverted.

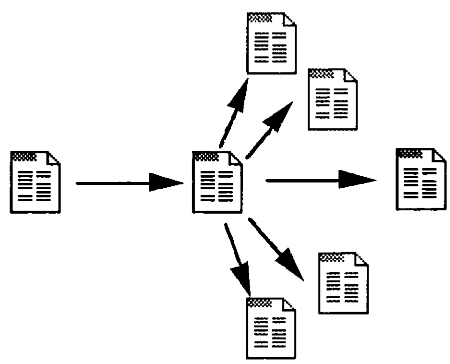


Fig 3.6 Backtracking to review the other candidates at a previously visited object

Each branch in the tree of paths is intended to represent a separate information-need – in a similar manner to the diagrams of Section 1.2. Each information-need can thus be seen to have a common heritage. This represents the tangentially splitting sub- and secondary- information-needs (Fig 3.7).

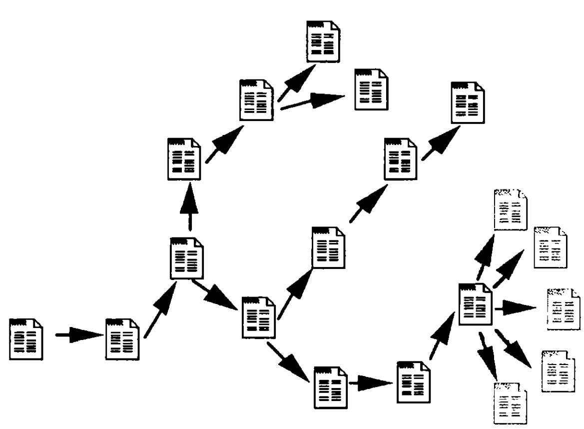


Fig 3.7 Four information-needs represented by four path branches

The user can at any time jump from object to object, review the candidates at that point and then investigate them further. This provides the support for multiple, embedded and threaded information-needs.

3.4.2 Avoiding clutter within a consistent layout

During a complex or thorough search, the user is going to generate many branches. Techniques will have to be developed to ensure that there is always sufficient display space for the branching, without destroying the overall layout. The maintenance of the general layout is key to providing orientation for the user – and thus essential for supporting the embedding and threading of information-needs.

One technique proposed is that of effectively 'tearing' the surface of the plane upon which the objects are drawn and 'folding' a set of branches underneath. This allows another set of branches to grow unrestrictedly on top. Instead of presenting a tear in the surface, a depth cue could be used to give the impression that the folded branches are further back into the display surface (Fig 3.8).

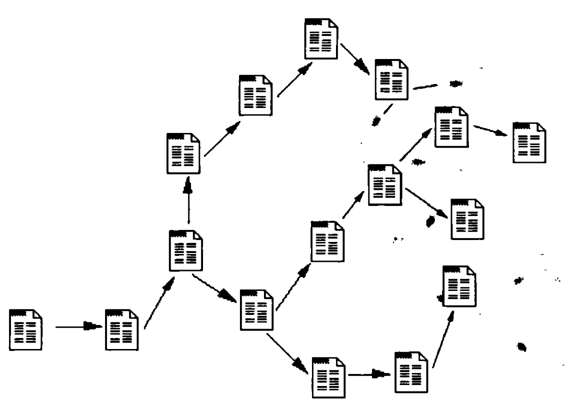


Fig 3.8 Folding (by depth cue) of one set of branches under another to allow growth

By applying a spherical transformation (known as a fisheye view [Lamping94]) to the display surface, the parts of the path-tree that are distant from the current object are reduced in size (Fig 3.9).

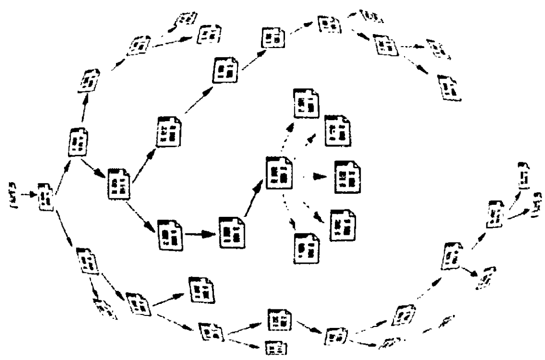


Fig 3.9 A fisheye view of the display surface

This means that the clutter is effectively brushed aside, providing a clear view of the area of immediate interest. The overall visual context and its orientation effects are still present however. That is, the general shape of the path-tree remains intact (although somewhat distorted) – e.g. sharp turns remain shape turns, and long linear stretches remain, more or less, so. An additional advantage is that all objects remain visible and on-screen at all times – regardless of the size of the display and the size of the path-tree.

Using this technique, it might be possible to avoid folding and tearing. As the current search path expands, the surrounding area could simply be pushed out of the way. This would involve modifying the display transformation function. A simple spherical transformation would not be sufficient. One problem is that, due to the branching of paths, the surface is much more cluttered at the right hand side than at the left.

3.4.3 Displaying object contents

There is clearly insufficient space available on the proposed display surface to allow the full contents of individual objects to be shown. Two things are proposed: firstly clicking to access or move to an object, and secondly, moving the mouse over an object to view its abstracted form without selecting it as the Current Object.

Clicking to access an object and its contents

The display would consist of two windows, the first would be a view of the display surface, and the second a document contents view (Fig 3.10). The document contents window would continually and automatically show the full contents of the current object (or more precisely, the contents of the basic object, one of whose interpretations is the current object on the display). The current object would be selected by clicking on it. Therefore, to view the contents of a particular object the user need only click on it. That object would immediately become the current object, its contents displayed and its candidate next-steps shown around it in the centre of the path display.

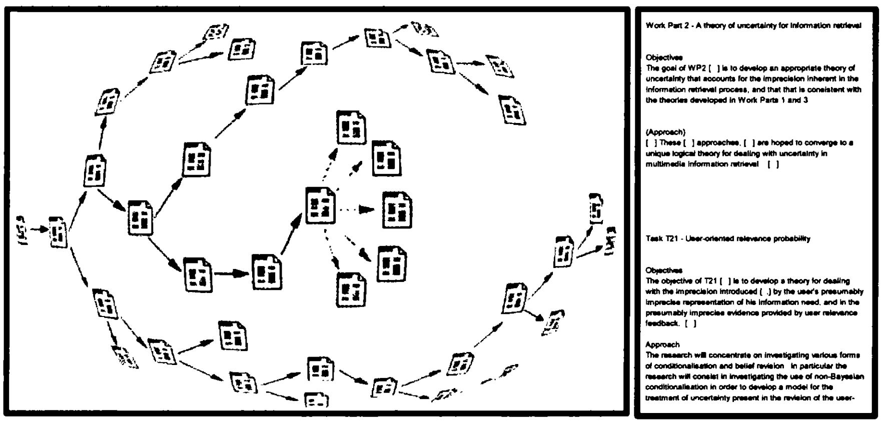


Fig 3.10 Separate display containing the contents of the current object

Roll-over to view an object's abstract

Simply moving the pointer over any object in the path display would result in a transient pop-up window appearing over the object (Figs 3.11 &3.12). This is similar to the 'tool-tips' effects of most modern programs, which offer a tiny piece of descriptive text when the pointer is moved over a button or control.

The pop-up window would remain displayed as long as the pointer was over the object, and would contain a highly abstracted form of its contents. This abstracted form would depend upon the object in question and might be, for example, the title of the document, a miniature version of the image, a short sound bite from an audio recording, a fragment of video, etc.

The combination of roll-over to view, and clicking to select, should allow the user to flick around the browsing paths at will, quickly and conveniently calling up the desired degree of information relating to an individual object, and subsequently act upon that information without the requirement to use or remember complex commands.

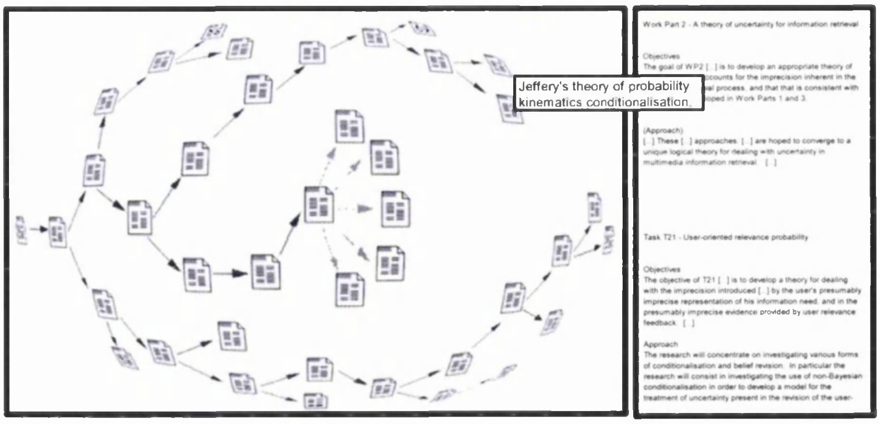


Fig 3.11 A text abstract being displayed on top of an object

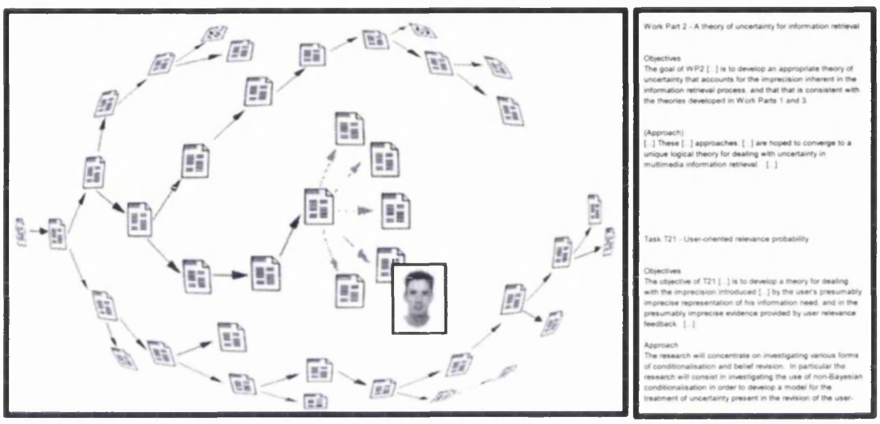


Fig 3.12 An image abstract being displayed on top of an object

3.5 Summary

This chapter began with the informal observations that were the seeds of the ideas of path-based query-free searching. Those observations recommended a concentration of support in a system for the core *functional* task that a user performs – i.e. identifying relevance. Conversely, they recommended striving for a reduction in the *procedural* aspects that are merely a side effect of the searching system – e.g. avoiding queries and their maintenance. Further, that same objection to procedural loading was made against systems using Relevance Feedback – i.e. managing sets of relevant documents is a similarly non-core procedural task to that of managing queries.

From the observations, it was proposed that a representation of a developing information-need might be captured from the objects that make up a 'path'. The objects in a path are considered as a concrete manifestation of an information-need – including the encoding of its development. It was proposed that the weight that each of the path objects has in such a representation should be reduced with their age. It was pointed out that this is essentially an exercise in evidence capture, combination, and application.

Contextual interpretation was highlighted as a novel feature of the path-based approach. This produces a space presented to the user that is infinite, and made up of interpretations, not of objects. It brings along both advantages (e.g. dynamism of the search space), and disadvantages (e.g. additional conceptual load of the interpretation space).

A proposed look-and-feel was presented to complete the picture of the envisaged environment. The look-and-feel displays the additional potential advantage of the path-based approach – i.e. there is no exposure of the user to internal representations.

The path-based approach could provide an environment in which a user could explore information objects, unaware of the system performing searching in the background. The user might even have the impression that he was in fact browsing a static structure of information objects arranged thematically.

Part II:

The Model

The chapters of this part are developments of the work published in [Campbell96].

In this Part, I present a model of the progressive development of information-needs. It is a model that recognizes the changing uncertainty inherent in a user's cognition of his information-need. The approach centres on the collection and combination of ostensive evidence. I present a new notion of relevance – Ostensive Relevance. This notion recognises the transient, inaccessible, spatio-temporal nature of relevance. I describe how these components come together to allow the Ostensive Model to be integrated with the traditional Binary Probabilistic Model. The integration reveals an implicit assumption in the conventional estimation procedure for a particular conditional probability. The temporal aspects of the Ostensive Model allow a weakening of that assumption. The integration allows direct implementation of the Ostensive Model. Finally, I present an example that demonstrates the intuitive appeal of the approach over existing approaches to Relevance Feedback.

4 A model of developing information-needs

This chapter presents the Ostensive Model of developing information needs. It is the formalisation of the ideas of Chapters 1 to 3. The model is of the iterative update of an information-need resulting from its exposure to information. The model identifies information objects on a browsing path as being observable manifestations of the information-need. The principle of ostension is applied to interpret evidence gathered from the path, placing profiles of uncertainty over those sources of evidence.

Contents of this chapter

Terminology and assumptions about the workings of a brain that underlie the model are presented in Section 4.1. The components of the model are described in Section 4.2. Section 4.3 makes the important distinction between observable and non-observable components of the model. Section 4.4 introduces ostension and its part in the model. Uncertainty in evidence is discussed in Section 4.5 – in particular, profiles of uncertainty with respect to age. Finally, Section 4.6 expands the discussion on ostension.

4.1 An abstract functional model of the brain

MacKay [MacKay69] proposed that one might consider the brain as a black box. This means that although one cannot understand the exact working mechanisms involved in cognition, one can still hypothesise about the effects of its operation indirectly through the inputs provided to it, and through the outputs resulting from it. That proposition has been important in the ideas of Chapters 1 to 3, but it becomes critical in the ideas of this chapter. The discussions here strive to avoid statements regarding *how* procedures involved in the internal workings of the brain, and restrict themselves to statements on what they do and how that manifests itself.

Information is defined by McKay as merely a sequence of bits, symbols, signs, etc. At that level, it is unchanging and absolute – i.e. if a modification is made to it, it becomes a different piece of information. This is quite different from the definitions used in fields such as library and information science where such matter is regarded as *data*, or *potential information* – that only becomes information when perceived by a cognitive agent [Ingwersen84]. In the MacKay view, no such distinction is made, and no such implicit importance is given to a cognitive agent – information is information regardless of the existence of agents, and it will remain so after any activities of such agents. This difference might only be a terminological one, around how the concept of 'interpretation' is encoded – i.e. data/potential-information when interpreted by a person becomes information. Therefore, to make it clear, the term 'information' in this chapter (and, in fact, in this thesis) refers to its un-interpreted 'raw' form.

Following the ideas of MacKay, the brain of a human is regarded as a black-box pattern-response mechanism. It is a probabilistic mechanism that has a *state of conditional readiness* to output certain responses conditional on certain inputs. These inputs and outputs are information, as is the instantaneous state of the brain itself.

The brain has inbuilt mechanisms that not only produce outputs as a result of inputs, but that also modify internal probabilities related to the particular input received – i.e. it recognises and adapts to patterns in inputs. This is termed an *internal matching response*. It is by that mechanism that the brain learns.

Given the input of information, *interpretation* is the process of the internal matching response, and *meaning* is the resultant change in the state of conditional readiness. Here, the brain (i.e. its state and matching behaviour) is the *context* within which the information is being interpreted to produce a meaning. The interpretive process that generates a meaning might or might not result in the output of information from the black box.

Information itself has no intrinsic meaning, but has as many potential meanings as there are contexts in which it could be interpreted. This point becomes important in Chapter 5, where the lack of any objective meaning *in* information is central to a new definition of relevance.

4.2 The components of the Ostensive Model

The model relates changes in the knowledge state of a user in response to information encountered during information seeking activities. The model is presented diagrammatically with associated propositions and assumptions. The core components of the diagram are shown in Fig 4.1.

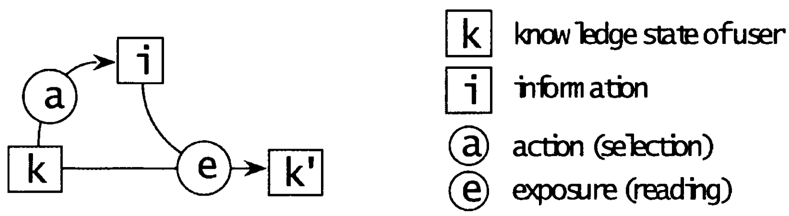


Fig 4.1 The updating of a knowledge state through the selection of, and subsequent exposure to, information.

Let *k* denote a knowledge state (i.e. an instantaneous state of conditional readiness). Within the context of information seeking, the following proposition is made regarding *k*:

> **Proposition 1:** Of all the motivating factors present in a user's state of knowledge, those influencing the immediate actions of the user to the greatest extent are those pertaining to the information-need most directly.

Let *a* denote the actions referred to in *P1*. With respect to those actions, the following is proposed:

> **Proposition 2:** The actions motivated by an information-need are most likely to be those that will obtain information from the environment that is regarded by the user to be the most likely to satisfy their information-need.

Let us restrict the environment within which this information seeking behaviour is taking place to that of a range of information items, each with an attached highly abstracted form – e.g. documents showing only titles. In such an environment, the possible actions available to the user are limited to selecting and reading documents. Here, *P2* would indicate the selection of the object that appears to the user to be the most likely to be relevant to the information-need – e.g. the document whose title or abstract suggested the highest relevance. Let *i* denote the information in that selected object.

The user would then expose himself to the information that made up that object – i.e. the user would read the document. This exposure of *k* to *i* would result (in all but the most trivial of cases) in an internal matching response. This internal update would therefore result in a new knowledge state *k'*. Let *e* denote that process.

In the Mackay view: *e* is the process of interpreting *i* with respect to, or within, a context *k*. That process of interpretation results in a change from *k* to *k'*. That change is the "meaning" of *i*. The meaning is therefore clearly dependent upon the *k* within which it is formed. Were it a different *k*, for example a different user, or the same user on a different occasion, then the meaning would be different. This dependence of meaning upon interpretive context is consistent with the 'cognitive viewpoint' of, for example [deMey77, deMey80, Belkin90, Ingwersen84, Ingwersen96].

The Ostensive Model does not speculate about *how* the processes involved in *e* proceed – it says only that a change from *k* to *k'* results from its exposure to *i*. Regarding the *nature* of the change, the following is proposed:

> **Proposition 3:** Given *P1* and *P2*, the majority of the changes in *k*, resulting from exposure to *i*, will be in those areas of *k* pertaining to the information-need most directly.

This new knowledge state *k'* might itself provoke an action *a'*, selecting a new information item *i'*, which through exposure *e'*, results in the further updated knowledge state *k"*. This process can iterate (Fig 4.2).

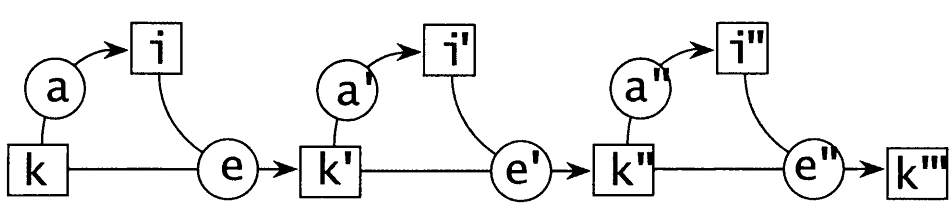


Fig 4.2 The iterative updating of a knowledge state.

4.3 Observables and non-observables

As already stated, the nature of *k* and the process *e* of it internally updating in response to incoming information is something that we cannot access directly, and of which we have little understanding. That nature and process are things about which we can currently only theorise based on actions resulting from them – i.e. the observed behaviour of a user.

With the above in mind, the components of the model can be classified into those that can, and those that cannot be observed directly (Fig 4.3).

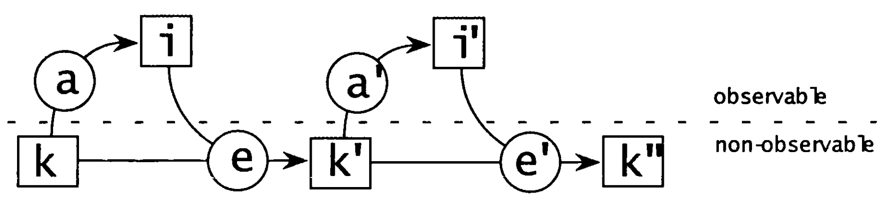


Fig 4.3 Observable and non-observable components.

One might consider the line separating the two classifications as representing the interface between the user and the outside world. Those components below the line are internal to the user and those above are external.

The action *a* of choosing from a selection (particularly a limited selection) of information objects can be observed, for example, in terms of the number and the nature of the objects rejected. The information *i* chosen (whether electronic or otherwise) can be observed trivially.

The other components (*k* and *e*), although not directly observable, might not be completely opaque. The following hopes to show that a limited grasp of certain qualitative aspects of *k* and *e* (that are of use in supporting information seeking) might be discernable from their observable manifestations.

P1 states that the action *a* is most strongly influenced by the information-need aspects of *k*. This means that *a* is indicative, in some way, of those aspects of *k*. Similarly, *P2* states that the chosen information *i* will be that regarded as most relevant to the information-need. Thus, *i* is also indicative of the information-need. *P3* states that the change from k to k', resulting from the exposure to i, will affect the information-need most. This means that i, not only being indicative of k (by *P2*), will also be definitive of k' – i.e. it will be the only factor apart from k that directly affects the nature of k'.

We now have a and i being indicative of their associated k, and i being strongly indicative of the associated k'.

The same principles could be applied to the relationship between the observables and e. Nevertheless, as e immediately produces a new k', and it is k' (the instantiation of the information-need that we are trying to capture) that then determines the next a' and i', a determination of e itself appears less interesting. Therefore, its presence is acknowledged but not investigated.

Ambiguity in the observables

Even upon unreserved acceptance of the above arguments regarding the relationships between observables and non-observables, it would seem unrealistic to expect to be able to glean a large amount of unambiguous information about k from single instances of a and i. Countless meanings could result from the tuple (a, i) and the uncertainty that would be attached to any single such possible meaning would therefore be high.

After several iterations of the k to k' process, several instances of a and i will have resulted and be available for observation. Individually, the possible meanings of each tuple would carry with them the same ambiguity. Nevertheless, when taken together, the tuples (unless made up of indistinguishable or identical instances of a and of i) would naturally demonstrate ambiguity resolution characteristics.

Similarly, taking together several documents indicated as relevant by a user, one can obtain a clearer idea of their interest than would be possible with only one. This principle is borne out in the efficacy of the Relevance Feedback process with which we are familiar – i.e. a larger number of relevant-indicated documents give a better statistical sample of the distribution of terms in relevant documents.

This idea will be returned to in Section 4.5.

4.4 Ostensive definitions from observable evidence

The Model provides an approach to capturing an information-need that is assumed to be developing during the course of a searching session. The capture of such spatio-temporal objects is inherent to the concept of 'ostension' [Quine53] & [Quine69].

The principle of ostension is summed up in the following entry in the Oxford English Dictionary [OED93]:

> **Ostensive definition** (Philos.), the explanation of a word by presenting, pointing at, or otherwise indicating one or more objects to which it applies.

The classic example is that of communicating a definition of a colour to a child. Adults might point to or hold up items to a child and say the word 'red'. There is no description happening, only ostension. The dynamic nature of such definitions can be exemplified in the changing nature of the child's conceptualisation of red. As he grows older, and is presented with more examples, and examples that are more sophisticated and subtle, his ostensive definition of red might change to exclude hues of red that we regard as 'pink'. That development of the conceptualisation of "red" could be described in pejorative terms as one of refinement or improvement. A non-pejorative development is exemplified in your changing definition of what someone likes to eat, based upon what they ask you to buy for them over an extended period.

Near ubiquity is claimed for ostension by Ayer when he says "The fact is that one cannot in language point to an object without describing it." [Ayer36].

Traditional Relevance Feedback is an ostensive process. Comparing with the dictionary definition: definitions are formed of an information-need (c.f. *"a word"*) by combining evidence from documents (c.f. *"one or more objects to which it applies"*) that the user has indicated as relevant (c.f. *"pointing at, or otherwise indicating"*).

This similarity of the process of definition of relevance, with that for a word, is perhaps stronger than it might seem. Quine [Quine60] talks of the ostensive definition of words "...being an implicit induction on the subject's part regarding society's usage...". An IR system, that incorporates Relevance Feedback, similarly performs an implicit induction of the user's use of relevance to indicate documents.

Observational ostension

Implicit in both the above dictionary wording, discussions of ostension (e.g. [Ayer36], [Quine53], [Ayer56], [Quine60], and [Quine69]), is the restriction of the evidence contributing to an ostensive definition to that of purposeful acts of communication.

It is now proposed that the restriction is lifted to allow other actions, not intended by the actor as communicative, to be included. This is exemplified in the ostensive definitions that we build of the kinds of clothes that other people like, simply by observing the repetition in, and the gradual change in, what they wear.

This weakening of the assumption/restriction is not proposed in order to increase the amount of evidence available, but instead to change the manner by which evidence is collected. In Chapters 2 & 3, the desire was expressed to remove the requirement of explanation of an information-need that currently burdens a user interacting with a query-based IR system. That motivates a switch in evidence from explicit communicative acts to observed actions. This, in turn, implies a move away from traditional Relevance Feedback, which relies upon communicative acts.

Ostensive definitions of information-needs could be built from observational evidence that has been collected, for example, from a user interacting with a browse-based IR system. Such evidence could be collected without interference with the user; without requiring description; and without exposing the user to any internal representation methods used by the system. It would allow the user to concentrate on the task in hand – i.e. identifying relevant information. A user might not be good at describing his information-need but is, by definition, able to identify something that is relevant to him. In fact, he is the only agent capable of doing so [Ayer56, page 63].

The idea proposed in Chapters 2 & 3 was that such ostensive definitions of the information-need be used to select appropriate information objects for the 'candidate next-steps' in a browse path. In that way, the system could adapt to individual users and information-needs throughout a session, but without the intrusive extraction of descriptions, and without even the (admittedly less intrusive) need to explicitly indicate the relevance or otherwise of individual objects. The act of a user selecting a next-step would be taken to be an implicit assessment of relevance. It is intended, therefore, to replace the communicative acts completely with observational evidence.

The candidate next-steps, would be the restricted environment, within which the action *a* of selecting an information object *i* would take place. The collected ostensive evidence would then lead to an ostensive definition of *k*', which would be used to predict the information objects most likely to be relevant to that *k'*. Those objects would be the next-steps and hence the next environment from which information is selected.

4.5 Uncertainty in ostensive evidence

The degree of uncertainty that we can attach to inferences about *k* and *k'* from *a* and *i* in each tuple *(k,a,i,k')* appears impossible to determine absolutely. Nevertheless, under certain conditions, comparisons can be made between the degrees of uncertainty associated with individual tuples:

Taking Fig 4.2 as an example, we have three iterations, each of which can be represented by a tuple: *t1=(k,a,i,k') ; t2=(k',a', i', k") ; t3=(k", a", i", k''')*. We also have the time ordered sequence: *t1,t2,t3* – i.e. *t1* occurred before (and led to) *t2*, which occurred before (and led to) *t3*. It is the central importance of this time sequence that sets the Ostensive Model apart from those implicit in current approaches. The time ordering of evidence will provide the key to relative degrees of uncertainty attached to individual pieces of ostensive evidence.

After the first iteration, the current knowledge state is *k'*. *a* and *i* are our best, and only, ostensive evidence from which to make inferences about *k'*. After the second iteration we will be trying to make inferences about *k"* based upon the accumulated evidence *a, i, a', i'*. After the third, inferences about *k'''* will be made based upon *a, i, a', i', a", i"*. Therefore, as the amount of ostensive evidence increases, our uncertainty of having an accurate representation of the information-need will reduce.

There are no grounds to say that, for example, the degrees of uncertainty associated with inferences within *t1* are any greater or less than those for the inferences within *t2* or within *t3*. For example, *a* and *i* say as much about *k'* as *a"* and *i"* say about *k'''* – i.e. the uncertainty of intra-tuple inferences are not comparable.

The model shows the causal path of all the ostensive evidence leading to the current knowledge state *k"*. There is clearly a more direct path from *a'* and *i'* to *k"* than there is from *a* and *i* to *k"*. Note, we are now talking exclusively with respect to *k"* – i.e. the current knowledge state. *k'* is now of little importance as it has disappeared.

With the directness of the causal links in mind, it seems reasonable to assume that the uncertainty attached to inferences about *k"* based upon *a* and/or *i* will be higher than the uncertainty attached to similar inferences based upon a' and/or i'. This can be generalised as:

> **Proposition 4:** As the age of an item of ostensive evidence increases, the uncertainty attached to inferences made upon it about the current knowledge state will increase.

Uncertainty profiles

P4 tells us that the uncertainty attached to evidence increases with the age of that evidence – the question arises as to the nature or form of that aging. In an effort to support further the suggested relationship and to give intuitions as to the meaning of such relationships in general, three classes of profile will now be discussed – a decreasing, a flat, and finally the preferred increasing profile:

Uncertainty decreasing with age

Consider first a profile where the uncertainty decreases with the age of the evidence (Fig 4.4). In all the profiles, the 'age' axis will increase to the left in order to retain the normal presentation of the passage of time as left to right. The zero point on the age axis is the current time, with the axis growing into the past.

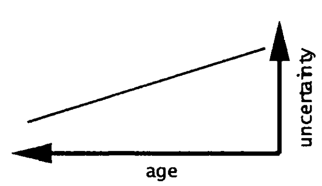


Fig 4.4 A decreasing profile of uncertainty w.r.t. age.

This profile implies that old evidence is more certain to be indicative of the current knowledge state than evidence that is more recent. This implies that early evidence should have the most influence on the ostensive definition, and that subsequently observed evidence becomes of less and less importance. In practice, the accumulation of additional evidence quickly becomes insignificant, with the ostensive definition changing little, if at all – i.e. the ostensive definition will effectively converge. This has the disturbing implication that the user knew what he wanted at the start, got it, and is now simply wasting time by continuing to expose himself to information – in effect, the precise antithesis of the Ostensive Model.

Uncertainty remaining constant

Conventional Relevance Feedback approaches assume that all the evidence (i.e. relevant-indicated documents) has an identical degree of uncertainty. This equates to a flat profile of uncertainty with age (Fig 4.5).

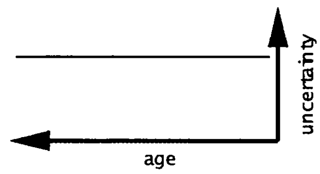


Fig 4.5 A flat profile of uncertainty w.r.t. age.

Using the terms of the Ostensive Model: Relevance Feedback implicitly assumes that either all evidence is generated by the same knowledge state, or that all the generating knowledge states are identical. In the Ostensive Model, such a situation is impossible, as exposure to one document changes the knowledge state and thus increases the uncertainty associated with it before another document can be observed as evidence.

In practice, the combination of the evidence in Relevance Feedback is process of accumulation, with each observation contributing the same incremental amount of evidence. This results in the ostensive definition converging as the amount of already accumulated evidence gradually overwhelms any new evidence. This is a similar situation to that of the decreasing uncertainty profile, although less extreme.

Uncertainty increasing with age

The profile suggested by the Ostensive Model is one of increasing uncertainty with age. Such a profile is shown in Fig 4.6.

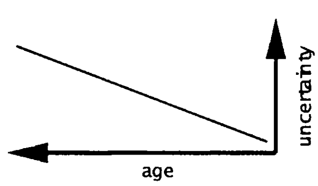


Fig 4.6 An increasing profile of uncertainty w.r.t. age.

This corresponds to the Ostensive Model in that the most recent evidence has the lowest attached uncertainty and will thus should have the most influence on the ostensive definition. Here, all evidence plays a part in the definition, but the most recent will have the greatest effect. This means that the ostensive definition will follow recent trends in the ostensive evidence, but will always have a component of the historical evidence.

The rate at which the uncertainty attached to evidence increases with age can be altered whilst retaining the general increasing nature required by the Ostensive Model. Further, the rate of change of uncertainty can also be modified within a particular profile class – Fig 4.7 shows two such profiles.

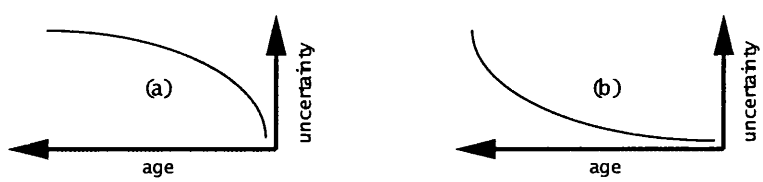


Fig 4.7 A decelerating (a), and an accelerating (b) increase in uncertainty with age.

If the bias of Fig 4.7a were taken to an extreme, we would have a step function where there would be zero uncertainty attached to the most recent evidence and maximum uncertainty attached to all other evidence. That would correspond to the models implicit in traditional browsing systems, where the links available at any given object are based upon pair-wise accessibility relationships between other objects in the space and the Current Object alone.

Paths and profiles

As presented in Chapter 3, the intended application environment for the Ostensive Model is that of a browsing system where the user moves from node to node (i.e. information object) via links (i.e. accessibility relationships). The links will be generated dynamically based upon an ostensive definition of the information-need. The definition will be made from the evidence collected along the path to that node. The path will be the sequence of nodes that was followed by the user.

The rapid/gentle bias mentioned above effectively means balancing the information relating to how the user got to a particular object (i.e. the path) against the information in the object itself.

There is clearly a range of 'increasing with age' profiles of evidential uncertainty. The Ostensive Model does not specify any particular one – it tells only that the profile should be of that general form. Perhaps an appropriate bias reflects more the details of the process e of interpreting information than we are able either to measure or to understand. It seems a clear candidate for empirical determination.

4.6 The three elements of ostension

Quine [Quine53] proposes (almost as an aside) three components necessary for ostension to be used to capture spatio-temporal concepts (such as developing information-needs). He first requires 'pure ostension' which equates to simple observed evidence, then he requires 'identification' which refers to the recognition of identity[1] of the concepts being defined by the individual acts of pure ostension, and finally he requires 'induction' which is the process of combining the evidence (i.e. essentially: interpretation).

The Ostensive Model has pure ostension in the form of objects indicated and not indicated as relevant. It has identification in the form of the assumptions inherent in the propositions *P1*, *P2*, and *P3* – i.e. that all observed acts (within the restricted environments outlined above) are ostensive with respect to the information-need. It has induction in the form of the uncertainty profiles placed across the evidence. The profiles determine the manner of its combination. The mechanics of combination is not part of the model as described here. Chapter 5 will show the Ostensive Model being integrated with the Probabilistic Model of IR to provide operational ostensive induction.

[1] This concept of identity is that of "sign", not of "object" [quine60, page 116].

4.7 Summary

An iterative model of knowledge/information-need update based on information exposure was presented. Components of that model were distinguished as being either observable or non-observable.

The concept of ostension and ostensive definitions was presented. The idea that relevance is a spatio-temporal concept suited for ostensive definition was introduced. An implicit restriction of evidence to purposeful communicative acts in traditional conceptions of ostension was highlighted, and it was motivated that this restriction be lifted. The lifting would allow purely observational evidence to be used for ostensive definition – e.g. evidence based upon user-selection of relevant objects.

The difference between the Ostensive Model and traditional Relevance Feedback was clarified as twofold: 1) the Ostensive Model exploits passive/observational ostension rather than the active/indicative ostension; 2) the Ostensive Model treats relevance as a spatio-temporal concept rather than as merely spatial.

Uncertainty in ostensive evidence was described and three classes of profiles of that uncertainty were discussed.

Finally, the three elements of ostension (indication, identification, and induction) were identified with their counterparts in the Ostensive Model.

5 Implementing the Ostensive Model

In this chapter, the integration of the Ostensive Model of developing information-needs with the Binary Probabilistic Model of IR ([Maron60], [Robertson76], and e.g. [vanRijsbergen79], and [Crestani98]) is presented. The result of the integration is a concrete operational retrieval model. It provides the final part of the induction that completes the three necessary elements of ostension.

The integration is achieved through adding information from the Ostensive Model into the estimation of a parameter in the Probabilistic Model. This weakens an assumption previously made in the estimation of that parameter.

A new formulation of relevance is presented – that of Ostensive Relevance. It is a conception that recognises the inaccessibility of relevance, but it is one that is operational in essence, and one that incorporates time and the importance of the *current* information-need.

Contents of this chapter

Section 5.1 offers a short description of the aspects of the Binary Probabilistic Model that are of interest here. In Section 5.2, the assumption made in the estimation of a parameter is highlighted, and a weakening of it proposed. Ostensive Relevance is introduced in Section 5.3. Section 5.4 presents the integration of Ostensive Relevance into the parameter estimation of the Probabilistic Model. Finally, Section 5.5 offers an intuitive example that highlights the advantages of the integration.

5.1 The conventional binary probabilistic model

Following the derivation given by van Rijsbergen [vanRijsbergen79], we have for an information object D_j and a set of desired features x_1 to x_t, a linear decision function relating the probabilities of observing relevance and of observing non-relevance given a particular object:

$$\log \frac{P(Rel \mid D_j)}{P(NonRel \mid D_j)} = \sum_{i=1}^{t} w_i \cdot x_{ij} + C$$

where

$$x_{ij} = \begin{cases} 0 \text{ ; absence} \\ 1 \text{ ; presence} \end{cases} \qquad w_i = \log \frac{p_i(1-q_i)}{q_i(1-p_i)}$$

C is constant for all objects with respect to a particular set of desired features – therefore, it does not affect the resultant ordering of objects, and is typically ignored in implementations.

There are two random variables:

$$p_i = P(x_i = 1 \mid Rel)$$

$$q_i = P(x_i = 1 \mid NonRel)$$

which are the probabilities of observing a particular feature given that we have observed relevance, and non-relevance, respectively. Traditionally, they have been estimated by counting within the set of objects that have been indicated as relevant by the user. For example, the estimation for p_i:

$$P(x_i = 1 \mid Rel) = \frac{P(x_i = 1 \wedge Rel)}{P(Rel)}$$

$$\text{Estimates:} \quad P(x_i = 1 \wedge Rel) = \frac{r_i}{N} \quad , \quad P(Rel) = \frac{R}{N}$$

$$\text{gives} \quad P(x_i = 1 \mid Rel) = \frac{r_i}{R}$$

with respect to the standard contingency table:

	Relevant	Non-Relevant	
$x_i=1$	r_i	n_i-r_i	n_i
$x_i=0$	$R-r_i$	$N-n_i-R+r_i$	$N-n_i$
	R	$N-R$	N

i.e. the probability of observing a feature given relevance is estimated as the proportion of the relevant-indicated objects that contain that feature.

A similar process is followed for q_i producing the proportion of objects *not* indicated as relevant (and hence assumed to be non relevant) that contain the feature:

$$P(x_i = 1 \mid NonRel) = \frac{n_i - r_i}{N - R}$$

It is through an alternative and an arguably more appropriate estimation procedure for the conditional probability p_i that the uncertainty profiles of the Ostensive Model are incorporated.

5.2 Not all evidence is created equal

Imagine a set of six objects indicated as relevant by the user (Fig 5.8), three of which contain a particular feature x_i (i.e. $R=6$, $r_i=3$). Following the derivation of the conventional binary probabilistic model given above, the estimate of the probability of observing the feature given relevance will therefore be $p_i=0.5$

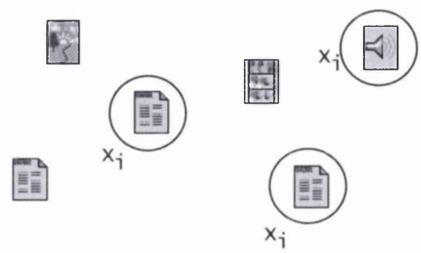


Fig 5.8 A set of six information objects marked as relevant, three of which contain a particular feature.

Implicit in the conventional approach is the assumption that all objects marked as relevant are equally useful and appropriate as sources of evidence for the estimation of p_i. There is no account taken of any property that could affect their individual appropriateness. The Ostensive Model is opinionated with respect to the appropriateness of the individual objects. That opinion is based upon the 'age' of the evidence – i.e. the age of the relevance-indication, not of the information-object itself.

If we take those same six objects and structure them according to their ages, we have a *sequence* of relevance indications. For example, Fig 5.9 could be the result of such an ordering.

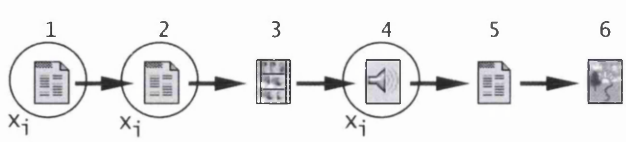


Figure 5.9 A time-ordered sequence of six information objects marked as relevant, three of which contain a particular feature.

This is a spatio-temporal record of objects regarded as relevant by the user. The Ostensive Model regards that record as an ostensive definition of the development of the information-need – i.e. the sequence is the observable manifestation of the information-need. With this structure across the objects, we can associate degrees of uncertainty according to an "increasing with age" profile, as discussed in Section 4.5, (Fig 5.10).

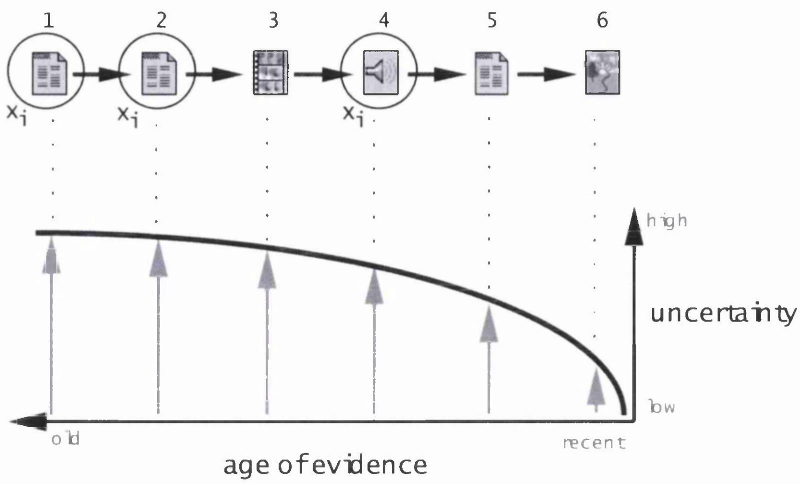


Figure 5.10 Age-based degrees of uncertainty associated with six time-ordered pieces of evidence.

From the diagram, we see that the most recently indicated object has the lowest attached uncertainty – i.e. we have the highest confidence in it being representative of the current information-need.

In conventional systems, objects are indicated as relevant at various times and are not the only objects observed in detail by the user. In the Ostensive Model, it is assumed that they are observed immediately after each other (as a result of the restricted environment within which the information seeking takes place). The arrows in Figs 5.9 & 5.10 emphasise the path nature of the sequence of objects.

5.3 Ostensive Relevance

There has been recognition made of the dynamic nature of information needs, and cases made for its adoption as the norm ([Belkin82], [Bates89], [Ingwersen96], and [Borlund98]). Nevertheless, clear intuitions of *how* such dynamism might be incorporated into a retrieval algorithm in order to improve its effectiveness have remained illusive. The exception, is the idea of Polyrepresentation ([Ingwersen94] & [Ingwersen96]), which recognises contextual interpretation of information as being inherent, ubiquitous, and unavoidable. It recognises the multiplicity of relevance that results from contextual interpretation. Polyrepresentation is conceived primarily from the perspectives of an author and of a searching user. Ingwersen appeals for support for multiple interpretations to form an integral part of searching systems. Polyrepresentation, as presented, is an essentially spatial conception – but if it is extended to become spatio-temporal, the Ostensive Model can be seen to be a consistent part of that broader framework.

The Ostensive Model not only recognises, but also defines, a dynamic notion of relevance. Further, the Ostensive Model recognises that relevance cannot be determined in advance – its nature will only become apparent at the instant that it is formed in the brain of the person making the relevance assessment. Finally, the model recognises the inaccessibility of relevance – i.e. the inability to directly observe the knowledge state of the user and the actual perception of relevance within it. It can only be observed 'from a distance' through its external manifestations – i.e. through ostensive observable evidence.

With these thoughts in mind, a new notion of relevance is defined from the point of view of an external agent attempting to assist the searcher (which in this case, is an IR system, but that might equally be another person).

> **Proposition 5:** The Ostensive Relevance of an information object is the degree to which evidence from the object is representative/indicative of the current information-need.

Ostensive Relevance is an operational conception of relevance: To talk of "evidence from the object" inherently implies a process of evidence identification and extraction. Similarly, to talk of "representative/indicative" is to admit to a similar process of application of that evidence. Both of these processes have a utility that is key to the relevance measure. It is also a temporal conception: it references explicitly to the *current* information-need.

Beyond those characteristics of Ostensive Relevance, it is possible to attempt to classify it in one of the many classifications that have appeared in the literature – a recent and comprehensive survey of such work can be found in [Mizzaro97]. There are problems with classifications; for example:

The inferential nature of the process suggests that Ostensive Relevance can be categorised as a 'System or Algorithmic' notion of relevance. Nevertheless, the intentional flavour of "representative/indicative" suggests that it might be better categorised as a 'Cognitive or Pertinence' relevance. But there again, as the current information-need is inextricably linked with motivational aspects such as the user's current situation and task, perhaps 'Situational relevance' would be a more appropriate categorisation.

The above classifications happen to be from [Saracevic96], but the following criticisms are intended generally. Classifications can easily become blurred, and regarded merely as a form of error-prone shorthand. The very nature of relevance is such that there is unlikely ever to be a universal conception of it. Therefore, would it not be better to relegate relevance to simply a definition of what makes a particular system choose one document over another? This would put it on a par with, for example, the definitions of how probabilities are estimated from evidence, how evidence is collected, or how documents are represented. Classifications of relevance, and discussions based upon them, have not produced operational improvements. Essentially: classification activity is no more than just that – classification. Taking that to heart, no more will be said about how Ostensive Relevance relates to other conceptions.

Ostensive Relevance, in effect, captures how 'ostensive' a piece of observed evidence is. This notion is only valid when evidence is observed from objects that the user selects under the conditions set out in the abstract description of the model (i.e. under propositions *P1* to *P4* of Section 4.2 & 4.5).

Such confidence can be expressed in terms of probability. If we take a binary view of this notion of relevance (i.e. 'representative' or 'not-representative') – we can talk of the probability of observing that relevance, i.e. *P(Rel)*.

That probability of relevance can be measured at each information object D_j and therefore expressed as a conditional probability – $P(Rel|\ D_j)$. The value of $P(Rel|\ D_j)$ is inversely related to the ostensive uncertainty at D_j. This produces an opinionated function of $P(Rel|\ D_j)$. Fig 5.11 shows the relationship between evidential uncertainty (as prescribed by the Ostensive Model), and the object-conditional form of Ostensive Relevance – i.e. they are inversely related.

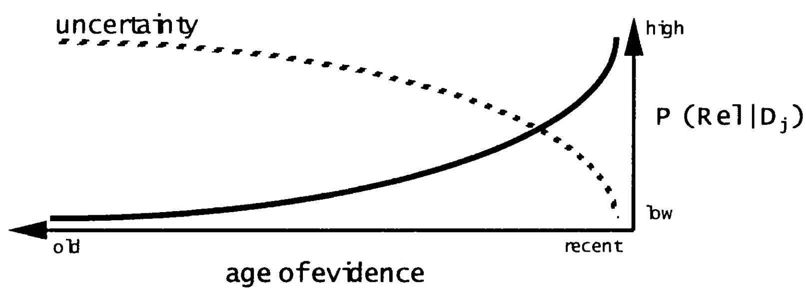


Figure 5.11 The inverse relationship between uncertainty and the object-conditional form of Ostensive Relevance.

The total probability function shows the relationship between the overall probability of Ostensive Relevance and the individual object-conditional probabilities:

$$P\ (Rel) = \sum_{j=1}^{R} P\ (Rel\ |D_j) \bullet P\ (D_j)$$

5.4 Incorporating the ostensive evidence

The interpretation of the evidential uncertainty profile as an object-conditional probability function of Ostensive Relevance provides the key to incorporating the ideas of the Ostensive Model into the conventional Binary Probabilistic Model.

The new estimation procedure is developed for the random variable p_i:

$$p_i = P(x_i=1 \mid Rel) = \frac{P(x_i=1 \wedge Rel)}{P(Rel)}$$

Applying the total probability function allows the above probabilities to be expressed in terms of their component probabilities at each of the R relevant-indicated objects:

$$= \frac{\sum_{j=1}^{R} P(x_i=1 \wedge Rel \mid D_j) \bullet P(D_j)}{\sum_{j=1}^{R} P(Rel \mid D_j) \bullet P(D_j)}$$

The notation can be simplified slightly by defining new probability functions at each D_j. That is, $P(x_i=1 \wedge Rel \mid D_j)$ becomes $P_{Dj}(x_i=1 \wedge Rel)$. In the absence of evidence to the contrary, the prior probability of observing any one document from the collection, $P(D_j)$, can be assumed to be identical for all D_j, and can be cancelled[1]:

$$= \frac{\sum_{j=1}^{R} P_{D_j}(x_i=1 \wedge Rel)}{\sum_{j=1}^{R} P_{D_j}(Rel)}$$

Assuming the conditional independence of x_i and Rel with respect to D_j:

[1] If there were evidence to the contrary (i.e. that leads to $P(D_j)$ varying over D_j), the term could simply be retained, with no interference with the derivations that follow in this section.

$$= \frac{\sum_{j=1}^{R} P_{D_j}(x_i=1) \cdot P_{D_j}(Rel)}{\sum_{j=1}^{R} P_{D_j}(Rel)}$$

At first sight, this appears to be a disturbing assumption to make, as it is the very relationship between the observation of features and of Relevance that underlie all models of retrieval.

Nevertheless, the independence that is being assumed here is *not* across all documents in the collection, not even across the relevant-indicated documents – it is assumed only within the confines of a single relevant-indicated document.

If one places the restriction on $P_{Dj}(Rel)$ that it is non-zero (i.e. that none of the 'relevant-indicated' documents have their relevance set to zero by the Ostensive Relevance profile) then Relevance will always be present/observed in a Boolean sense (although its precise value will vary). Therefore, the value of $P_{Dj}(x_i)$ can take any value and be said to be independent of $P_{Dj}(Rel)$.

That restriction on $P_{Dj}(Rel)$ is not exceptional. Probabilities of *0* and of *1* can be troublesome – e.g. Jeffery appeals for their avoidance on the grounds that they leave no room for functional revision away from those end-points in the light of new evidence [Jeffery65]. In fact, one could argue intuitively that probabilities of *0* and *1* make no sense, except in the retrospective case.

The design of Ostensive Relevance profiles should take this restriction into account if they are to remain theoretically sound – e.g. by reducing $P_{Dj}(Rel)$ to extremely low values instead of zero.

$P_{Dj}(x_i=1)$, the probability of observing the feature that x_i represents, given the observation of D_j, is trivially observable as equal to the value of x_i at that object.

$$= \frac{\sum_{j=1}^{R} x_{ij} \cdot P_{D_j}(Rel)}{\sum_{j=1}^{R} P_{D_j}(Rel)}$$

This leaves the individual object-conditional probabilities of observing Ostensive Relevance, $P_{Dj}(Rel)$, to be inserted. Therefore, for each relevant-indicated information object D_j we need only substitute the value of x_{ij} at that object and the value of $P_{Dj}(x_i=1)$ from the uncertainty profile of the Ostensive Model.

In effect, the numerator works by 'switching' the $P_{Dj}(Rel)$ component 'on' for each object containing the feature x_i. The size of that component is determined by the Ostensive Model's opinion of the uncertainty attached to the source of that evidence. The denominator normalises those components into a 0..1 range. Thus, each x_i-containing relevant-indicated object contributes an amount towards the estimation of p_i depending upon its probability of Ostensive Relevance.

5.5 An intuitive description and example

This section shows an example of how the incorporation of an uncertainty-profile/Ostensive-Relevance-profile affects the final estimations given to the random variable p_i. To do this, we will create a profile and then compare its effects to that of the conventional estimation procedure, when they applied to two different sequences of relevant-indicated objects.

Let us adopt a profile and resulting probability function of the form outlined in Figs 5.10 & 5.11. The probability function could be something like 2^{-a} where *a* would be the age of the evidence (e.g. the number of steps since the object was indicated as relevant). This might give a sequence of six object-conditional probabilities such as *{1/64, 1/32, 1/16, 1/8, 1/4, 1/2}*. The actual values are not important for this example, only that they follow the general profile suggested by the Ostensive Model.

Adopting the object sequence of Fig 5.9 as the first test sequence, and inverting the presence/absence of features to produce the second test sequence, gives the two sequences shown in Fig 5.12.

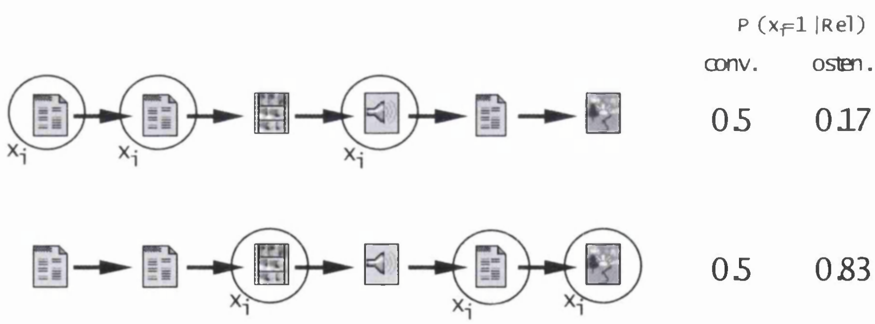


Figure 5.12 Two sequences and their respective p_i values from the conventional and from the ostensive estimation procedures.

The conventional procedure would, as previously shown, produce an estimate for $P(x_i|Rel)$ of *0.5* for the first sequence. The Ostensive Model's estimation would be *0.17*.

For the second sequence, the conventional procedure would again produce an estimate of *0.5*. The Ostensive Model, however, would produce *0.83*. These values are shown in Fig 5.12 alongside the sequences for comparison.

The conventional approach sees no difference between the two sequences. In fact, it does not see the sequence at all – only an unordered bag. It simply identifies half of the objects as having the feature and the other half as not having it, and so estimates the probability of observing the feature in any future relevant objects as *0.5* for both sequences.

Looking at the first sequence, the occurrence of the feature is more evident at the beginning, less evident in the middle and not evident at the end. Conversely, in the second sequence the feature is not evident at the start, but becomes more evident towards the end. It can be argued that intuitively one would expect to see the feature in an, as yet unseen, object more highly in the second sequence than in the first.

If one accepts those intuitions, then the probabilities produced by the Ostensive Model appear more appropriate than do those of the conventional approach.

5.5 Summary

Ostensive relevance – an operational and temporal notion of relevance captures the utility of evidence in an object with respect to building an effective representation of the current information-need.

The uncertainty profiles of Section 4.5 were translated into profiles of Ostensive Relevance, as they are essentially inverses of each other. It is arguably easier to talk of 'Ostensive Relevance' than 'Ostensive Uncertainty', as it is a positive rather than negative concept.

Using the profiles of Ostensive Relevance, the Ostensive Model can be integrated with the Binary Probabilistic Model to form an operational retrieval model. The integration places a minor restriction that values of Ostensive Relevance used for calculation must not be zero.

The Ostensive Model produces probability estimates within the Binary Probabilistic Model that appear to be more intuitive, and in keeping with the developing nature of an information-need.

Part III:

The System

In this Part, I present the IR system built for, and used in, the work of this thesis. The system was used for the evaluations to be detailed in Part IV. It follows the client-server model – therefore the two components are presented separately:

Chapter 6, presents the server portion of the system – a stand-alone piece of code that can service multiple requests on multiple document collections from multiple clients. It implements the binary probabilistic retrieval model, traditional relevance feedback, and, incorporating the work of Chapter 2, it supports relevance feedback using partial relevance – thus allowing it support the Ostensive Model.

Chapter 7 presents the interface – a fish-eyed, media-neutral, query-free browsing environment. The look & feel and the manner of use of the interface are described, along with the extensions made to specifically support the low-cost evaluation that will be described in Part IV.

6 An ostensive probabilistic IR server

The IR Server described here resulted from a process that was started in response to the difficulties of using the research IR engines that were available at the time (1989). In fact, there was only one widely available engine, the SMART system [Salton71]. SMART was a collection of programs that had grown incrementally with the experiments of the group surrounding it. It was rejected because: it was difficult to set up; difficult for customisations to be incorporated into a system that had grown to be quite complicated; and finally because it was primarily a Vector-Space retrieval system – which did not fit with the probabilistic experiments for which the IR system was required.

Textual versus non-textual

The Server was conceived originally as a text retrieval system. Modifications have since allowed it to index and retrieve objects of other media that have associated textual descriptions. Nevertheless, the operation of the system remains essentially text-based. Therefore, the following descriptions will assume text documents and textual queries.

The contents of this chapter

The overall architecture of the IR Server is described in Section 6.1. Section 6.2 describes the Indexer component. The Retrieval Engine component is described in Section 6.3. In an effort to bring the thoughts together, Section 6.4 provides a short description of the steps through which a notional client would interact with the engine to perform common retrieval tasks. Section 6.5 lists some clients (i.e. user-interfaces) that use the Engine.

6.1 The architecture

The IR server consists of two components: the *Indexer* that takes a collection of documents and builds indexes of their contents; and the *Retrieval Engine* that uses those indexes to find documents that are appropriate to requests that it receives over an Internet connection – Fig 6.1.

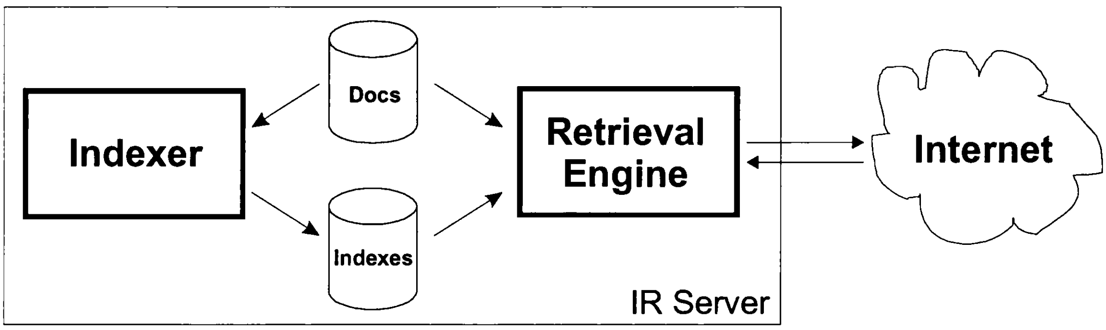


Fig 6.1. The two components of the server.

Implementation environment

Both components of the server were written in the *ANSI C* programming language for the *Solaris* (i.e. Unix) operating system. The standard C libraries were used to interface with the Unix file-system, and the standard Unix/C 'Sockets' libraries provided the interface to the TCP/IP networking.

With the exception of those libraries, the IR Server constitutes approximately eight thousand lines of code.

In addition, two versions of the Retrieval Engine have been ported to the *Apple Macintosh* operating system. One version acts in an identical manner, using a TCP/IP connection to communicate with a client, and the other version was embedded into a standalone application that had a graphical user-interface (more details are presented in Section 6.5). To use the MacOS versions of the Engine, the indexes must first be built on a Unix/Solaris machine and the resulting files made available to the Engine on the MacOS machine through either file-transfer or file-sharing.

A client system using Retrieval Engines on either platform will see no difference in the manner of connection, interaction, or format of data – such issues have been standardised (i.e. hidden) within the Engine.

6.2 The Indexer

The Indexer is run once on a static collection of documents. It scans the directories and files containing the raw documents, and produces two indexes: the *Document Index* that records the location of each document; and subsequently, the *Word Index* that records the occurrences of words across the documents – Fig 6.2.

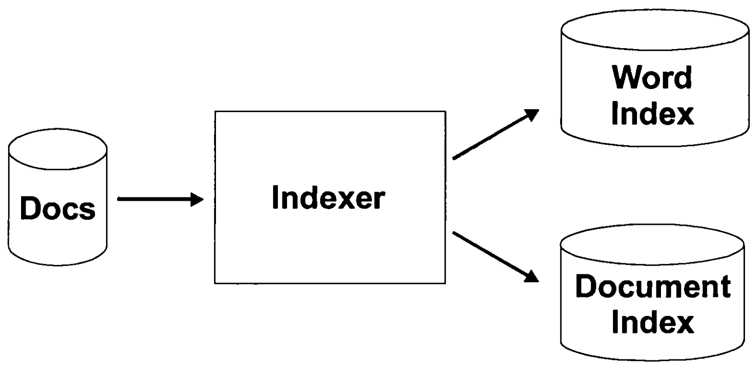


Fig 6.2. The Indexer produces a Word Index and a Document Index.

The Document Index

The Indexer assigns a *Document Identifier* (i.e. serial number) to each document that it encounters. The identifier is unique within a document collection, but will be reused in other collections. The index contains a mapping from those identifiers to the actual location of the respective documents. In the simplest case, the index would map a Document Identifier to a filename.

As most collections that the system has used have been a delimited set of text documents within a single file, there is an alternative format of the Document Index where each entry is made up of a start- and end-position. All such start- and end-positions in the Document Index refer to the same file containing the documents.

The Document Index is there to simplify and speed-up access for the Retrieval Engine to the full contents of any individual document, based upon its document identifier. Once this index has been created, all references to documents within the IR Server are made using Document Identifiers only. Once created, the Document Index is used by the Indexer itself whilst producing the Word Index.

The Word Index

The Word Index is an inverted file – i.e. the collection of documents, each containing words is 'inverted' into a list of words, each of which contains documents. The representation is 'binary' with respect to word-document occurrences – i.e. only the presence/absence of a word in a document is recorded, not any indication of the number of times. Hence, the inverted-file index is a list of all words appearing in the collection of documents, with each of those words having a list of the Document Identifiers of those documents in which it occurs at least once.

This index is made up of a *word-file* and an *occurrence-file.* The word-file contains the list of words, sorted in alphabetical order. The occurrence-file contains the lists of identifiers for those documents in which particular words occur. Each entry in the word-file contains the word, the number of documents in which the word occurs, and a pointer to the location of its list of occurrences in the occurrence-file.

The alphabetical-sort of the word-file allows the Retrieval Engine to rapidly read it into memory ready for efficient binary-searches to be performed on it to look up query words.

Lexical scanning

When parsing the document texts, the indexer will treat as a 'word' any sequence of alphanumeric characters of a length greater than or equal to three. Alphanumeric characters are regarded as: 'a' to 'z', 'A' to 'Z', and '0' to '9'. All other characters are treated as word separators. The system will *not* correctly handle accented, umlauted, and other non-English language characters.

Hyphenated words are treated as separate words – an arbitrary decision that, if it has a measurable effect, is likely to increase recall and reduce precision. For example, 'house-party' would be split into 'house' and 'party'. This would mean that, as a query term, it would match with documents about any kind of 'party' and with documents about any kind of 'house' – thus reducing precision. On the other hand, it would successfully match with "house party" – i.e. it would match with inconsistent use of hyphenation – thus increasing recall.

Stemming

The Indexer can apply a stemmer to the words that it encounters. If it is switched 'on', all words are stemmed before they are processed. That means that the word-file will actually contain stems rather than words, and the lists in the occurrence-file will be of documents in which words occurred that share the same stem. As a result, the number of words (and occurrence lists) will be less, but the average length of the occurrence lists will be slightly higher.

The stemmer implemented is the vowel/consonant counting algorithm of Porter [Porter80] – which is targeted to English language words.

Stop Words

A file containing a list of 'stop words' can be supplied to the Indexer. All words on that list will be ignored during indexing. This is essentially an efficiency measure, used to prevent commonly occurring words (that are less likely to be useful in retrieval) from taking up space in the word-file and, more crucially, in the occurrence-file. If these words are regarded as unlikely to be of use in retrieval, then time can be saved at retrieval time by not having to process their long occurrence lists.

6.3 The Retrieval engine

The retrieval engine accepts requests over an Internet connection and carries them out using the collection of documents and the indexes that were generated by the Indexer. The requests, and the Retrieval Engine's responses to them, conform to a protocol that is designed to support searching using weighted-term queries, traditional relevance-feedback, and ostensive-relevance – Fig 6.3.

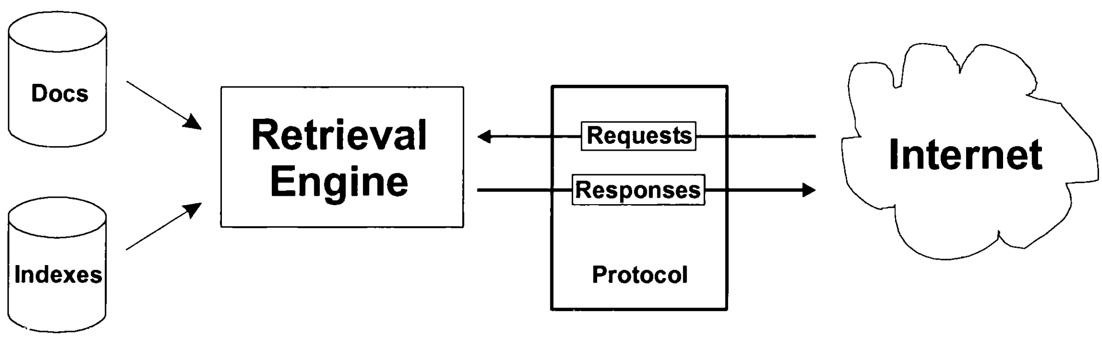


Fig. 6.3. Requests and Responses conforming to a protocol.

The protocol defines the interface between the IR Server and the outside world – it determines the services that the Server can and cannot provide. Section 6.3.1 describes how connections are made to the Engine and how requests and responses are exchanged.

The Retrieval Engine presents a number of objects at the interface, and the requests and responses relate to operations on those objects. Section 6.3.2 describes the functionality provided by the Engine through those objects.

6.3.1 Communicating with the Retrieval Engine

The Retrieval Engine does not handle incoming connections directly it only handles the retrieval-oriented processing once a connection has been established. The connection handling is managed by a Unix-style 'daemon'.

The daemon listens on a particular TCP port for incoming connection requests (Fig. 6.4a). Once it has accepted a connection (Fig. 6.4b), it 'spawns' a process containing the Retrieval Engine, and hands over the incoming call to it (Fig. 6.4c). The Retrieval engine then processes the retrieval requests from the client. The daemon, meanwhile, returns to listening for incoming connection requests and spawns an addition instantiation of the Retrieval Engine for each subsequent connection (Fig 6.4d).

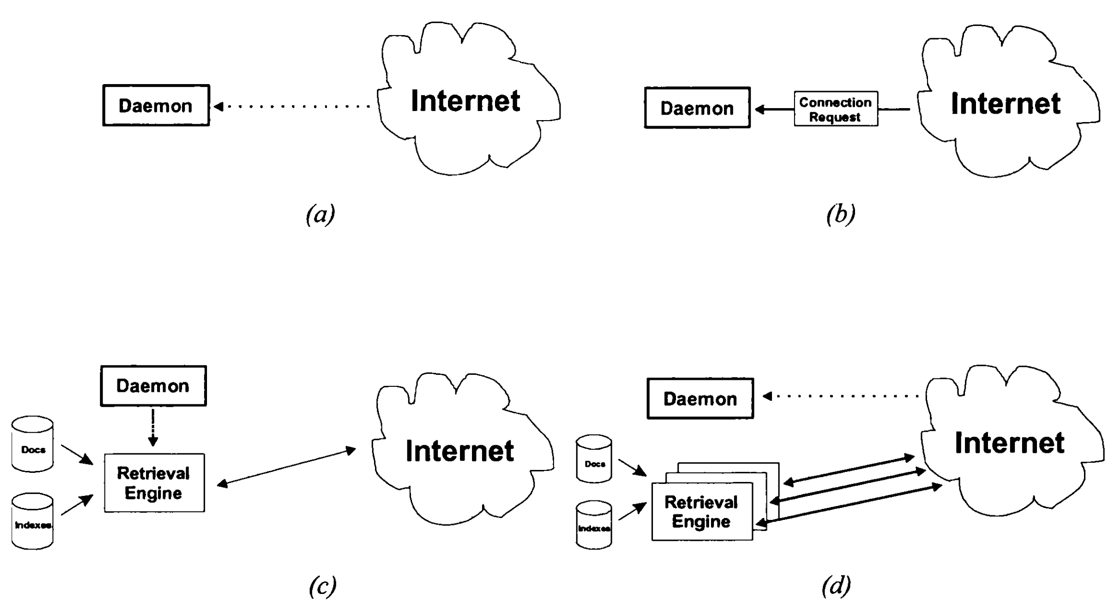


Fig 6.4. The Daemon spawning multiple instances of the Engine in response to connection requests.

In this way, multiple clients can each be communicating with their own instance of the Retrieval Engine. The number of simultaneous instances is limited only by the memory and processing resources of the machine on which they are run. Each of the instances can share access to a collection with any number of other instances, or can access a different collection. This spawning of multiple instances and any sharing of resources is transparent to the client.

Making a connection to the Retrieval Engine

There is a small library of ANSI C code that provides a client with all the facilities for both connection set-up and request/response transfer. Therefore, the implementer of the client need not concern himself with how the connections are made or how information is exchanged with the Retrieval Engine. The library provides four functions – connect, disconnect, send, and receive.

The 'connect' function requires two parameters: the IP address (or name) of the machine on which the Retrieval Engine sits, and the TCP Port address (or name) on which the service is provided (i.e. on which the daemon is listening). The 'send a request' function requires one parameter – a string containing the command. The 'receive a response' function returns one parameter – a string containing the response. The 'disconnect' function requires no parameters.

6.3.2 Retrieval Engine objects

The requests and responses relate to operations on objects that the Engine exposes through its interface. There are eight objects:

- **Server**

  This object is not directly related to retrieval, but represents the Retrieval Engine as a whole. It allows the Engine to be reset to its initial default state; for general information about it to be accessed (e.g. its version number and hence its capabilities); and for the Engine to be shutdown and the connection terminated.

- **Collection**

  The Engine can serve a number of collections. From the collection object, a list of available collections can be obtained, and information about each collection obtained (e.g. short textual description, number of documents, type of document, number of terms, stemming state, etc.).

  For retrieval, the collection object is instructed to load one of the available collections.

- **Query**

  The query object holds the terms and weights that will be used for searching. The object allows terms (with associated weights) to be added and removed from the query; the query to be reset to an empty state; and its current contents to be listed.

- **Filter**

  The filter object holds a list of Document Identifiers. All documents on this list will be ignored during retrieval.

  The filter object can be reset to its empty state, its contents listed, and one or more Document Identifiers added or removed.

- **Relevance**

  The Relevance object is used to store a list of documents that the Engine should regard as 'relevant' when performing a relevance-feedback search.

  Each Document Identifier that is supplied to this object must be accompanied by a 'degree of relevance'. The degrees are simply integer values that will be used during the relevance-feedback process to weight each of the relevant documents. For traditional relevance feedback, it is sufficient to make all such weights equal and non-zero. For relevance-feedback using partial relevance, those weights are set to indicate the relative relevance levels of the documents. The Ostensive model, for example, would set those values consistent with a particular Ostensive Relevance profile.

  The relevance object can be reset to its empty state, its contents listed, and one or more Document Identifiers added or removed.

- **Document**

  Once a collection has been loaded, each of its documents is available via this object. Documents are specified using a Document Identifier (i.e. a serial number ranging from *0* to *N-1*, where *N* is the number of documents in the collection).

  Using the Document Identifier, the following can be requested: the whole document contents; the 'header' of a document; or an arbitrary number of characters from the start of the document.

  The 'header' of a document is a collection-specific short summary of a document – intended for use in presenting lists of documents at a user-interface. Commonly, the header includes the title of the document along with date and authorship information. It may also include the first line of the document body-text. The header is restricted to four lines of text (each line can be up to 1000 characters long).

The ability to request an arbitrary number of characters from the start of a document is intended to allow a user-interface to use its own idea of a header/summary without the necessity of requesting the whole of a (potentially very large) document.

- **Stemmer**

  The Engine implements the Porter Stemming algorithm. This object allows arbitrary fragments of text to be converted into a list of stems. In addition, the Stemmer will accept a Document Identifier, and return a list of stems from that document (this removes the need for a client to first fetch a document and then send it back to the Engine in a stemming request).

  The list of stems that the object returns will have any duplicates removed, and will include a weight for each stem. The weights are calculated using the binary probabilistic model as described in Section 5.4, with the simplifying assumptions proposed by Croft & Harper in [Croft79] – i.e. they are default weights calculated in the absence of Relevance Information.

  If the particular collection that is currently loaded was not stemmed, then this object will return words instead of stems (again, with no duplicates, and with associated default weights).

- **Retriever**

  This object provides the core functionality of the Engine. It can be requested to search the currently loaded collection using either the Query object or the Relevance object as the information-need description.

  The Retriever expects two basic parameters – a maximum number of documents to be retrieved, and a switch indicating whether it should use the Filter object during the search.

  It returns to the client a list of Document Identifiers with associated scores. The Identifiers are of the top-scoring documents. The scores are *log(P(Rel))* – i.e. the base-ten-logarithm of the probability of Relevance for the document.

If a search is requested using the Relevance object, then the retriever expects three additional parameters: The maximum number of terms (of those generated by the analysis of the relevant-indicated documents) to use when performing the retrieval; a threshold (i.e. minimum) for the value of the relevance-weights to be included in the retrieval; and the amount of each document to be used when performing the analysis of the relevant-indicated documents.

A search using relevance-feedback returns not only the top-matching documents, but also returns the terms and weights that resulted from the analysis. That list of terms and weights is actually the new contents of the Query object, which was replaced during the analysis.

The 'maximum number of terms' parameter was provided as a result of the investigations by Harman [Harman92] who identified a variation in retrieval effectiveness related to the number of terms in a query. Her empirical studies suggested that using as many terms as were available was not the optimum approach and that a small fixed number appeared to be superior (approximately 20 terms). Recognising that this may well be a collection or application specific effect, the Retrieval Engine allows this parameter to be set.

The 'weight threshold' parameter was originally added for experimental purposes to see if there was a similar relationship to be found with minimum weights – but not taken further. For normal use, this threshold is simply set to zero – i.e. only terms with positive relevance-weights are used for retrieval.

The 'sample size' parameter was implemented to allow the analysis of the relevant-indicated documents to be restricted to the first *x* characters of each document. As above, this was to allow experimentation. More pragmatically, it was included to permit a speed-up of the relevance-feedback retrieval. By restricting the analysis to only the first one or two thousand characters rather than the whole document (often five to ten times that size), the analysis can run much faster and can still able to get a useful snapshot of the document content.

When speed is not paramount, and accurate application of the techniques is required, this parameter can be set to zero and the whole document will be used.

Designed for a slow connection

At first sight, the operation of the above set of objects may appear a little clumsy. This is a hangover from when the construction of the server was begun (circa 1989/1990). At that time, Internet connections were slow and therefore it was necessary to minimise the transfer of data between a client and a server. The server was intended to support real-time interactive retrieval – therefore, anything that could reduce the retrieval time perceived by a user was advantageous.

The result of this is particularly apparent in existence of the Query, the Filter, and the Relevance objects.

The Query object is there because sending a large query each time a search was requested was time-consuming and noticeably slow for the user. It was noticed that each successive query made by users often shared all but one or perhaps two terms. The Engine holding state in this way allowed only the changes to the previous query to be sent prior to a new search request being sent. With subsequent increases in Internet bandwidths, this is no longer necessary, and commonly the query object is simply reset, filled with the new query, and then a retrieval request sent.

The Filter object was motivated by the fact that some user-interfaces did not want to redisplay documents already seen by the user. Not to use a filter would mean the unnecessary transfer of unwanted documents in the results of a search. Transferring a list of the Document Identifiers of unwanted documents along with each search request involved a noticeable delay. The list would usually grow throughout a session, and that growth was generally simple additions to the list. Therefore, the Filter object was created to hold the list of unwanted documents, and only the 'deltas' sent over the communications link. Although no longer strictly necessary, it is still used by some clients to save the effort of filtering out unwanted documents.

The Relevance object had similar motivations to those of the Filter object – although the amount of information-transfer that it saved was small. Its existence was motivated more by consistency in the Engine's interface than communication efficiency.

The ability of the Stemmer object to apply the stemming algorithm to a collection document and pass only the resulting list of stems to the client, is an additional and less-used example of minimising communication.

Now that Internet connections are much faster, the above three objects are no longer necessary. It would seem more appropriate to include the information contained in each of the objects, in individual search requests instead. Nevertheless, there is no significant overhead in the manner of operation that is common with recently built clients – i.e. simply reset each object, supply them with new contents, and then request a search.

6.4 Using the Retrieval Engine

A query-based search

To perform a simple query-based search, a client would (after connecting to the Retrieval Engine) perform the following steps:

- **Load the desired collection**

  In the simplest case, the client would request the Collection object to load a known collection.

  More commonly, the client would request a list of available collections from the Collection object, present that list to the user, and then request that the user's selection is loaded.

- **Stem the user's query**

  The client would send the user's query to the Engine's Stemmer object, and in return, receive a list of the terms and their default weights. The client may choose to show that list of terms and weights to the user. Further, the client may also allow him to edit the weights, delete terms, or perhaps to add new terms – perhaps to allow the user to experiment with different weights. New terms would imply additional stemming requests to the Engine.

- **Perform a search**

  The client would first enter the terms and weights into the Query object, and then request that the Retriever object perform a query-based search, and indicate the maximum number of documents it wished to be returned.

  The client would receive a list of Document Identifiers with scores in return.

- **Fetch document headers**

  The client would request that the Document object provide headers for the documents. It would then present those headers to the user for browsing.

- **Fetch full document texts**

  The client would make requests to the Document object for the full text of individual documents, as required by the user.

To perform subsequent searches, the client would: Request stems and weights for the user's new queries, or additions to queries; Reset the Query object (i.e. delete the current contents) before entering the new query to be used for retrieval; Request a retrieval.

As the session progresses, the client might choose to use the Filter Object to prevent certain documents from being retrieved again.

The current collection could be replaced with another without disconnection – the client would simply request the Collection object load the new collection.

A relevance-feedback search

This presupposes that query-based retrievals have taken place and documents have been found that the user regards as relevant. The client would reset the Relevance Object, register the Document Identifiers of those documents that the user regards as relevant, and then request that the Retriever object perform a relevance-feedback search.

Along with the list of top-matching documents, the client would receive the query that was used for the relevance-feedback search – the client might choose to present that to the user.

A 'similar document' search

To find similar documents to an individual document (e.g. for a client that presents a traditional browsing interface), the client would follow the same procedure as above, but with only that single relevant document being registered in the Relevance object.

An Ostensive-relevance / partial-relevance search

The process is the same as that for a traditional relevance-feedback search, but instead of supplying identical weights with the Document Identifiers when they are registered in the Relevance object, weights that reflect the partial or Ostensive Relevance would be supplied.

In a path-based ostensive environment, it is likely that the client will not want to show the current document or the most recent documents in the list of available next- steps. Therefore, the Filter object is likely to be used to avoid those documents.

6.5 Client interfaces that use the Engine

A number of clients use the Retrieval Engine – they offer either components of a user-interface, or a full interactive searching environment.

- **A stream-based client**

  This is a Unix command-line program that makes a connection to the Retrieval Engine and then accepts requests on the Unix 'standard input' stream and passes them onto Engine. The responses from the Engine are outputted on the Unix 'standard output' stream.

  This is the most basic of the clients and is essentially an alternative to the ANSI-C communications library – allowing non-C programs (such as shell scripts and web-server cgi-scripts) to use the Engine by connecting to it by piping and redirection of input and output streams.

  It was originally conceived as a development and debugging interface for the Engine as it *relies* only upon the most basic facilities of the Engine, although it can *offer* the most complex ones.

- **A command-line client**

  This is the simplest 'real' user-interface. It allows a user to interact with a textual interface that allows collections to be loaded, queries to be entered, results to be viewed, documents to be viewed, and relevance-feedback to be performed. It hides from the user all the details that the Retrieval Engine and its objects and protocols impose – replacing them with a text command language that, although basic, is easier to use.

  Figure 6.5 shows the user typing a '?' for a list of commands; issuing the 'collection' command to select a collection (in this case the "ft85" collection – articles from the Financial Times newspaper of 1985); issuing the 'lookup' command with a query of "nuclear waste dumping"; being shown that the stems of that query are 'nuclear', 'wast', and 'dump' and being shown the titles and scores of the top 10 documents; viewing the raw contents of document '5'; issuing a relevance-feedback search using documents 1, 5, and 19; then finally, quitting.

```
Ver 15.01
Calling flotta...
Connection established.

Ready> ?

Collections     List and load collections.
Lookup          Lookup something in the collection.
Headers         List the headers of retrieved documents.
nn              Show retrieved document.
Similar nn nn   Search for similar documents.
Quit            Quit from system.

Ready> collections
Available collections are:
bbc2
european
ft85
herald
iht
indie
ldoce
le_monde
paris
unixman

Collection to load> ft85
#OK#

Ready> lookup nuclear waste dumping

Query is: nuclear wast dump

Top 40 Documents retrieved.

 1. (100) Controversy Likely Over Dumping Of Nuclear Waste / London
 2. (100) Radiation Risk At Vickers Dismissed / Warship yard rejects
 3. (100) Nakasone Starts Pacific Tour (364)
 4. (100) Separate nuclear facilities urged / Statement by Paul
 5. (100) Decision Denied On Nuclear Tip / Government denies 'secret
 6. (100) Billingham Rejected As Nuclear Waste Dump (388)
 7. (100) Parliament and Politics: More Tests Ordered For Potential
 8. (100) Mahathir Faces Political Difficulties: Malaysia's leader is
 9. ( 71) Construction Contracts / Fairclough Civil Engineering (210)
10. ( 71) Technology: Computer System Aids Search For Underground

Ready> 5
..HL.-
# 09:03:85 Decision Denied On Nuclear Tip / Government denies 'secret
decision' has been made on choice of site for dumping nuclear waste (111)
..BL.-
   By IVOR OWEN
..TX.-
Mr William Waldegrave, a junior Environment minister, assured the Commons
yesterday that the Government had taken no 'secret decision' on the choice
of a site for disposing of nuclear waste.
..TX.-
He was replying to a debate in which Bedfordshire MPs underlined the concern
aroused by the earlier announcement that a site in their county at Elstow is
one of three to be investigated for the shallow burial of short-lived
intermediate and low level waste.
..DS.-

The Financial Times
..XP.-
London Page 4 #SECTIONB
*****************************************************************

Ready> similar 5 1 19
Top 40 Documents retrieved.

 1. ( 67) Radiation Risk At Vickers Dismissed / Warship yard rejects
 2. ( 23) Decision Denied On Nuclear Tip / Government denies 'secret
 3. ( 16) Controversy Likely Over Dumping Of Nuclear Waste / London
 4. ( 14) Resources Review: Hazards of dismantling a gas-cooled
 5. ( 14) Billingham Rejected As Nuclear Waste Dump (388)
 6. ( 14) Parliament and Politics: More Tests Ordered For Potential
 7. ( 13) State-Owned Company For Nuclear Waste / Government plans
 8. ( 10) Separate nuclear facilities urged / Statement by Paul
 9. ( 10) Management: Tunnelling Through A Mass Of Complexity / The
10. (  9) Letter to the Editor: Display Units And Radiation (312)

Ready> quit

Connection closed.
```

Fig 6.5. A short record of a dialogue with the command-line client.

- **"News Retrieval Tool (NRT)"**

  NRT is a windowed interface for searching documents (primarily newspaper articles) developed by Sanderson and van Rijsbergen [Sanderson91]. The interface has a window for the query, one for the result of each retrieval that the user requests, and one for each document that the user inspects in full (Fig 6.6).

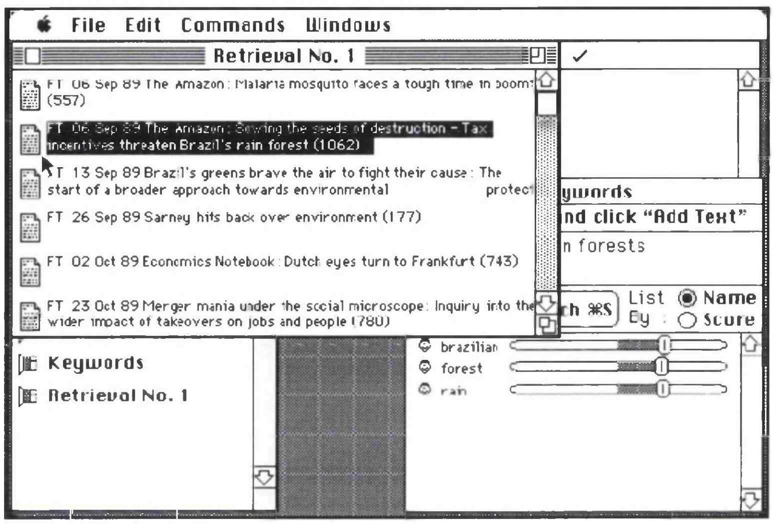


Fig 6.6. A screenshot of NRT, showing the query window and a result window.

The query window contains list of terms and their weights, with the weights presented as sliders. The user can add and delete terms to the query, and modify the weights of terms at will.

Later versions of the query window also contain an area into which the user can drag documents from any of the result windows. Once the user has collected document he regards are relevant, he can request a search.

That manner of relevance-feedback provision was novel at the time. Soon after NRT was first demonstrated publicly, the same metaphor was to be seen in a new version of the widely known WAIS interface [Stein91].

NRT ran on an Apple Macintosh, and came in two versions – a client-only version that connected to the Retrieval Engine using the standard TCP/IP method, and a standalone version that used the embedded version of the Engine.

The client version of NRT was used for a number of years by Efthimis Efthimiadis and his students as part of a course on Information Retrieval at the University of Berkley in California; connecting to a Retrieval Engine at the University of Glasgow.

- **Investigation of interactive query expansion**

  Magennis built a bespoke interface to connect to the Retrieval Engine for his experiments into effective strategies for interactive query expansion [Magennis97].

- **Evaluation of simulated work tasks**

  The Retrieval Engine provided the retrieval services to the bespoke interface used by Borlund & Ingwersen [Borlund99] in their interactive experiments of the efficacy of using *simulated* work tasks in the evaluation of IR Systems.

- **The Ostensive Browsing Interface**

  The Ostensive Browsing Interface is the most recent client to use the Retrieval Engine. It is the subject of Chapter 7.

6.6 Summary

The IR Server provides a binary probabilistic searching service. It can be contacted either locally on the same machine or remotely over an Internet connection. It can offer several collections to several clients simultaneously and transparently.

In addition to the standard query-based searching service, the Server offers Relevance Feedback services, and the Binary Probabilistic implementation of the Ostensive Model.

The IR Server has grown from a simple test-bed for experimentation into a fast, flexible, and reliable service that, decoupled from the advances made over the years in user-interfaces, has remained a useful service – e.g. providing the retrieval services for research work in addition to that of this thesis.

7 An ostensive media-neutral browser

The Ostensive Browser was built to demonstrate, to experiment with, and to evaluate the Ostensive approach. It provides a searching environment that is query-free[1], representation-free, and media-neutral. The interface is the tangible result of the ideas of Chapters 1 to 5. This chapter describes the 'look and feel' of the interface through its features and the issues around them.

Contents of this chapter

Section 7.1 describes the system architecture and the flexibility that it brings. Section 7.2 describes the visual and interactive features of the interface. Section 7.3 describes some of the behaviour of the interface in use. Section 7.4 describes the additional features provided in the interface to support interactive evaluation.

[1] Except for some restrictions on the starting point in the prototype. Non-query operation is, however, possible – see "Starting a session" at the end of Section 7.2.

7.1 Architecture

Implementation environment

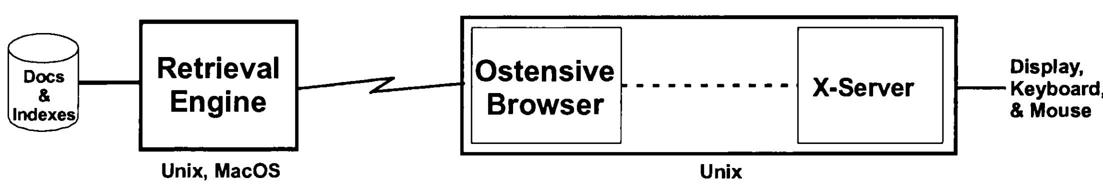
The interface was written in the *ANSI C* programming language for the *Solaris* (i.e. Unix) operating system. The standard C libraries were used to interface with the Unix file-system. The communications library of the IR Server provided the interface to the Retrieval Engine. The graphical environment was *X-Windows*: with the control panel and settings panel components built using the *X-View* toolkit (an X version of the old *SunView* toolkit); the graphics rendered using the low-level *X-lib* functions; and the thumbnail image display achieved using the *PBM-Plus* toolkit.

With the exception of those libraries, the interface constitutes over six thousand lines of code.

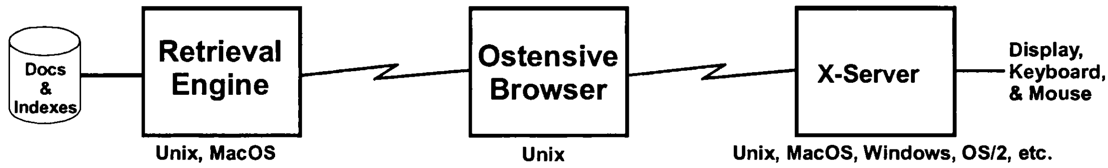
X-Windows was thought to be the approach most likely to provide an interface that would support the fast and compute-intensive interactive environment of this interface, and still allow display on a wide range of machines and operating systems. At the time of development (around 1995), the only other cross-platform alternative, the *Java* programming language, was still relatively early in its own development and not nearly fast enough for this project.

Platform flexibility

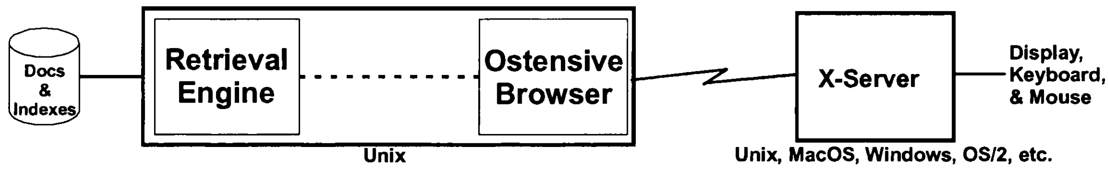
X-Windows allows the interface code to run on one machine (in this case a Solaris Unix machine), with the display possible on a separate machine running only an off-the-shelf X-Server system (available for Unix, MacOS, Windows, OS/2, etc.). This gives a number of possible arrangements of the Retrieval Engine, the Ostensive Browser, and the X-Server (Fig 7.1).



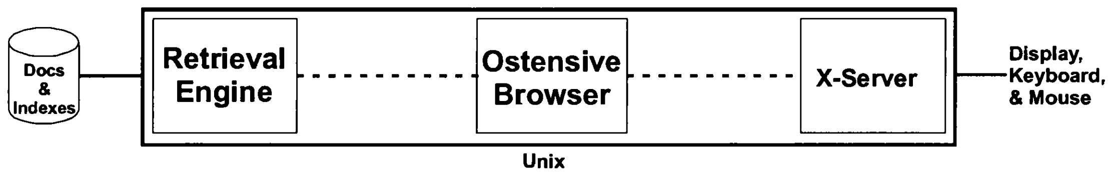
(a) Engine on one machine, Interface on another.



(b) Each component on a separate machine



(c) Everything on one machine except the display.



(d) Everything on one, standalone, machine.

Fig 7.1. Arrangements of the system components, and their platform dependencies.

Figure 7.1a shows the arrangement that has been most commonly used – and used for the experiments of Chapter 9. A Unix machine, hosting both the Browser and the display, connects to a machine hosting the Retrieval Engine.

If a user wishes to interact with the system from a non-Unix machine, then the arrangement of Fig 7.1b (and less likely, Fig 7.1c) can be used.

Finally, Fig7.1d shows all components running on a single machine – for example, a laptop without network connections.

All such arrangements of the components are invisible to the user – he sees only the graphical user-interface.

7.2 Interface elements

Overview

The interface presents two windows to a user: a *Browse* window and an *Object-viewer* window (Fig 7.2). Most interaction takes place in the Browse window – that is where searches are initiated, paths are displayed, the browsing space is investigated, and new objects are encountered. The Object-viewer window is to show the full contents of any selected object.

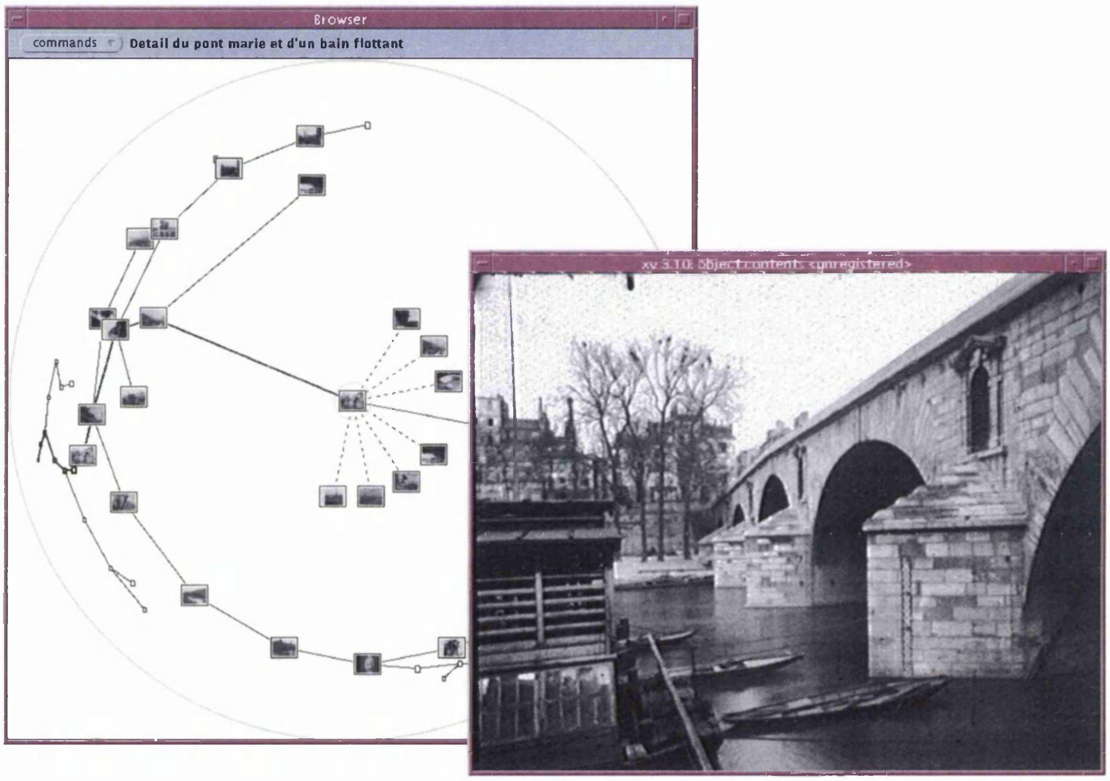


Fig 7.2. The two windows of the Browser interface

There are two additional windows in the interface that are occasionally seen: the Settings window, and the Annotation Window.

The Settings Window

The Settings Window provides control over various aspects of the appearance and behaviour of the interface (Fig 7.3). The individual controls will be described where appropriate below.

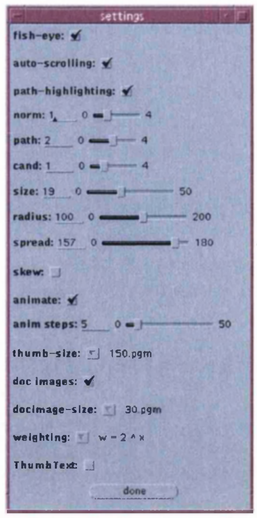


Fig 7.3. The settings window

The Annotation Window

The Annotation Window displays the fragment of text that represented an object during indexing (Fig 7.4). The annotation is there only because the retrieval behind the prototype system is text-based. If 'real' image (or other non-text medium) retrieval were behind the interface, this window would not be there. In the case of the 'Paris' image collection, used for the evaluation in Chapters 8 & 9, the annotation is the short fragment of text associated with each image.

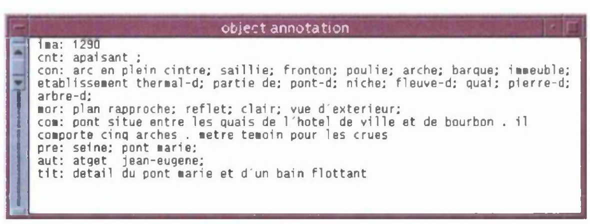


Fig 7.4. The Annotation window

The annotation window is not for presentation to a normal searching user – it is included for demonstration and debugging purposes only. If such texts were considered part of the object, then it would be displayed to the user along with the image/sound/video by the Object Viewer.

The Commands Menu

The menu provides access to functions not normally, or not frequently, used – i.e. not accessed in the middle of a searching session (Fig 7.5).

Fig 7.5. The commands menu

Amongst other things, it allows the user to: provide a new path starting-point from a text query, or from an existing object; bring up the Settings Window, or the Annotation Window; and quit the program. The other options will be described along with the features with which they are associated.

The Object Viewer

This window displays the full content of an object – be that text, image, sound, video, or any combination thereof. As this function is highly data-dependant, and is likely to require sophisticated bespoke viewers, it is not handled directly by the Browser. The Browser passes the data that constitutes an object onto an appropriate external application.

For example, in the case of the JPEG images of the Paris collection, the viewer used is the commonly available *xv* image application for Unix and X (Fig 7.6).

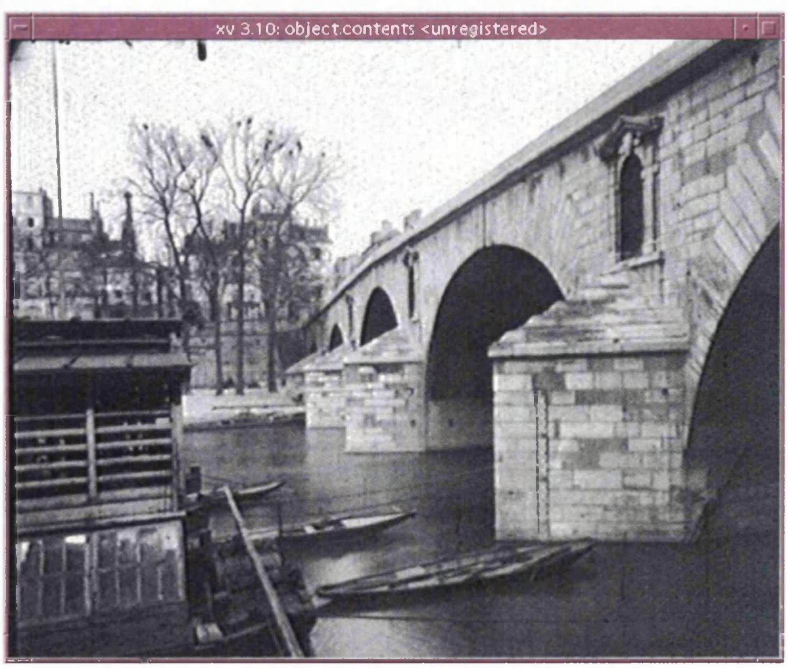


Fig 7.6. The Object viewer window, showing an image from the Paris collection

As the viewer is an external application, and as it might not be as reliable as the Browser, it is possible for that application to crash whilst the Browser continues to run normally. Therefore, the Command Menu includes an item that will launch a new copy of that external viewer.

Objects and links

Each object of the browse surface is represented either by a generic document icon (Fig 7.7), or by a miniature thumbnail of its contents (Fig 7.8). If the starting point was a query, then that is shown as a small circle. Links are shown as lines connecting the objects, and show the path on which they lie.

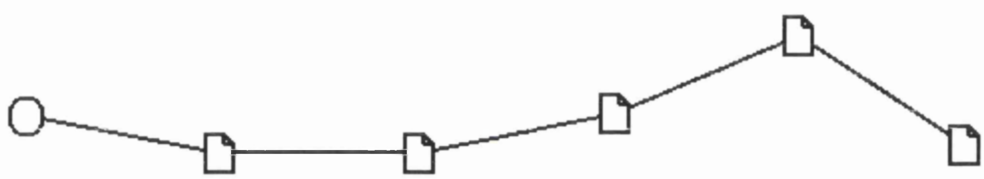


Fig 7.7. A query starting-point, and objects shown as icons

The thumbnail is a small graphic intended to provide "at a glance" information regarding the contents or the nature of an object. In the case of an image object, this would be most likely a miniature version of the image; for other types of content, then the thumbnail might graphically indicate some other attribute of the object – e.g. the author, the date, the time, the size, the source, the recency, etc.

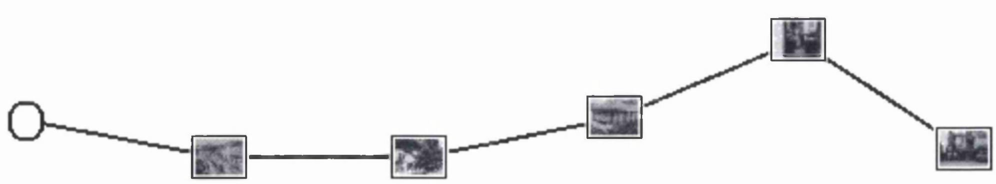


Fig 7.8. A query starting-point, and objects shown as thumbnails

Control over the presentation of objects as icons or as thumbnail images is provided in the Settings Window (Fig 7.9).

Fig 7.9. The switch for showing/hiding object thumbnail images

The screen-shots of the Browser used in this chapter are a mix of both icon and thumbnail views.

Current-Object and Current-Path

There is a *Current Object*, which is the object whose contents are currently shown in the Object Viewer window. The Current Object is highlighted with a grey circle around it, and its 'next-steps' shown around it. Any object can be made the current object by clicking on it. The 'path' to the Current Object (i.e. the sequence of objects from the starting-point that define the available next-steps) is highlighted by default (Fig 7.10).

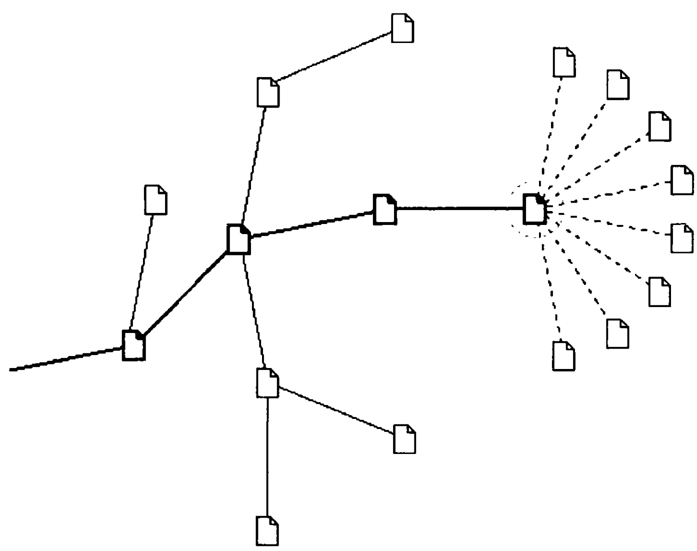


Fig 7.10. The Current Object, its next-steps, and the highlighted path to it.

Links to next-steps are shown as dotted lines – to indicate that they have not yet been followed. Only the Current Object has its next-step shown – this is to reduce clutter on the display. Only those objects that have been visited by the user (i.e. made to be the Current Object at some point) are shown permanently on the display. Current Objects are not restricted to only those at the end of a path – any object, at any point in a path, can be clicked-on to make it the Current Object and to show its next-steps (Fig 7.11).

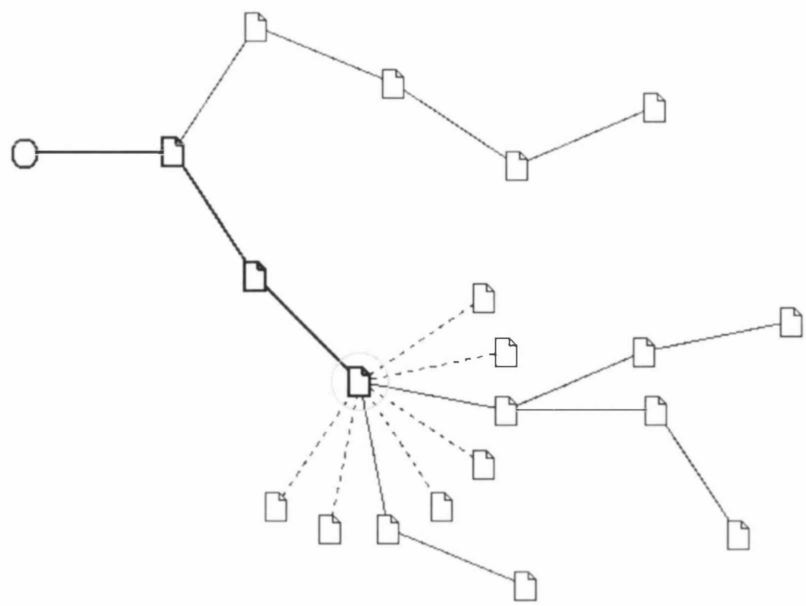


Fig 7.11. The Current Object in the middle of a path

In such cases, next-steps that have been explored have normal links shown between them and the Current Object.

The relative line-weights of normal links, links on the path to the Current Object, and links to the next-steps can be modified on the Settings Window (Fig 7.12).

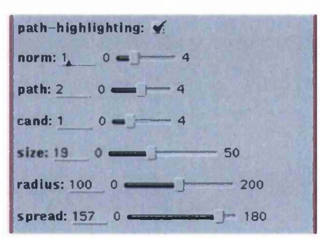


Fig 7.12. Settings for the paths, icons, and next-steps

The other, related, settings shown in Fig 7.12 are those for the size of the object icons, the length of the links (marked as 'Radius' in the diagram), and the angle over which the candidate objects are spread.

Rollover thumbnails

Whenever the pointer moves over one of the object thumbnails, a larger thumbnail pops up (Fig 7.13).

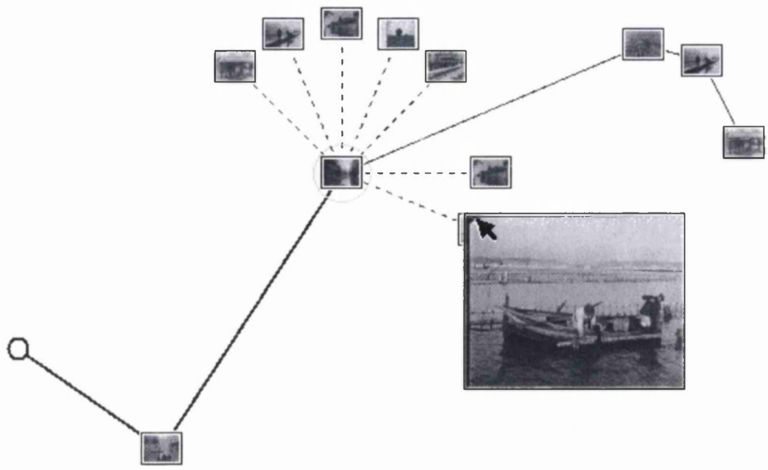


Fig 7.13. A rollover thumbnail (the pointer can be seen in its top left)

It remains visible only for as long as the pointer is over the object. The motivation behind it is to provide a greater amount of information about an object than is possible in the (necessarily) small thumbnails on the browse surface. These rollover thumbnails avoid the additional clicks and the delays of rendering the full object-content in the Object Viewer.

The interface is also able to show text information along with the image – for example, the Paris collection has titles for each image (Fig 7.14). The settings window has a switch to show or hide such thumbnail texts.

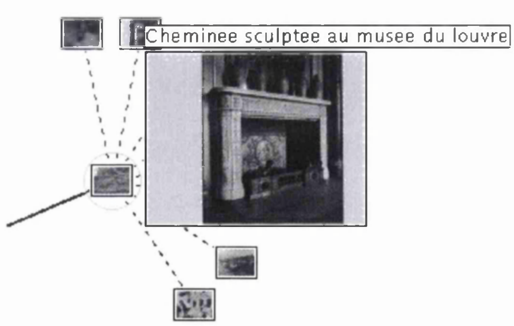


Fig 7.14. A rollover thumbnail consisting of an image its title

Combined with the object-thumbnails and the Object Viewer, the rollover thumbnails provide a range of combinations of information access versus effort and time:

- Basic information can be seen on the object itself without any action on the part of the user.

- For the small additional effort of rolling the pointer over an object, the user can see more information. He can do this quickly to any object on the screen by flicking from object to object with the pointer.

- Only when the full contents are required, need the user click on the object, and wait for the external viewer to render the contents. This might become particularly time-consuming with large complex multi-media objects.

Starting a session

As the work of this thesis does not consider starting points for ostensive browsing, but with the browsing and information-need development, the Browser includes a mechanism to start from a textual query. The Command Menu has an option for starting from a query, which brings up a small window into which the user can enter text (Fig 7.15).

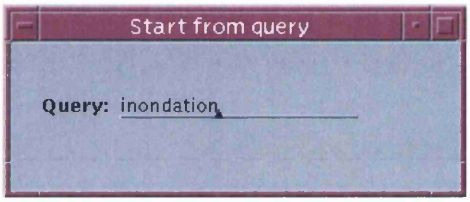


Fig 7.15 The query-entry window

The Command Menu also has an item that allows a new path to be started using the Current Object from an existing path as the new starting point.

7.3 The Fish-eye projection

A fish-eye projection ([Furnas86], [Lamping94] and [Schaffer96]) is a view transformation that effectively 'warps' what is being viewed to achieve two, apparently contradictory, goals: 1) to show the focus of current attention in close-up detail; 2) to show the whole of what is being viewed within the restricted size of a window. This 'Focus and Context' is achieved by showing the part that is the current focus in a manner that is near to normal, and by shrinking all other parts around the focus to an extent proportional to their distance from the focus. The overall visual effect is that the focus is large and everything else gets smaller and smaller towards a horizon around the edge of the display.

The details of the fish-eye projection used in the Browser, and issues around it, will be described in the course of describing the two problems that it was employed to solve – i.e. scrolling and collisions:

7.3.1 Problems with scrolling

As a path grows, it inevitably reaches the edge of the Browse Window. At that point, the user must drag the path (using the middle mouse button) away from the edge in question. Nevertheless, this problem will, all to soon, occur again. This procedural load is not core to the searching task, therefore efforts were made to reduce/remove it.

One alternative to the path dragging would be to offer traditional scroll-bars on the window. These would minimally transfer the procedural effort rather than reduce it. In fact, it could be argued that by forcing the user to move his pointer away from the object with which he is interacting, locate a (commonly small) scroll button elsewhere on the window, scroll an appropriate amount, and then move his pointer back to the object of interest, would involve a greater amount of interference and inconvenience to the searching task. On those grounds, scrollbars were rejected.

Auto-scrolling

A partial solution to the interference and inconvenience associated with dragging/scrolling is offered by 'auto-scrolling'. Auto-scrolling will automatically scroll the Current Object to the centre of the window. This means that the Current Object will always be shown in the centre, and the user can concentrate his actions there during a path exploration. Auto-scrolling can be switched on and off on the Settings window, as can the speed of the animation of the scroll (Fig 7.16).

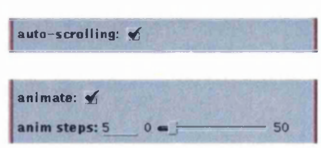


Fig 7.16 Settings for auto-scrolling

Although auto-scrolling solves the scrolling problems associated with path growth, it retains procedural effort when moving around the whole of a long path, or of a large tree of paths. Central to the problems of movement around large path-trees is that of visibility – it is not possible for the user to see the whole tree, or to access directly objects that are not currently shown in the Browse Window. As a result, the user must click on several intermediate objects to progressively scroll the view to the desired object – this problem grows with the size of the tree (Fig 7.17).

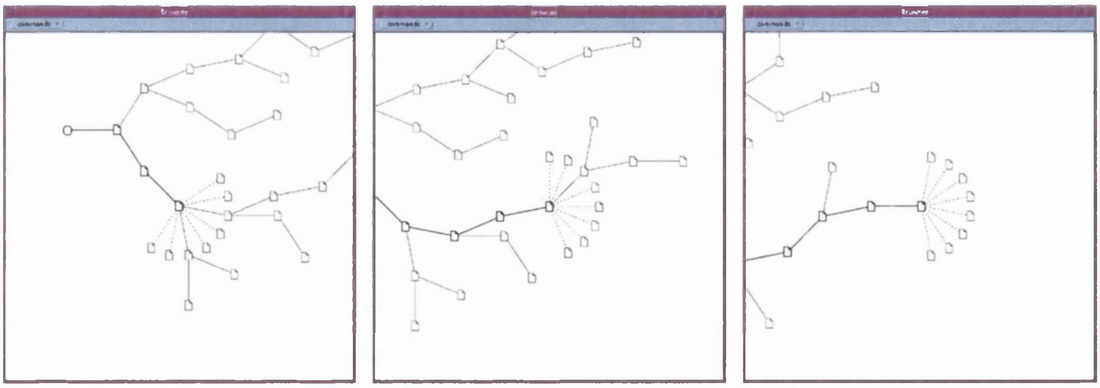


Fig 7.17. Auto-scrolling keeps the Current Object in the centre, but large scrolls remain difficult

Everything is visible with a fish-eye view

Applying the fish-eye transformation to the view of a path tree allows the whole path tree to be seen in varying degrees of miniaturisation, but with the current object full-sized (Fig 7.18).

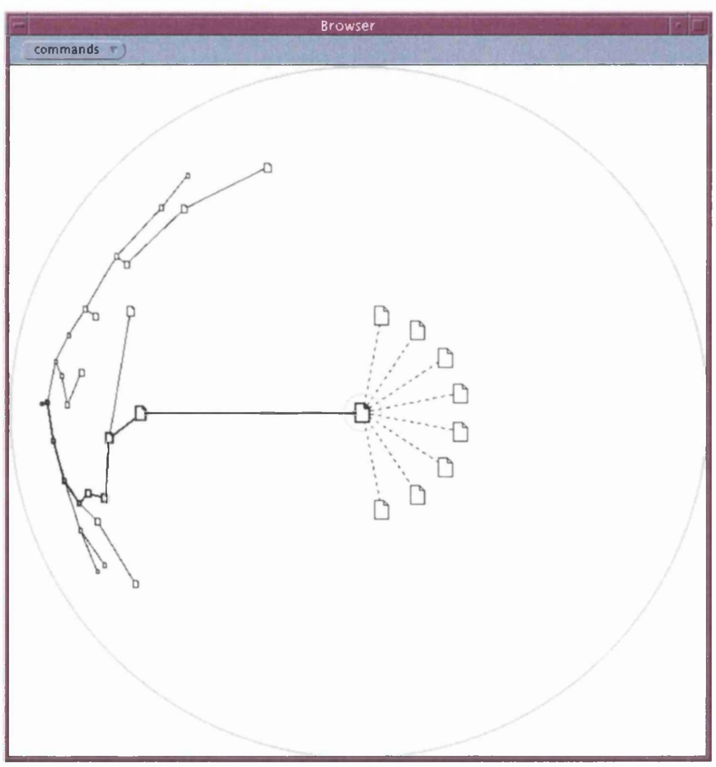


Fig 7.18 A fish-eye view of a path tree.

The objects in Fig 7.18 can be seen to get smaller as they get further away from the focus.

With the fish-eye view, the user is able to see all objects and interact with them directly without the need for scrolling. For example, he can move the pointer over any of the objects and get the rollover thumbnail pop-up. Similarly, the user can click on any object and it will auto-scroll to the centre of the window, and become the Current Object. A (sadly poor) presentation of this dynamism is attempted in Fig 7.19, where the fish-eye view has been applied to the three successive Current Objects views of Fig 7.17.

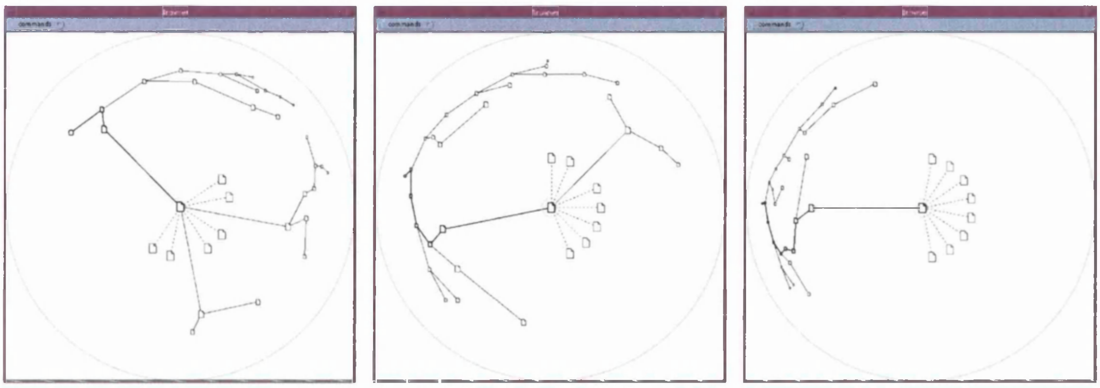


Fig 7.19. A fish-eye view of the same objects and paths of Fig 7.16

Costs of the fish-eye view

The fish-eye transformation results, by definition, in a distortion of the layout of the path tree. This can be seen in the three views of Fig 7.19, where the structure (in parts, and as a whole) is *recognisable* across the three views, but it is not *identical.* This may present problems of orientation for some users – empirical investigation is needed.

As objects become further from the focus, they get smaller. With particularly large path trees, this can lead to objects becoming so small that it becomes difficult to accurately place the pointer over them. This, in turn, makes it difficult to get a rollover thumbnail, or to click on the object to make it the Current Object. One solution, ironically, is the same technique as was necessary with an auto-scrolling flat view – i.e. multiple scrolling steps. Nonetheless, this problem is limited to very large path trees and only to those objects furthest from the Current Object – therefore, it an be argued that the problem (if it really is a problem) is a small one.

There is an issue with the current version of the Browser's fish-eye view – object *thumbnails* do not shrink as they become distant from the focus. This is simply a result of the cost of the time and effort required to implement an image-scaling algorithm that would be fast enough to allow the smooth animated scrolling and re-drawing of the display. Instead, the current version, switches off an object's thumbnail when its distance from the focus reaches a threshold (Fig 7.20).

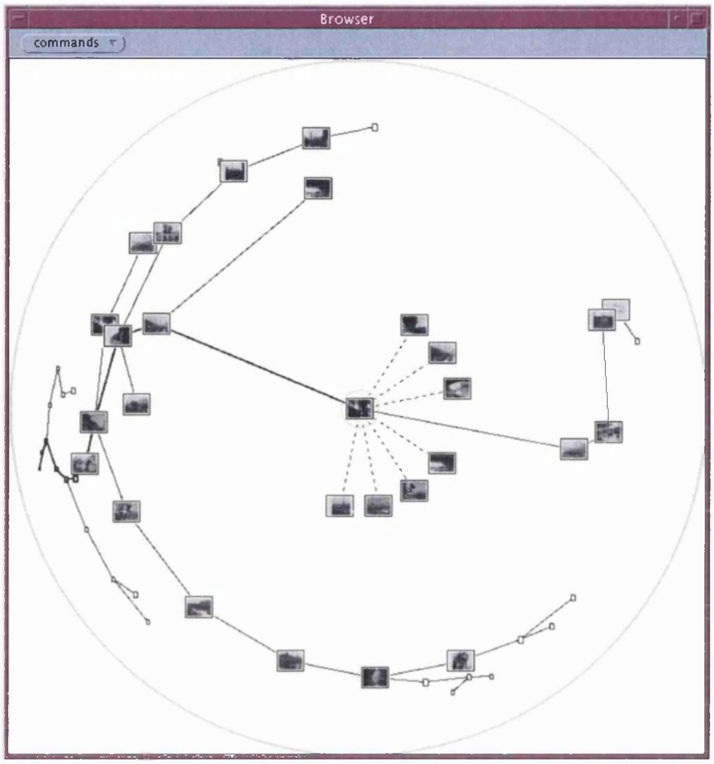


Fig 7.20 Object thumbnails are switched-off if they are over a certain distance from the focus

The thumbnails are switched-off to prevent the clutter of them overlapping that would result from their increasing mutual proximity as they become closer to the edges of the display.

7.3.2 Problems with collisions

As the presentation of the splitting paths is that of a tree, widening from left to right, 'collisions' between different branches of the path tree are likely. Such collisions would be difficult to avoid without strong, inconsistent, and localised distortion of the view.

In an effort to reduce the effect of these unavoidable collisions, the following modification to the classic fish-eye view was implemented: Instead of applying the view transformation to all objects, it is *not* applied to unvisited next-steps. The result is that visited next-steps are pushed out well beyond the radius of the unvisited next-steps. This can be seen back in Fig 7.19 & 7.20. The motivation is that visited objects are likely to have links to other objects, and are thus likely to cause collisions with unvisited next-steps. This does not prevent all collisions, but it does prevent them with the Current Object and next-steps, and it pushes other collisions outside of the central focus area. Fig 7.21 shows a degenerate case of collisions, in a flat and in a fish-eye view.

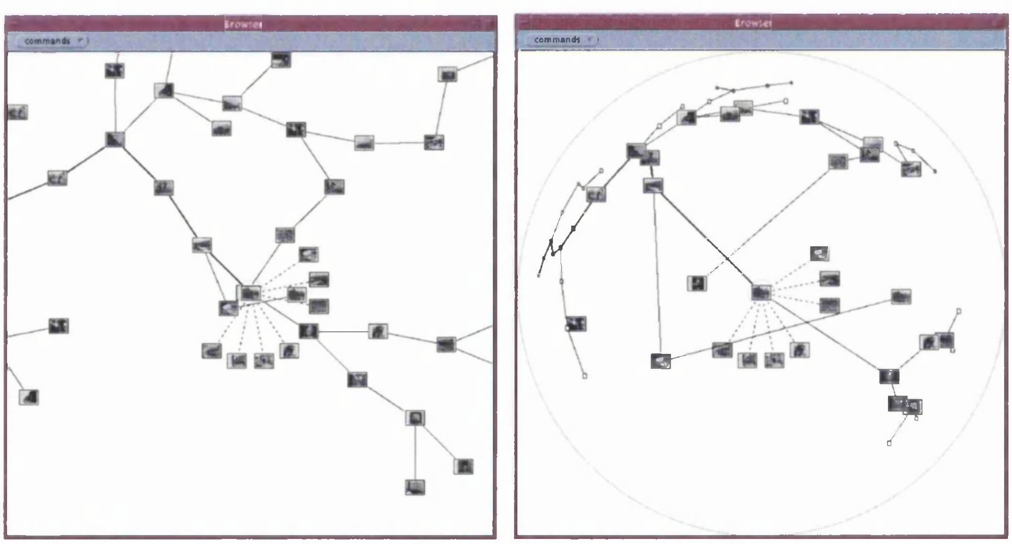


Fig 7.21 A flat view and a fish-eye view of the same collision area

In the flat view, there are two objects sitting over the links to the next-steps. Worse: there is an object from another path occluding the Current Object. Worse still: that occluding object and its path predecessor happen to be situated such that they appear to form an additional (i.e. ninth) next-step.

By contrast, in the fish-eye view, none of these non-focus objects is sitting on any links or objects – they are pushed away from the focus, with only a stray link traversing the focus area. A minor refinement to the Browser could render the portion of that link that crosses the focus area in a greyed-out colour – thus further reducing its impact.

I believe that the combination of fish-eye view and selective flattening offers a superior display, and that Fig 7.21 demonstrates that superiority – both in the visibility of the whole path tree, and in the avoidance of collisions in the important focus area.

7.3.3 The form of the transformation

Of the many possible ways to effect the desired 'focus and context' transformation, there are two classes commonly considered: rectangular and circular.

Rectangular transformations

With this class of transformation, the rectangular coordinates of a point (with respect to the focus) are used to define its position, and each coordinate component (i.e. x and y) is transformed separately, using the same transformation function. One of the results of this approach is that the whole of the rectangular window can be used to present the objects. Informal testing with a rectangular transformation in the Browser indicated that it immediately appeared unnatural – and remained so. This was particularly the case when objects were in motion (e.g. during the animation of auto-scrolling). Therefore, it was rejected.

Circular transformations

With this class, the point's position is defined in polar coordinates (i.e. Radius R and Angle θ), with only R being transformed. This resulted in a view that was much more pleasing in the informal tests. It presents a suggestion of a hemi-spherical display surface. The informal results agreed with the reports in [Lamping94], and this class was adopted.

Circular horizon

The choice of the circular transformation approach means that the points on the fish-eye display that corresponded to points on the flat surface whose distance neared infinity, forms a circle around the focus. In the Browser, the idea of the circular horizon was reinforced by a large grey circle showing its location (e.g. Fig 7.20).

Logarithmic transformation function

The function used to transform the radii in the Browser fish-eye view was a logarithm. The *general* shape of the function is shown in Fig 7.22 along with lines showing a variety of points being transformed from the flat surface to the fish-eye surface.

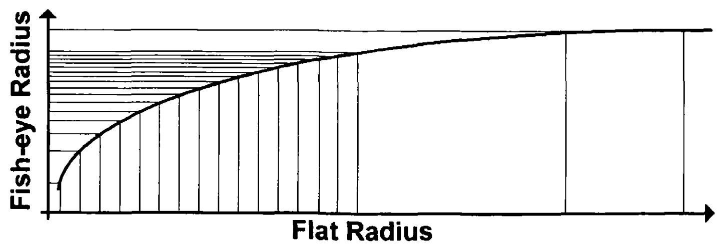


Fig 7.22 The logarithmic fish-eye transformation function, with some indicative points displayed

Three things to be noted from Fig 7.22:

- As the flat distance from the focus increases linearly (in the diagram, the lines near the origin are linearly spaced), the fish-eye radius increases by less and less;

- As the original radius becomes large, even large changes result in an effectively identical fish-eye radius – hence the circular vanishing point horizon when used with a circular transformation.

- Near the focus, the fish-eye transform actually magnifies the distance between points – this is part of the reason that the fish-eye view avoids collisions in that area.

The inverse of the transformation function is used to scale the individual object icon sizes. As objects get further away from the focus their size reduces – tending to a minimum that will still allow the user to see, rollover, and click it.

7.4 Extensions to support evaluation

To support the evaluation of the Ostensive Model several additional features were implemented in the Browser. The desire was for an environment where: multiple users could be set multiple tasks; the Ostensive Relevance profiles could be changed for each task (without the user being aware); users could indicate the relevance of objects that they encountered during their performance of a task; and users' path explorations and relevance indications could be logged for later analysis.

As the demands of a normal searching session are different to that of an experimental subject carrying out an evaluation task, the Browser can be started in one of two modes – one mode for normal information seeking whose appearance is as described in Section 7.2 & 7.3; and an evaluation mode that will now be described.

The evaluation-mode Browse window is different only in the controls that appear at the top of the window. There are now three menus: a User menu, a Query menu (initially disabled), and the familiar Commands menu (Fig 7.23).

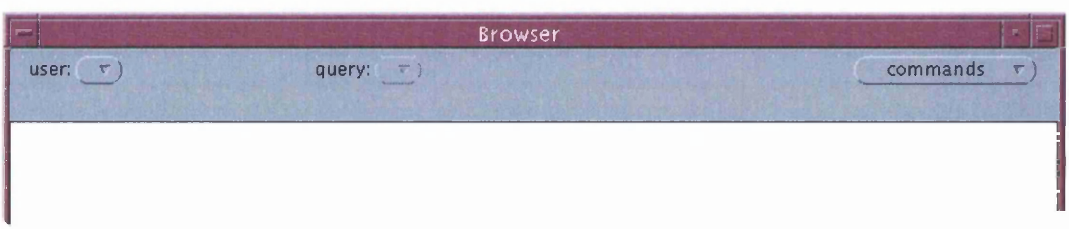


Fig 7.23 The evaluation-mode menu bar

When first encountering the system, a user would select their name from the list of evaluation subjects in the User menu (Fig 7.24).

Fig 7.24 The User menu listing the names of the evaluation subjects

Upon selecting their name from the User menu, their name would be displayed and the Query menu would become enabled (Fig 7.25).

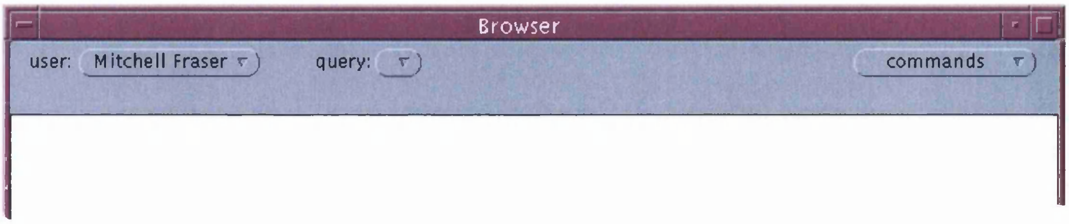


Fig 7.25 The selected user's name is displayed and the Query menu is now enabled

The user would then select the particular query that they have been instructed to perform, from the list presented in the Query menu (Fig 7.26).

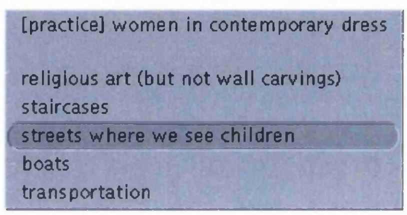


Fig 7.26 A query menu for a particular user, showing one practice and five real queries

That list of queries is specific to each user. It also contains a practice query to allow, for example, the user to learn to use the Browser and to get a basic feel for the collection in which they will be searching.

Once the user has selected a query, that query is displayed on the browse window next to their name, and the pre-determined starting point for that query is presented in the Browser (Fig 7.27). The user would then browse as normal from the given starting point.

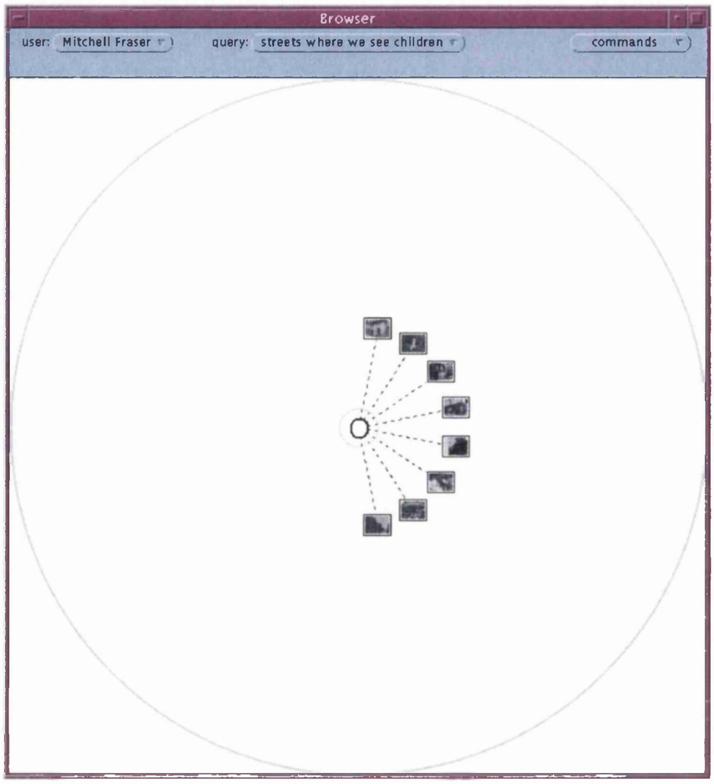


Fig 7.27 The current query is displayed, and its predefined starting point is presented

At any time, the user can indicated that an object is relevant to the task/query by pressing the right mouse-button. This toggles the relevance indication for that objects. In the current version of the Browser, marked objects are shown with a (rather primitive) black dot on their thumbnail or icon (Fig 7.28).

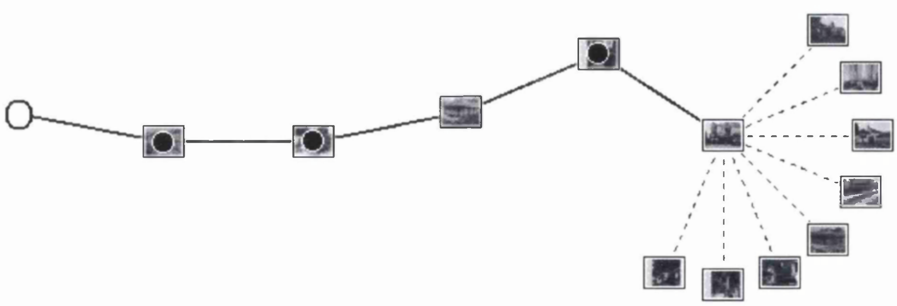


Fig 7.28 Three objects marked as relevant (indicated by a black dot)

Once a session is completed (either by user-decision, or by time-elapse), the user terminates the query/task by selecting the "End Query" item in the Commands Menu. In the Browser's evaluation-mode, the Commands Menu has only the end-query, the quit, and (just in case the external viewer application misbehaves) the launch-new-viewer items (Fig 7.29).

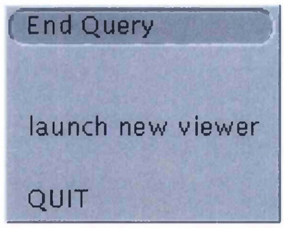


Fig 7.29 The Commands Menu in the evaluation mode

Once a task/query has been ended, the Browser will save the information that it has logged during the task.

Logging of user actions

Currently, the logging that takes place is basic, and is as follows: each object that is selected as a Current Object by the user is logged in the order that the selection was made. Along with that selection-record is the object's position in the path tree. That is, the complete path tree can be rebuilt afterwards from the logs, and done so in the same order that the user explored. Any objects that are marked by the user as relevant are recorded.

The Browser writes out, for each user, and for each query/task, a file containing the record of path-growth, and a file containing the list of marked-relevant objects.

Input files and profile selections

To set up an evaluation, the experimenter must supply three files: a file containing the names of all the evaluation users for display in the User Menu; a file containing all the query names (for display in the Query Menu) and their associated pre-defined query-text (i.e. the text that will be used to form the starting point for that query); and a file containing the association of queries, users, and Ostensive Relevance profile (i.e. which users are to perform which queries, in which order, and which profile is to be used for each of those).

A variety of Ostensive Relevance profiles is implemented within the Browser. When the Browser is in normal-mode, any one of these profiles can be selected by the user in the settings window. When the browser is in evaluation-mode, their selection is hidden and driven by the evaluation files.

The above extensions for evaluation were developed for the evaluation that will be detailed in Chapters 8 & 9.

7.5 Summary

This chapter presented the actual prototype interface that was motivated and designed in the previous chapters. First, the system architecture was described, highlighting the flexible, cross-platform nature of the interface. Then the interface components and how it worked in practice were described – that description was necessarily limited as the interface is highly dynamic and its presentation here is restricted to static images. Finally, extensions were described that allow the interface to be used for interactive evaluations of Ostensive Relevance profiles with all query setting, profile selection, and logging operations hidden from the user.

The drive towards a reduction of procedural load was demonstrated by: the minimal nature of the controls necessary to operate the system; object thumbnails and the larger rollover thumbnails that offer varying amounts of information in return for varying amounts of effort and time; auto scrolling reducing the need for scrolling; the fish-eye view reducing that scrolling effort even further, combining it with the ability to see the whole display surface at one time, regardless of its size, and also reducing the number of visual collisions on the display surface around the current object.

The use of Ostension made possible the crucial reduction in procedural load – i.e. the absence of queries and their management. It also made possible a further reduction in procedural load – i.e. the absence of explicit relevance indications and their management. The removal of querying brings with it the hiding of all internal representations.

The hiding of the internals of the underlying IR system make the Ostensive Browsing prototype a truly representation-independent, and media-independent searching environment.

Part IV:

The Evaluation

The chapters of this part are developments of the work published in [Campbell00].

In this Part, I present the evaluation of the relative effectiveness of different uncertainty discount functions. I describe the construction and characteristics of a new image test collection utilising multiple binary relevance-assessments. I discuss the use of such assessments and multiple interpretations of them. The evaluation environment is detailed in terms of the interface, test collection, and tasks set to users. Multiple interpretations of the results and the statistical significance of comparisons are presented. Shortcomings of the approach are discussed with suggestions for improvement. The results obtained in the evaluation are consistent with the proposals of the Ostensive Model – reinforcing the predicted evidence profile.

8 A multi-assessment image test collection

This chapter describes the construction of a new image test collection. The basis was a collection of 666 images with text annotations. For that collection, thirty queries of varying types were generated. For each image in the collection, four binary assessments of its relevance to each of the queries were obtained. The process used to transform the collection into a test-collection is extensible – allowing the four binary assessments to be extended later.

Motivation

The collection was built to allow for the evaluation of the Ostensive Model. That evaluation required a collection of non-text objects that were of general interest and that had textual annotations describing their content. Non-text objects were specified in order to make more obvious the lack of user interaction with, and the lack of visibility of, anything that might constitute an internal representation. The requirement for general interest contents was to make easier the job of obtaining suitable evaluation users, reduce the difference in their domain knowledge, and finally to make easier informal observations of all aspects of searching using the collection. Text annotations were required to allow existing text-retrieval technologies to be applied 'behind the scenes'.

The absence of such a test-collection on the above grounds would have been sufficient to motivate its construction, but it was also desired that it had multiple assessments – i.e. capturing some of the subjectivity of relevance. This was not necessary for an evaluation of the Ostensive Model, but it was regarded as a useful resource to construct for this and later experiments. As with the Ostensive Model itself, the use of multiple assessments is in keeping with the principles of Polyrepresentation [Ingwersen94 & Ingwersen96].

Contents of this chapter

Section 8.1 describes the documents that make up the 'Paris' image collection. Section 8.2 describes how the queries were generated. Section 8.3 details how the relevance assessments were obtained for the queries. Section 8.4 presents some issues relating to subjectivity and error in the assessment of relevance that were encountered during the process.

8.1 The documents of the collection

The 'Paris' collection, as it has become known, was obtained from the French Ministry of Culture as part of the FERMI project – an ESPRIT multimedia research initiative [Fermi94]. The image collection, as used here, consists of 666 annotated black & white photographs taken mostly in and around Paris, around 1900. Each image has a textual annotation consisting of both free and controlled vocabulary. An associated thesaurus provides a guide to the intention of the controlled vocabulary.

Images

Each document has a 640x512-pixel greyscale image (although there are about 12 colour photographs amongst them). The subject matter is varied and includes: Portraits (some of famous people); Buildings, bridges, gardens, monuments; Interiors of buildings; Street scenes; Coastlines; Public events; Close-ups of architectural details; Historical and decorative artefacts. Examples are shown in Fig 8.8.

Fig 8.8 Example images from the 'Paris' collection.

Annotations

The annotations consist of seven text fields. The contents of the fields are either free-text or controlled vocabulary. The language used is exclusively French. All field contents are in upper case. There is an accompanying guide to the controlled vocabulary.

The fields are as follows:

Code	Name	Description
IMA	*Image No.*	The serial number of the image.
TIT	*Title*	The title of the image.
CNT	*Connotation*	Feelings conjured up by the image – e.g. love, hate, serenity.
CON	*Content*	Important objects present in the image – e.g. a man, a horse, a house, a fountain.
MOR	*Morphology*	Perspectives or positioning of the view – e.g. elevation, 3/4, plan, exterior, interior, reflection.
COM	*Comment*	Extra information beyond the title – sometimes explanatory.
PRE	*Precision*	Proper nouns, dates, events – e.g. place names, people's names, building names.
AUT	*Author*	The photographer, or agency from whom/which the image originated.

Apart from the IMA and TIT fields, none of the fields is mandatory. The annotations were built by professional image cataloguers at the Ministry. There is no information as to which particular cataloguer is responsible for which annotations, how many cataloguers were involved, or what techniques (if any) were used to maximise consistency and minimise labelling errors. Therefore, one can only assume that the information in each annotation represents a single individual's view of the attributes of an image, at the time of indexing.

An example annotation:

Field	Data
IMA:	4567
TIT:	GARCONS DE CAFE OFFICIANT SUR DES BARQUES SOUS L'OEIL VIGILANT ET DISCRET DE L'ARMEE
CNT:	INSOLITE;
CON:	DEBIT DE BOISSON; BARQUE; PERSONNAGE; GARCON DE CAFE; RUE-D; PHOTOGRAPHIE EVENEMENTIELLE; INONDATION; PIED; MILITAIRE; REVERBERE; DEVANTURE; SCENE DE RUE;
MOR:	PLAN MOYEN;
COM:	INONDATIONS DE 1910 A PARIS ; LE PERSONNEL DU " CAFE DE LA CHAMBRE " TRAVAILLANT EN BARQUE DANS LES RUES INONDEES
PRE:	CAFE DE LA CHAMBRE; PLACE DU PALAIS BOURBON;
AUT:	SEEBERGER ATELIER;

Thesaurus

The accompanying guide provides thesaurus-like information relating to the controlled vocabulary. It has roughly 6,300 entries. Each entry contains up to six fields as follows:

Code	Name	Description
NOM	*Name*	The term.
DO	*Domain*	The theme or domain of this term.
TG	*Generic*	Broader, more generic terms.
TA	*Associated*	Semantically similar or synonymous terms.
TS	*Specific*	More specific terms.
EP	*Replace*	Terms to be avoided and replaced by this term.
EM	*Preferred*	Terms preferred over this term (e.g. because of ambiguity).

An example thesaurus entry:

Field	Data
NOM:	DECOR D'ARCHITECTURE
DO:	ARCHITECTURE
TG:	ARCHITECTURE COMMUNE
TS:	BALUSTRADE; AMORTISSEMENT; ALCOVE; VERRIERE; BOISERIE; COURONNEMENT ARCHITECTURAL; PLAFOND; CARRELAGE; CHEMINEE; PARQUET; PAN DE BOIS; BALCON; ORDRE ARCHITECTURAL; MOULURE; DECOR D'ARCHITECTURE-D
TA:	PEINTURE MURALE; ORNEMENTATION; PAPIER PEINT

One obvious application of the thesaurus would be to expand terms in annotations by adding the more specific terms in an effort to improve recall.

8.2 Generating the queries

The members of the FERMI project group (twenty-five people, spread over four sites) were asked to suggest queries. They all had access to the collection, and were asked to provide examples in three broad classes:

1. Objects – e.g. a man, a tree, a horse, the Eiffel Tower
2. Arrangements of objects – e.g. objects to the left of, in front of, below other objects.
3. Implicit abstractions – e.g. work, transportation, death

From the submissions, a selection was made of those queries that offered a variety of subjects, and of nature. That is, they were chosen to be sufficiently different from one another and to retain examples of the three classes of query. All queries were generated with general knowledge of the contents of the collection. Thirty queries were produced:

1. Military
2. Women in contemporary dress
3. Dining halls / restaurants
4. Death
5. A person to the right of a table.
6. Horses, a crowd, and stairs ...where the horses are to the left of the stairs.
7. Person to the right of a fountain.
8. Streets where we see children.
9. Door to the right of stairs
10. Religious art (but not wall carvings)
11. Buildings near water
12. Examples of 'early modern' architecture
13. Staircases
14. People at work
15. Buildings where the roof can be seen clearly
16. Transportation

17. Reflection
18. Interiors
19. Street scenes
20. Animals (real and icons)
21. Seated people
22. Men with beards (i.e. not just a moustache)
23. Views from an elevated position
24. Floods
25. Seaside/beach
26. People wearing hats
27. Words visible (in any language)
28. Boats
29. Commerce
30. Parkland/countryside

8.3 Generating the relevance assessments

Eighteen people (again members of the FERMI project group) declared a willingness to participate as relevance assessors. Thirty queries were to be assessed against 666 images – i.e. roughly 20,000 individual image-query assessments. There is clearly a limit to the number of individual assessments that a single person can be expected make. In addition, our assessors were volunteers – so the amount of inconvenience and quality assessment time we could expect from them was limited.

Designing the assessment environment

The full collection of 666 images is a lot for someone to go through for each query. In addition, that number of images can become unwieldy to handle even with automated presentation tools (e.g. a 'slide show' set-up). A number of informal tests were performed with different automated viewers. A page of thumbnails appeared to be the most effective – minimising the amount of button pressing and waiting for image loading that was required to view and (re-view) images. It was then thought it might be possible to reduce the general levels of procedural load by using printed thumbnails.

A further set of informal tests showed that printed A4 pages of fifteen thumbnails (i.e. five rows of three) were being searched significantly faster (between two and four times faster, for the whole collection) than an on-screen equivalent. One additional advantage was that the assessment exercise could be carried out wherever the assessor found most convenient – i.e. they did not need a computer to hand. This was confirmed by several assessors who reported that they needed to find an environment away from their desks and offices free of distraction where they felt relaxed and able to concentrate.

There is an obvious criticism of the use of thumbnails – i.e. that the full detail of the image is not available. In an attempt to lessen that problem, a web page was made available where assessors could view the full size version of any image they felt necessary – although, in the event, this was little used

Given a limited number of query-image assessments per assessor, there was a choice between asking an assessor to compare a large number of queries against a subset of the collection, or a small number of queries against the whole collection. The perceived advantages of the first approach was that, if the subset were small enough, the assessor might become familiar (to an extent) with the images, and therefore, better able to assess their relevance to queries. For example, an assessor might be less likely to miss relevant images. The perceived advantage of the second approach was that all the assessments for any individual query would have been made by the same assessor – giving a degree of consistency.

It was planned to have more than one set of assessments performed for each query. Therefore, the advantage of the subset approach appeared limited, and the effect of assessors missing images in the second approach would be lessened. The second approach (full collection per assessor) was adopted.

The informal tests indicated that the time taken to perform a binary relevance assessment for all images with respect to one query varied between as little as four minutes to up to twenty minutes. The quickest queries were the class of queries requiring only the presence of objects. The time-consuming queries belonged, roughly, to the 'object arrangement' and 'abstraction' classes.

The assessment exercise

The thirty queries were distributed three times across the eighteen assessors. This resulted in each assessor being given five different queries, and each query being assessed by three different assessors. Each assessor was given the image collection as a pack of 45 pages of 15 thumbnails. Each thumbnail had its title and image identifier underneath. The titles were included on the suggestion that it might help to clear up cases of ambiguity. On reflection, any ambiguity to which an assessor might be subject may well also apply to a searching end user. In addition, relevance is inherently subjective. Therefore, in retrospect, it might have been better to allow this to happen and not to have included the titles. The image identifiers were present to allow the assessors to list those images that were relevant to each query. In addition, as explained above, they allowed the assessors to obtain full-size images where necessary.

The assessors were told three things:

1. To identify those images that, *in their opinion*, were relevant to each query.
2. Not to discuss their assessments with anyone else.
3. The process should take between four and twenty minutes per query.

One can see that the relevance assessments would be binary in nature, and made solely by the assessor. The time indication was given to show that the whole process should take less than an hour and a half. This was an effort to reassure the assessors in the hope that they would not be tempted to rush through as fast as possible, potentially making more errors. In fact, conversations with the assessors after the exercise had been completed indicated that they had taken slightly longer than this. They said that they had been keen to give accurate answers and therefore, spent whatever time they felt necessary. This is in contrast to the informal timing tests where it could be argued that the same 'weight' or 'seriousness' was not present in the task.

The results of this exercise came in much faster than expected, and the reports were that the task was not nearly as onerous as originally thought. In the meantime, additional people had come forward, willing to be assessors. Therefore, a second phase was performed where the thirty queries were distributed across five assessors – i.e. each new assessor was given six different queries, and each query was assessed once.

The relevance assessments

The result of the assessment exercise was thirty queries, each having been binary-relevance assessed against the 666 images by four different people. For each query there is a list of tuples, where each tuple has the image identifier of a relevant image, and the number of assessors who regarded that image as relevant.

In general, the number of images relevant to a query is inversely proportional to the number of assessors who agreed on its relevance. Fig 8.9 shows the minimum, average, and maximum number of relevant-assessed images per query across the test collection.

	1 Assessor	*2 Assessors*	*3 Assessors*	*4 Assessors*	*All Assessor-counts*
Minimum	2	1	0	0	5
Average	26	13	11	9	59
Maximum	84	52	47	27	167

Fig 8.9 Number of relevant-assessed images per query.

Re-expressed as a percentage of the total number of relevant-assessed images per query, they give an indication of the degree of 'overlap' [Lesk69] or agreement between the assessors (Fig 8.10):

	1 Assessor	*2 Assessors*	*3 Assessors*	*4 Assessors*	*Total*
Minimum	15	5	0	0	
Average	44	20	19	17	100%
Maximum	71	32	35	36	

Fig 8.10 Number of relevant-assessed images per query.

These figures are lower than those of [Borlund97] which were, however, for *retrieved* documents – i.e. not all documents. The figures are slightly higher than those of [Ellis94] for a hypertext-linking task – a related, but not identical task. The variability reported there is echoed here. The figures are lower than those of [Voorhees98] for a directly comparable task. In general, it might be said that the overlaps here are low – and it has been suggested that it is a result of the image medium instead of the traditional text.

The range of the number of images with a particular level of assessor-agreement decreases correspondingly. This is confirmed in histograms showing the distributions of the number of the thirty queries that have a particular number of images marked by one, two, three, or four assessors (Fig 8.11).

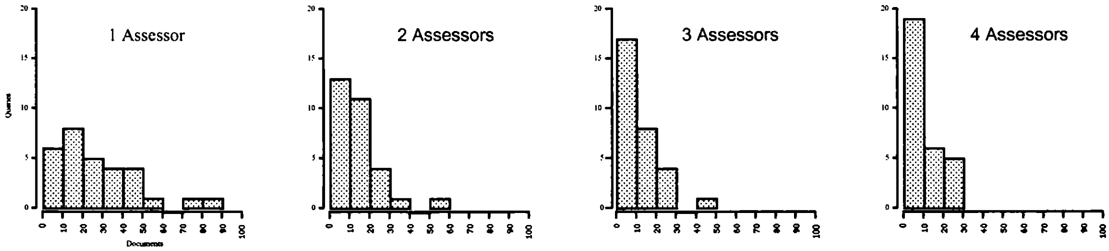


Fig 8.11 Number of queries versus number of relevant-assessed images.

8.4 Subjectivity versus error

The multiple assessments are retained in their original form – hopefully retaining any subjectivity. This allows them to be combined in whichever manner is thought to be most appropriate for a given experimental or evaluative situation.

The thin line between 'error' and 'subjectivity' is perhaps demonstrated by the images of Fig 8.12. All four assessors for query number 24 "Floods" marked the first two images relevant, whereas only one of them marked the third one relevant.

4 assessors 4 assessors 1 assessor

Fig 8.12 Side-by-side images of "Floods".

The image on the left that could easily be just a rainy day in a derelict street rather than a flood scene, but it was judged as a flood by all four assessors. The image on the right has the same elements as the middle image (i.e. men in boats, and water right into the doorways of buildings), but only one of the four judged it as a flood – was is mistaken as a photograph of a canal in Venice? This is particularly mystifying as the three images were presented side-by-side, forming the top row of a page of thumbnails.

For the same query, only one of the assessors marked the image in Fig 8.13 as relevant.

Fig 8.13 An unpopular "Flood" image.

It is an image of people fishing on the banks of the Seine. They are standing on the quay/roadway that is halfway up between the normal water level and the ground/street level. The level of the river often rises to, and covers, that quay/roadway without the river being said to have actually "burst its banks". Nevertheless, we can clearly see that something is partially covered in water from the river. Although clearly a subjective issue, this example also calls into question the adequacy of thumbnails as a basis for relevance judgement. A larger image would make the partially covered cobbles in the foreground more clearly visible, along with the ramp at the far right.

Multiple assessments resulted in at least one assessor judging the images as relevant, and thus, at worst, promoting the images from definitely not relevant to potentially relevant. If ten or more assessments had been made per query-image pair, then the difference in the assessor-count because of error or subjectivity may have been less. Nonetheless, that alone could not 'fix' the 'problem' – an appropriate interpretation of the assessor-count, that captures our intuitions about such problems, is also required.

Such issues motivate a requirement to perform not only multiple assessments, but as many of them as possible, and to develop an appropriate manner of interpreting the multiplicity of assessments. Such multiplicity and interpretation of assessments can be argued as necessary if a test collection is to be regarded as an acceptable standard against which to evaluate a retrieval system. After all, it seems reasonable that a test collection should, in some way, capture the inherent distribution of interpretations of relevance in a target community.

The assessments are extensible. Using the same thumbnail-pad technique, the number of unique assessments for each query can be easily increased. In fact, this is currently underway, and will continue on a piecemeal basis. This will improve the degree to which the subjectivity is captured, and therefore increase confidence in relevance conclusions drawn from them.

8.5 Summary

The 'Paris Collection' was described – an image collection of 666 photographs with textual annotations of content, origin, and connotation. Thirty varied queries were constructed for the images in the collection. An inexpensive (in terms of time and perceived effort) process of capturing multiple binary relevance assessments was described. This produced a test-collection of 666 images, and thirty queries, with each image and query pair having four binary relevance-assessments made on them.

The multiple assessments provide a manner of assessor fault tolerance, and capture a degree of the subjectivity associated with such assessments. The current four-assessor assessment is extensible, allowing additional levels of assessment to be incorporated – thus both improving the fault-tolerance and increasing the subtly of the subjectivity captured.

The fine distinction between error and subjectivity was highlighted, using examples from the collected assessments.

9 Evaluating Ostensive Relevance Profiles

This chapter presents an interactive evaluation of three different Ostensive Relevance Profiles using the test-collection of Chapter 8. In doing so, it uses multiple interpretations of the multiple binary relevance assessments. It is as much an investigation of the experimental method as it is of the profiles under evaluation.

Motivation

The Ostensive Model has been developed both informally and formally, it has been instantiated in a novel searching environment. The basic efficacy of the Ostensive Relevance Profiles, have not been established, neither has the precise form that they should take.

Contents of this chapter

Section 9.1 presents the profiles that were evaluated. Section 9.2 presents the setting in which the evaluation was carried out. Section 9.3 presents an interpretation and combination of the test collection's multiple assessments. Section 9.4 uses the interpretations to analyse the data captured during the evaluation. Section 9.5 discuses some strengths and weaknesses of the evaluation method, proposing improvements.

9.1 The subject of the evaluation

Of the many novel aspects of the Ostensive approach, the Ostensive Relevance Profiles stand out as being a central component, and one that requires first empirical confirmation, then refinement. For the evaluation presented here, basic confirmation of utility is the goal.

To establish the true utility of the Ostensive Approach, it is likely that there will be series of evaluations performed on a number of profiles, starting conditions, media, and combinations thereof. Establishing a reliable and effective evaluation *method* is as important as the profiles.

For this confirmatory evaluation, three Ostensive Relevance profiles were used. Note that, in contrast to the use of uncertainty profiles of Section 4.5, the presentation of the profiles here is of Ostensive Relevance – an arguably more intuitive sense. Nevertheless, the ideas behind them are still those presented in Section 4.5.

The "document in context" profile (DIC)

The Ostensive Model suggests a profile where the Ostensive Relevance of a relevant-indicated document decreases with age. One such profile distinguishes itself: a Fibonacci series where the Ostensive Relevance of any object is equal to the sum of that of all of its predecessors. This has the general form of Fig 9.1.

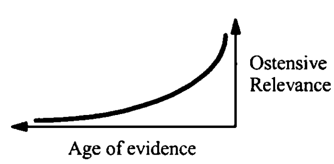


Fig 9.1 The "document in context" profile

The intuition is that with such a profile, the importance given to a document (when building a representation of the current information need) is equal to that given to its 'context'. The particular profile described here can be regarded as a turning point in the relationship between document and context. A flatter curve would give more importance to the context; a steeper curve, more importance to the document.

The "context biased" profile (CB)

If one were to flatten the curve completely, it would correspond to the model inherent in traditional relevance feedback (Fig 9.3).

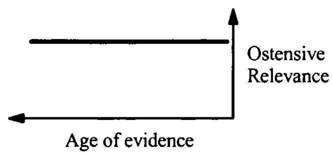


Fig 9.3 The "context biased" profile

The "context only" profile (CO)

This class of profile is expected to be the least effective. It has been included for completeness, to rule it out.

The true form of a context-only profile would have a step function with only the first document having any connection with the information need – all other indications being regarded as purely random with respect to the information need. This profile can be relaxed slightly into an increasing-with-age profile, which makes it less of a 'straw man' in this evaluation (Fig 9.4).

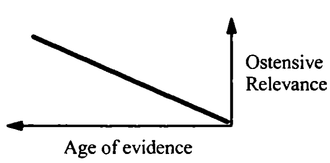


Fig 9.4 The "context only" profile

Summary

The objective was to compare the preferred balanced profile *DIC* with the flat *CB* profile associated with traditional Relevance Feedback. This was to be done in an environment of image retrieval using the graphical interface of Chapter 7. It was minimally hoped to determine that the preferred profile *DIC* was at least as good as the *CB*. As a form of lower bound, and to rule out a whole range of such profiles, the 'context only' profile *CO* was also evaluated.

9.2 The experimental set-up

Indexing the documents of the Paris collection

As the details of retrieval were not the primary concern of this evaluation, the thesaurus was not used and the annotations were simply flattened to text. That is, the fields were concatenated together, no special meanings were attached to each field, and the controlled-vocabulary terms were treated as normal words. The flattened annotations were used as surrogates to index the images in a traditional binary probabilistic manner, usi9ng the system presented in Chapter 6. Informal experimentation first confirmed that 'reasonable' retrieval was taking place, with relevance feedback finding images regarded as having similar title, content, connotation, morphology, etc.

By way of contrast: A more sophisticated indexing of the collection has been carried out by Chiaramella and Mechkour [Chiaramella97] (and [Ounis98a]) – also as part of the FERMI project. It provides a highly structured spatial and structural representation of the images based upon Conceptual Graphs. It details spatial and descriptive relationships between the various objects within the image. For example, a particular graph may encode "A woman is standing to the left of a man and in front of tree, the man is watering a horse, the horse is in a stable". Included in the representation is the polygonal area of the image that each object occupies. An indexing environment was developed to support the manual generation of the spatial and structural relationships in a graphical manner, and to allow their combination with the existing textual annotations.

Returning to the indexing used here: At indexing time, the stemming algorithm (Porter80) was accidentally enabled in the engine. This resulted in the French words of the annotations being stemmed by a vowel/consonant-counting stemmer designed for English! This error was not noticed until after the evaluation had been completed. Nonetheless, the system consistently gave results that appeared reasonable to onlookers, and when informally compared with the results with the stemmer disabled, there was little or no difference. This is perhaps not surprising when the conclusions of [Sanderson94] are considered – i.e. the effectiveness of traditional IR techniques is remarkably resilient to ambiguity in the term space.

The users

Twelve volunteer-users were recruited. They were all postgraduate-level educated, and familiar with windowed environments, graphical interfaces, and had experience with a number of searching systems. They had not used this system before.

The tasks

The three Ostensive Relevance profiles were implemented in the system of Chapters 6 & 7. The evaluation extensions (Section 7.4) allowed a user to simply sit down and select their name, and have the evaluation session controlled automatically.

Ten queries were randomly selected from the thirty of the test collection. A starting point was determined for each query by simply translating (rather naïvely) the query into French and supplying that as a virtual document to the system.

08.	Streets where we see children.	rue enfants
10.	Religious art (but not wall carvings)	croix procecessionelle reliquaire eglise
13.	Staircases	escalier
16.	Transportation	transport omnibus train
18.	Interiors	vestibule interiour salle
19.	Street scenes	rue de paris boutique
22	Men with beards (i.e. not just a moustache)	homme
24.	Floods	innondations
28.	Boats	barques
30.	Parkland/countryside	parc

Users were asked to browse from the starting point and to identify and mark images that they thought were relevant to the queries. Users marked/unmarked such images using the right-hand mouse-button as described in Section 7.4.

Informal tests suggested that five minutes was more than sufficient time to explore from a starting point and encounter a reasonable number (five to ten) of relevant documents. Therefore, in the evaluation, users were given five minutes to spend on each task.

The ten queries and three profiles under test were distributed across the users, such that each profile was used towards the beginning and towards the end of a user's five-task session. This was intended to reduce the influence of inevitable user learning over the course of their five allotted queries. No user did the same query twice.

The laboratory arrangements

Each user was given a five-minute demonstration and explanation/discussion of how to use the system. They were then told what was required of them.

The users were explicitly asked not just to mark every document that remotely looked as if had a connection with the query. They were asked to choose those images that, *in their opinion*, genuinely appeared relevant to the given query.

It was impressed upon users that it was not their ability to find relevant images that was under test. They were told that it was different arrangements of objects and links that were being tested with respect to what they (the users) regarded as relevant images. It was made clear that the marking/unmarking of objects did not affect the system in any way – it was merely a tag for them to indicate those they regarded as relevant. They were told that some tasks might be less productive than others.

There was no mention of a retrieval system working in the background. The idea was for users to feel that they were performing a confirmatory role on a pre-determined hierarchy. This corresponds to the actual situation quite well – i.e. although the Ostensive Model approach generates the links based upon the path taken to an object, as soon as a start position is established all possible paths from it (although infinite in number and length) are implicitly determined.

Each user was given one practice query, plus five real queries. Each query had a predetermined document as a starting point. The users practised for five minutes using the practice query, with supervision on demand. The practice query was the same for all users. The system automatically presented the query to the user and set up the corresponding virtual document as the starting point.

At any given time, between one and four users were trained and then performed their tasks simultaneously – all supervised by a single person. The time for a user to complete an evaluation session was 5 minutes demonstration, 5 minutes practice, then

5 x (5 minute task, plus a 1 to 2 minute rest) – i.e. 40 to 45 minutes in total. Therefore, in about 45 minutes, a single supervisor was able to collect data from twenty runs (i.e. four users, five queries each).

Summary

Each of the ten queries was run twice using each of the three profiles (i.e. six runs in total). Each user did five runs. No user did the same query more than once, therefore, each query was run by six different users. The profiles were distributed across the five runs of each user. In total, sixty runs were made (i.e. twenty for each of the three profiles).

For each run, the system logged all images that were presented on-screen, all images that were selected by the users as current objects, and all images that the user had marked as relevant. As the user was given the freedom to mark and unmark images at will, the system logged only those marked as relevant at the end of the session. The order of all image presentations and selections was logged – allowing the whole session to be played-back if desired.

During the runs (i.e. after the demonstration and the practice query), *no* questions or problems were encountered by the users. The supervision of the users, therefore, amounted to merely reminding them periodically of the remaining time for each run and then thanking them for their help at the end.

9.3 Interpreting the assessor-count

The use of a test collection with multiple relevance judgements provided an opportunity to observe differences in the effect of different interpretations of the assessor-count. Such interpretations can be regarded as orthogonal to the comparison methods. The interpretations take the form of transformations from the 0..4 space of the assessor-count, to a 0..1 space of relevance.

Step-functions (i.e. thresholds) offer the simplest transformations (Fig 9.5). The outcome is a binary notion of relevance based upon the document's relevance popularity amongst the assessors. For example, with a threshold of three, an image is considered relevant if three or more assessors judged it so. As the test collection had four assessors (with no distinction made between their quality of their respective judgements) per query, this produced four threshold transformations:

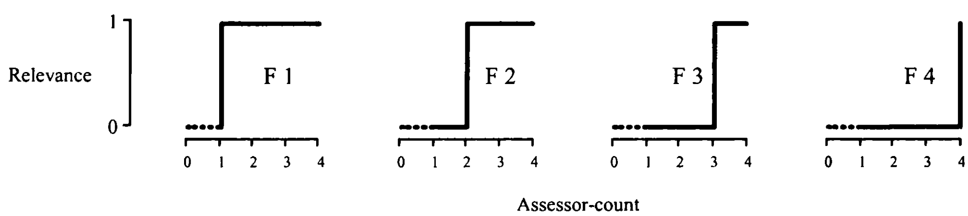


Fig 9.5 - Step function transformations of assessor-count.

Continuous functions offer more subtle transformations that do not destroy as much information. They transform the assessor-count into a value that can be said to represent a 'degree' or 'probability'' of relevance. Four such transformations were used (Fig 9.6). This gave a total of eight transforms, which were labelled *F1* to *F8*. *F5* is a simple linear relationship between assessor-count and relevance.

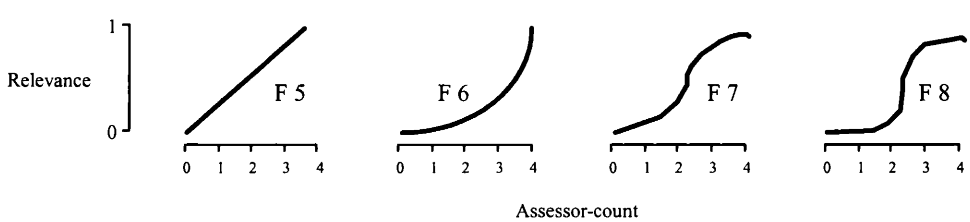


Fig 9.6 - Continuous transformations of assessor-count.

F6 encodes the intuition that each additional assessor that a document attracts gives a progressively larger increment in its relevance.

F7 and (to an exaggerated extent) *F8* code a slightly more complex intuition that attempts to incorporate not only subjectivity, but also error. A single assessor indicating relevance could potentially have done so in error (e.g. an error of transcription, or a misreading of the document due to distraction, or simple carelessness). Therefore, it should result in only a small increase in the degree/probability from zero assessors. Because of similar errors, the value for three assessors should be only slightly less than that for four assessors. One can argue that the probability of a second assessor making the same error, to that of a first assessor, is lower. Therefore, the increment in relevance associated with that second assessor should be greater – i.e. the second 'vindicates' the first, and now allows it to be treated with a higher weight. A similar argument applies to the step from three assessors down to two. These arguments on a discrete scale of four assessors when taken to a limit give the S-curves of *F7* and *F8*. A larger number of assessors in the test collection would allow continuous functions such as these to be expressed more effectively.

These eight functions produced relevance values that could be attached to a document with respect to a query. Therefore, a total relevance was measurable for each user performing a query in the evaluation, and for the maximum achievable for each query. Thus allowing per-run recall comparisons to be made.

That 'total relevance' is regarded as indicative of the relevance that would be perceived by the target population (i.e. as captured from the test-collection assessors).

9.4 Comparing the Ostensive Relevance profiles

The logging performed by the retrieval system during user-evaluation runs (using the method described in Section 7.4) provided raw data on the number of objects that were seen, selected, and marked as relevant. That information, along with the test collection and the eight assessor-count interpretation functions, provided the data for the following analyses.

Average Recall

Using the eight assessor-count interpretations, the recall values can be calculated for the twenty runs per profile. Averages of these are shown in the following table:

	Number of documents			*Recall after interpretation*							
Profile	*Seen*	*Selected*	*Marked*	*F1*	*F2*	*F3*	*F4*	*F5*	*F6*	*F7*	*F8*
CO	132.2	15.6	13.5	.25	.34	.43	.57	.33	.37	.34	.35
CB	157.9	19.6	16.4	.28	.37	.48	.59	.36	.41	.38	.38
DIC	153.4	18.2	15.6	.29	.41	.53	.62	.39	.44	.41	.42

Although *DIC* had less documents marked relevant, on average, than *CB* (15.6 versus 16.4), its recall levels were higher across all interpretations. This is because the fewer documents (on average) that were marked relevant by users whilst using *DIC* had a higher average relevance (with respect to the test collection).

It can be seen that various interpretations produce different relative values of recall for any particular profile. Nevertheless, they do not change the rank position. This is as expected because all the functions chosen were monotonic increasing, the relevant documents were only counted once, and the recall calculation is a monotonic function.

It can be seen that an Average Recall analysis supports the *DIC* profile over the traditional relevance feedback *CB* profile, for all interpretations.

Highest recall

Averaging the values for the two runs of each query gives ten recall values for each profile under each interpretation. From that, it is possible to count the number of queries in which a particular profile produced the highest recall. The following table presents these 'scores'. Shared first-places mean that the scores in the columns do not add up to ten. The shared first-places were not necessarily achieved with the same marked documents, nor were they with the same number of marked documents.

	Number of queries where highest recall was achieved							
Profile	*F1*	*F2*	*F3*	*F4*	*F5*	*F6*	*F7*	*F8*
CO	2	1	3	4	1	2	2	2
CB	4	4	4	5	4	3	4	3
DIC	5	6	6	8	6	6	6	6

The difference in the relative values across the interpretations is quite noticeable. *DIC* is a clear winner across all interpretations. The more stringent the threshold, i.e. going from *F1* towards *F4*, the more marked is the apparent superiority of the *DIC* profile. Therefore, as *DIC* most often produced the highest recall, this analysis clearly supports the *DIC* profile.

This simple analysis can be criticised as being naïve and reductionist. That is, it hides information such as the possibility that one profile wins by a tiny amount on two or three queries, whilst losing only on one query, but by a huge amount. For example, it might be an important aspect of the target environment that highest possible effectiveness would be traded for lower probabilities of catastrophic failure. Nevertheless, in some real-world environments, having a retrieval system that most often provides the best effectiveness may be the desired goal, and therefore, a valid selection criterion.

Statistical significance

Pair-wise statistical significance between the twenty runs of each profile was measured using the *uncoordinated t-test*. All interpretations produced different significance values. None of the values reached a level sufficient for even *one-tailed* significance at the 10% level. The values for comparisons between *CB* and *DIC* were generally twice that for comparisons between *CB* and the other two – suggesting that it lies somewhere in the middle between *CO* and *DIC.*

Returning attention to the two-run per-query averages used in the 'highest recall' analysis, there is a single recall value for each profile under each interpretation. As each query can be considered as a distinct microenvironment, these recall values can be regarded as ten related (i.e. coordinated) pairs. Therefore, the *coordinated t-test* becomes appropriate.

The tests were applied such that positive values would be consistent with the expected ordering of profiles – i.e. *DIC* better than both *CB* and *CO*; and *CB* better than *CO.* The following table shows the significance values (i.e. values of t) for the three profile comparisons under each interpretation. Values over the critical value for one-tailed significance at 5% (i.e. > 1.13) are shown emboldened:

	Significance values							
Comparison	*F1*	*F2*	*F3*	*F4*	*F5*	*F6*	*F7*	*F8*
CO < CB	**1.20**	0.93	**1.39**	0.42	**1.28**	**1.23**	**1.25**	**1.19**
CO < DIC	**1.34**	**1.72**	**1.64**	0.88	**1.67**	**1.66**	**1.63**	**1.74**
CB < DIC	0.33	0.93	1.03	0.74	0.76	0.87	0.79	0.89

The corresponding minimum critical values:

	Significance level			
Test	*10%*	*5%*	*2%*	
2-tailed	1.83	2.26	2.76	
1-tailed	0.92	**1.13**	1.38	(9 degrees of freedom)

None of the results is significant when the two-tailed test is applied. Given that there is a strong intuition to support both *DIC* and *CB* being better than the *CO* profile, a one-tailed test can be applied.

The *CB* profile is significantly better than *CO* (at the 5% level) under all interpretations except *F2* and *F4* – with *F2* achieving the weaker 10% significance level.

Under all but the *F4* interpretation, the Ostensive Model's *DIC* profile is significantly better than the lower bound *CO* at the standard 5% significance level. In fact, the same is true (apart from under F1 and F4*)* at the stricter 2% level, thus making this result as good as certain.

With respect to *CO*, the significance levels achieved by the DIC were higher, under all interpretations, than those achieved by *CB* – *suggesting* further that the DIC profile is better than *CB*.

The *DIC* profile is not significantly better (at 5%) than *CB*. Nevertheless, under *F2* and *F3,* the weaker 10% level is achieved.

The distribution of recall

As a visualisation, the following distributions show how many of the ten queries had a particular recall level, for each of the three profiles. Fig 9.7 shows this for F8 – the same general form was echoed for *F1* to *F7*.

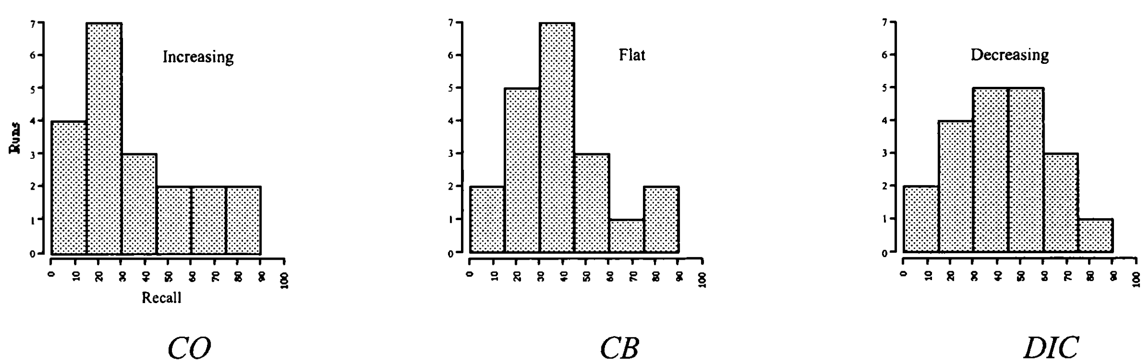


Fig 9.7 Distribution of the number of queries against Recall level, for F8.

The marked difference between *CO* and *DIC*, and the lesser difference between *CO* and *CB* that is evident in the significance table is echoed in the distributions. The *lack* of significance between the *CB* and *DIC* is perhaps a little less consistent with the impressions given by the distributions.

The distributions do suggest a general rightward (i.e. increased recall) shift as the profiles move from all-context (*CO*) through mainly-context (*CB*) to a balance of context and object (*DIC*).

Summary

All results are consistent with expectations from the theory and intuitions, with the 'highest recall' analysis showing a clear superiority of *DIC*. The lack of statistical significance was disappointing, except that its low level in the comparison between *CB* and *DIC* simply confirms that the two profiles (as measured in this experiment) are similar in nature. Given the belief that they are in fact similar, to confidently claim that one is better than the other, a two-tailed test (with its correspondingly stricter critical values) would be appropriate. Nevertheless, given that this experiment was originally conceived as only a pre-test, the results were unexpectedly encouraging.

9.5 Lessons learned from the evaluation

As expected, the evaluation provided more information on its *manner* of evaluation than on its comparative results. Some thoughts resulting from it are presented here, in no particular order.

Number of runs, number of users

Uncoordinated t-tests on all the runs (ignoring query associations) produced very low statistical significance values. This was due to the small number of runs available (i.e. 20 runs per profile) and the higher than expected variability in the data obtained across both users and queries. The low number of runs probably also reduced the validity of the two-run averages used for the query-oriented coordinated t-test results.

Due to the low-cost and high-speed of the laboratory technique, a larger number of runs can easily be obtained from each test user. This would reduce the number of extra people required as test-users. Further, additional users and runs can be arranged if required, and the existing evaluation data set expanded. This can be continued until statistical significance is reached, or confidence has increased sufficiently to decide that there actually is no difference to be found.

The profiles under test

An inappropriate concept of 'traditional' was used to motivate the choice of profiles to put under test. The flat relevance feedback profile (*CB*) is only traditional in the theoretical sense with respect to iterative query-based searching. The environment, within which the evaluation is taking place, is that of a browse-based system. Therefore, perhaps a more appropriate, and indeed telling, comparison for the *DIC* would have been against a context-less profile such as that from 'traditional' pair-wise similarity browsing. Both *CB* and *DIC* have a sense of the current object and its context, but just a different balance between them. Nonetheless, this evaluation did rule out the *CO* profile, and point further investigation towards the spectrum between context-biased, through balanced, to context-less profiles, and it did present a *suggestion* that the *DIC* is more effective than the *CB*.

Event time-stamping

The order in which user-interface events occurred was logged by the system – but the *time* at which they happened was not recorded. This oversight prevents analyses based upon the point in the session at which an event was performed. For example, one profile may have allowed most of the relevant images to be found within the first minute, whereas another might have required the whole five minutes to produce the same number

Assessor-count interpretations

Although the various assessor-count interpretations did not (could not) alter the rank positions of the profiles in the comparisons, *F4*, with its strict relevance criterion, reduced the number of relevant documents available over which measurements were made. This reduction in resolution resulted in non-statistical significance of the comparisons based upon it. Apart from that effect, there is a discussion to be had as to whether such monotonic increasing transformations exert any significant influence on the results of an evaluation.

Starting points

The starting points were almost exclusively 'good' in the sense that they were immediately, or single-step adjacent to one or more relevant documents. This would have allowed even the lower-bound *CO* profile to provide a number of relevant documents – despite it preventing any real movement through the document space. A larger document collection, with correspondingly higher numbers of relevant documents, and larger numbers of more distant, slightly relevant, documents might have shown increased differences.

CO would have allowed browsing in only the immediate locality of the starting point. The context-biased *CB* profile would allow browsing a little further afield – until each new document added to the path began to be overwhelmed by all those already there. With that profile, there is little sense of a path as such, only an incremental addition to a collection of images regarded as relevant. In contrast, the *DIC* profile has a balance of object and context. This allows a user to browse away from the starting point indefinitely, but always maintaining a sense of where he came from and the route taken.

Marking/unmarking objects

The system logged only those documents whose on-screen object representations were marked as relevant when each task ended. A log of all 'marking' and 'unmarking' events may have provided information as to how the user's information-need changed throughout the session as the user encountered new documents.

9.5.1 The design of a more comprehensive evaluation

Most users reported that they were finding very few new documents towards the end of each five-minute task. This anecdotal evidence (apart from supporting the assertions of the above 'Assessor-count interpretations' and 'Starting point' observations) suggests that less time could be assigned to each task. That would mean that more tasks could be completed by each user in the same or a slightly longer session. It is likely that a doubling of the number of runs per profile (i.e. to four per query – a total of forty) and a 50% increase in the number of queries could be achieved and still allow four profiles to be evaluated without much expense.

Were the multiple four-user sessions staggered by fifteen minutes or so, twelve users could potentially be supervised simultaneously by a single person, dramatically increasing the rate of data collection. Further, comments from users suggest that they could be asked to perform more runs per session. Some users indicated a willingness to return to perform additional sessions. This appeared to be a result both of the users finding the searching environment novel, and the images of the collection interesting.

The next step is to try several profiles, each subtly different, around the area of the best one from this evaluation (i.e. variations of the Ostensive Model). These will most likely be context-biased (*CB*), balanced (*DIC*), slightly flatter than balanced, slightly steeper than balanced, and the extreme context-less profile of existing browsing approaches. This would indicate, not only if the Ostensive Model's profile is better than the traditional browsing profile, but also if a variation of it were more effective. Nevertheless, given the very subtle (if any) difference between *CB* and *DIC* found in this evaluation, it must be accepted that it may be unlikely that any significant difference would be discernible between *DIC* and profiles that are biased only slightly towards either context or current object.

There may be other differences between the effects of different profiles that can be identified and for which a metric can be developed. For example, using interface event time-stamps, earlier success might be identifiable for one or more of the profiles. Such comparisons might be possible, even if statistical significance is still not achieved with the existing metrics, even on larger sample spaces.

The issue of starting point is difficult – it is another variable that will multiply the runs required. The difference between the profiles may well be larger when used with ‘bad’ starting points. Nevertheless, it is problematic to form intuitions or expectations as to how ‘good’, on average, a real-life starting point would be. There is the additional problem that the generation of starting points has not been developed in the Ostensive Model work so far – in particular, how they may be generated without descriptive querying. Nonetheless, it could be argued as desirable for an information-seeking environment to support effectively a user whom is in a less than optimal position. Therefore, if only one kind of starting point can be evaluated affordably, then a 'bad' starting point would seem a reasonable choice.

9.6 Summary

A low-cost approach to comparing Ostensive Relevance Profiles in an interactive non-textual environment was described, along with the results and considerations of an evaluation, comparing three profiles.

The evaluation technique was not only low-cost, but also rapid. It required very little training of users, almost no supervision during the sessions, allowed multiple users to be trained and supervised simultaneously, and had a time-per-run of only five minutes (and potentially less). This allows it to generate a large number of test points in a short time, with a minimal supervisory overhead.

Several shortcomings, of various severities, of the evaluation technique were highlighted – nevertheless, they offer clear indications of improvements to be made.

Eight transformations from assessor count to relevance did not alter the results of the rank positions within the profile comparisons used here. It was pointed out that under such conditions (which are common to many experiments in IR) they could not. Nevertheless, they did alter the size of the differences between the effectiveness results for each profile. As a result of that, and of them affecting the amount of information available for comparisons, statistical significance was prevented in a number of instances.

The results have successfully ruled out context-only Ostensive Relevance profiles. It has narrowed the area of study down to that between context-biased, through balanced, to context-less profiles. Further, the results *suggest* that the balanced 'document in context' is better than the traditional flat 'context biased'. This encouraging result points in the direction of a larger evaluation to confirm the suggestion.

Part V:

The Conclusions

In this final Part, I present what I regard as the achievements of the work of this thesis – both the individual, and the overall achievements. I then present the further work that I believe will either bring to completion some of the loose ends, or take the ideas on to new developments.

10 Summary and remarks

This chapter presents, in summary form, the achievements of the work of this thesis. It presents the many individual achievements made, and then presents five general achievements that can be claimed for the work as a whole. From those, and from comments made throughout the work, it proposes ways in which the work can be taken further.

Contents of this chapter

Section 10.1 lists the individual achievements. Section 10.2 presents the general achievements. Section 10.3 presents proposals for further work.

10.1 Individual achievements

This section gathers the achievements reported throughout the thesis. Each is listed along with the chapter in which it was presented:

An information-lack (Chapter 1)

Emphasising an information-lack highlighted the extent of the problems with the concept of describing an information-need – problems amounting almost to an inherent contradiction, i.e. how can one describe something one doesn't know about.

Characterisation of information-needs (Chapter 1)

Three structural (development, multiplicity, and tangentiality) and two operational (embedding and threading) characteristics of information-needs were presented.

Terms of reference and a comparative analysis (Chapter 2)

The elements of the characterisation were extended to include "an avoidance of information-need description" and "support for non-text media" to form seven terms of reference. Using them, a comparative analysis was presented of query-based and browse-based approaches. The analysis showed both approaches having specific advantages and disadvantages but moreover showing the complementarity of their support for information seeking.

A hybrid path-based approach (Chapter 3)

A hybrid of the two approaches was presented that used the novel concept of obtaining a handle onto a developing information-need from the nature of objects in a path. It proposed combining the evidence collected from the path objects by weighting the evidence based upon their age.

Contextual interpretation of documents (Chapter 3)

The importance of context in IR has been highlighted and its recognition argued (e.g. [Ingwersen94], [Ingwersen96], and [Cornelius96]) – but operational models that incorporate it have not resulted. The Ostensive Model has essentially taken up that challenge.

Based upon the particular route taken to reach a document, the path-based approach would form different views of where the user might wish to go next. Those views are based upon the current object, but *influenced* by the objects on the respective paths. If there was a theme to be discerned from those objects then that would effectively be the 'context' within which the document was 'interpreted'.

A model of the iterative development of information-needs (Chapter 4)

A model was presented of certain aspects of the cognition of information-needs during a searching session. The model was a formalisation of the hitherto informal path-based approach. Central to the model is the exposure of the user to information and the changing indicativeness of those pieces of information with respect to the current information-need.

Setting the model in a framework of Ostension (Chapter 4)

The operation of the model and the intuitions to which it appeals were identified as a process of ostensive definition. In the philosophical literature, Ostension has traditionally been restricted to discussion of language, therefore, there is an implicit assumption of its application to evidence gathered from explicit acts of communication. A relaxation of that restriction was proposed to include passively observed evidence of the sort to be found in the path-based approach. Reframed within the framework of ostension, the path-based model became the "Ostensive Model".

The distinction between active and passive evidence was highlighted as the key differentiator in approach between Relevance Feedback and the Ostensive Model respectively.

The conception of Ostensive Relevance (Chapter 5)

A novel conception of relevance was introduced – one that, at a conceptual level, recognises the developing nature of information-needs; and that, at an operational level, recognises the importance of the particular techniques/algorithms used to infer from observed evidence.

Probabilistic integration of models (Chapter 5)

An assumption was identified with respect to an estimation in the Binary Probabilistic Model – the assumption of *equality* of relevance of objects marked relevant by a user. The Ostensive Model motivated a weakening of that assumption, placing profiles of decreasing-with-age Ostensive Relevance across the objects.

Essentially, a cognitive model of information–needs and an operational model of retrieval were integrated, within a probabilistic framework, to produce a model of retrieval with more intuitively appealing properties.

The Retrieval Engine (Chapter 6)

A fast, flexible, and reliable networked text IR server was built. The engine can simultaneously service multiple clients, each accessing a variety of collections. It offers the Binary Probabilistic Model for searching, with Relevance Feedback. It also incorporates the integration of Chapter 5 and thus offers Ostensive Model searching. Beyond the work of this thesis, the Retrieval Engine has been used to provide retrieval services in experiments by other investigators.

Ostensive Browser user-interface (Chapter 7)

A user-interface was built instantiating the ideas of the Ostensive Model. It is novel in that it presents a graphical objects-and-links surface, the structure of which is determined by selections made by the user – thus incorporating the two prominent features of browse-based and query-based systems respectively. The interface is truly media-neutral because of its complete hiding from the user of all internal representation and retrieval techniques – the user interacts only with the objects being searched.

It presents a fish-eye view that facilitates visibility of the whole browsing surface, whilst retaining detail at the focus. The particular hybrid view-transformation developed for the Ostensive Browser also has the property of avoiding collisions at the focus.

The interface operates in one of two modes – a normal mode for searching, and an evaluation mode that incorporates a series of extensions to support interactive evaluation. The evaluation mode allows particular tasks to be presented to the user, and internal settings (e.g. selection of Ostensive Relevance Profile) to be automatically configured on a task-by-task basis. All such internal reconfiguration is invisible to the user. The extensions also allow relevance indications to be collected from users, and provide logging of user actions during the session.

Image test collection (Chapter 8)

A method for rapidly building an image test collection with binary relevance assessments was described. The method is extensible, allowing additional assessments to be incorporated.

Application of the technique produced a test-collection consisting of 666 general interest images, 30 queries, with 4 binary assessments per image/query. The multiplicity of assessments was intended to capture some of the subjectivity and lessen some of the error in assessments – examples of this were presented.

Evaluation method (Chapter 9)

A low-cost and fast method for the comparative evaluation of Ostensive Relevance Profiles was presented. Its speed and low supervisory-load was partly a result of the specific facilities offered by the Ostensive Browser when in its evaluation-mode. A critique of the method was presented, along with suggestions for its improvement.

Comparison of three Ostensive Relevance Profiles (Chapter 9)

Eight interpretations of the test-collection's binary assessor-counts were presented and motivated – four dichotomous and four continuous functions. These were used to translate the assessor-count into degrees of relevance for the images in the test-collections.

An analysis of the results of the comparative evaluation of three Ostensive Relevance Profiles was presented. The effect of the eight assessor-count interpretations were then used in the analysis of the results. The analysis confirmed that monotonically-increasing assessor-count interpretation functions would not effect the order of results, but that it would effect the relative values achieved in an evaluation, and hence affect the statistical significance of comparisons.

The ‘context only’ profile was ruled out by the evaluation. Lack of statistical significance prevented a *conclusion* being reached on the relative merits of the ‘Document in context’ profile proposed by the Ostensive Model, and the ‘context biased’ profile used by traditional Relevance Feedback. Nevertheless, the weak significance that was observed *suggests* that the ‘document in context’ is more effective under this evaluation environment.

The analysis of the evaluation suggests a number of improvements that could improve the chances of attaining statistical significance in the results. The most obvious suggestion is that of a larger numbers of runs. A less obvious example is reducing the ‘goodness’ of the prescribed starting points in the evaluation.

10.2 Overall achievements

Several achievements have been presented individually throughout this thesis, and listed in the previous section. In this section, five things are presented that I believe have been achieved when the work of the thesis is taken as a whole:

A shift from non-core procedural tasks to core functional tasks

Procedural elements of the interaction of users with existing systems were highlighted in Part I, these included:

- Generation, appraisal, and modification of effective terms – i.e. the need to think up words that might be in relevant documents, subsequently appraising their effectiveness in the light of retrieved documents, and finally thinking about what changes should be made to the selection of terms make up the query. On top of that was the effort of generation, appraisal, and modification of the Boolean operators or the weights that connected those terms.
- When Relevance Feedback is applied, the procedural tasks associated with terms are replaced by similar tasks managing lists of relevant-indicated documents.
- The effort of managing the embedding and threading of an information-need using, at best, a history of queries previously used.
- When working with a graphical presentation of a space (and, in particular, a growing space), the user was unable to view the whole space at one time and was forced to scroll around.

The application of a range of technologies from existing and simple ones such as the object thumbnails, through the fish-eye view, to novel and sophisticated techniques such as the ostensive path-based adaptation of the space, either removed or reduced such procedural tasks.

The scarcity of controls on the Ostensive Browser and its lack of traditional 'dialogue' between the user and the system is testament to the shift from non-core procedural tasks to a concentration on the core functional task of identifying relevant documents.

Provision of a media-neutral information-seeking environment

This is made possible only because of the above removal of things procedural from the interaction. The importance of the above is that the procedural interactions in traditional approaches are representation-specific and hence media-specific. Our familiarity with text as a communications medium disguises the fact that it is also the media-specific internal representation method for retrieval of text documents. It was an original desire to find some way of hiding representations that prompted the necessity to remove the query from the interactions, which in turn required a replacement, and provoked the conception of paths as a suitable source of evidence.

Instead of simply stating something such as the rather weak "media-neutrality is a desirable goal", the work of this thesis was driven by the recognition that *we have no alternative*. As demands for non-text media and multi-media retrieval grow, relying upon textual surrogates (e.g. annotations) becomes untenable due to the costs involved of generating them. Techniques for making similarity measures on non-text media are being developed, and these almost exclusively do not have human-accessible representations. Even if the equivalent of a text-query exists in the particular medium (e.g. a sketch of what image is wanted), the same problems exist of modifying it to reflect developments in the information-need – then there is the same old compounding problem of managing those specifications to support embedding and threading of the information-need.

The Ostensive Model, and the example interface provided by the Ostensive Browser, permit developing, embedded, and threaded information-needs to be supported in a representation-free and media-neutral environment.

The existence of an approach such as that of the path-based Ostensive Model could be argued to offer a degree of freedom to the development of non-text media representation techniques and retrieval algorithms. That work need no longer be constrained by practical considerations of how to translate textually specified requests, or how to facilitate incremental update of such requests. All that a new algorithm and representation need be able to do is take objects as examples, along with weights indicating their individual representativeness.

Instantiation of a cognitive idea into a working IR system

The two disciplines that have an involvement in Information Retrieval are those of Computing Science and Information Science. Generally speaking, the former is where one finds the development of probabilistic models and the building of systems, and the latter is where the development of cognitive models and the execution of user-studies are at home. There is relatively little interaction between the two disciplines; the work of this thesis builds a cognitive model with a strong relationship to an existing model in Information Science and instantiates it, first as a probabilistic model, then as a working retrieval system.

A combination of Objective and Subjective probabilities

In the philosophy of Probability, there has been a split between the ideas of Objective and of Subjective probabilities. The 'Objective' view is the traditional one that relates to probabilities that are grounded logically or statistically in measurable or countable events, and that considers them to amount to 'real' things. The 'Subjective' view is the more recent of the two, that claims probabilities can be set without recourse to counting or logic, and that they are essentially no more real than any other idea or opinion held by a person – a view made popular by [deFinnetti74]. The two views of probability are often regarded as mutually exclusive alternatives, and when not, they are regarded minimally as incompatible[1].

The Binary Probabilistic Model of IR is a formulation of 'Objective' conceptions of probability in the traditional mould. The Ostensive Relevance Profiles of the Ostensive Model are 'Subjective' in nature – i.e. they are a non-counted, non-logically-derived opinionated probabilistic weighting functions. The work of Chapter 5 presents, therefore, not only an integration of those two IR models, but also a concrete integration of the two philosophies.

Formalisation and implementation of intuitions and observations

It should be clear, particularly from the nature of Chapter 3, that the work of this thesis was borne in informal observations of searching behaviour, rather than the more *commonly claimed* deduction from existing models. Those observations,

[1] An excellent overview of this and other philosophical Probability Theory is given in [Cohen99].

combined with an understanding of what was happening within the IR systems, formed intuitions of what was happening and how it might be improved.

Forming intuitions is nothing unusual, but this thesis took those intuitions, formalised them into a model, and successfully integrated that model into an existing model. The result was (again, by appealing to intuitions) an improvement over the existing model. The integrated model was implemented in an operational system giving it a number of novel characteristics. An evaluation was performed of the intuitions that were central to the work. Finally, encouraging results from the evaluation point to, and perhaps more importantly, justify further in-depth investigation.

10.3 Taking things further

This section presents proposals for ways in which this work could be furthered. Some are directly motivated by recognised shortcomings of what has already been done, and others constitute potentially interesting developments. There is a weak ordering in the following, with those that remedy perceived shortcomings presented first, and the final two offering particularly attractive developments.

Starting Points

Not much has been made of the starting conditions for a searching session using the Ostensive Model, whether in the environment of the Ostensive Browser, or otherwise. In fact, one might be cruel enough to say that it has been conveniently ignored!

With an often-reiterated drive for the hiding from the user of internal representations, and more specifically, the removal of queries, the fact that a searching session using the Browser is started by providing a query, might seem contradictory. It is so, but only because time has not been set-aside for its removal. This was partly because there are already a number of approaches to starting a session, and it was the *development* of the session that was seen as the challenge. With path-based ostensive browsing now in place, attention can turn to consistent ways of generating starting points for paths in the absence of internal representations.

The simplest approach is, in fact, already present in the Browser – that of starting from an object. The current interface allows a user to start a whole new path-tree from any object. Generalising this, the Browser could accept an example object from the user that came from outside of the collection currently being searched.

A further generalisation could be the provision by the user of a number of such external documents. By, for example, finding the centroid of those objects, the Browser could form a staring point. The objects offered by the user could be of different media types (depending upon the capabilities of the available underlying retrieval facilities).

To completely internalise the starting conditions to the collection objects being searched, 'cluster descent' could be employed (e.g. the 'Scatter-Gather' approach

[Cutting92] & [Cutting93]). This involves clustering the collection (e.g. [Ling72], [Sibson73], [Hartigan75], [Willett80]) and presenting those clusters to the user. The user could then gain an overview of what was contained in the collection by browsing the clusters and their sub-clusters. At any point, the user could decide to start a browse and the Browser would use the centroid (or alternative representation) of the indicated cluster to form the starting point. It is perhaps worth noting why this would be different from simply searching using the cluster hierarchy: the clusters are static and would offer similar restrictions to those of traditional browse-based approaches, whereas the Ostensive Model would allow paths to develop regardless of such boundaries.

Incorporating one or more of the above would essentially take the Ostensive Browser to completion as a media-neutral, query-free, dynamic searching environment.

More comprehensive evaluation

This is also motivated by perceived shortcomings in what has already been done. Although the evaluation of this thesis was intended to be an investigation of the *method*, it is frustrating that it was not able to produce statistical significance. That was particularly so when the results that were obtained appeared so close to providing significant results in favour of the Ostensive approach.

Section 9.5 presents a number of suggestions for improving the evaluation, in terms of both *what* was evaluated and *how* it was done. The proposal here is that those suggestions are followed – in particular, increasing the number of runs, reducing the quality of the starting points (or even making that a subject of evaluation), and including the profile that corresponds to the traditional browsing approach. With hindsight, the omission of that particular Ostensive Relevance Profile was a glaring omission in the evaluation in Chapter 9. A further proposal is the investigation of other profiles each side of, and close to, the object-context-balanced profile.

True multimedia retrieval technologies

It seems the next most obvious thing to investigate would be a ‘real’ non-text retrieval mechanism underneath the Ostensive Browser. There are now many such techniques, the challenge would first be to determine candidates that can perform some kind of

Relevance Feedback operation, but that can also incorporate degrees of relevance to allow the Ostensive Relevance profiles to be applied.

Enhancing the test-collection

As the assessments of the current test-collection are extensible, the proposal is that they be extended – perhaps doubling or tripling the number of assessors. The questions one could then ask would include: Would more assessors (i.e. a higher resolution in image popularity) affect the statistical significance of evaluations using them? Would more assessors change the results of the evaluation (e.g. by introducing new previously non-relevant images, or by changing the relative popularity of individual images)?

More generally, one might investigate if there is a new conception of relevance to be formulated based upon popularity-of-relevance measures in test-collections.

Multi-lingual and cross-lingual retrieval

It is proposed that evaluations are carried out to determine if this approach, with its lack of queries, is an effective environment for cross-lingual searching – i.e. by freeing the user from the necessity to articulate their information-need in a foreign language, but by allowing them use their (likely) better ability to understand that language.

Naïve users

Naïve users (with respect to: the subject of their search, and/or to the use of retrieval systems) would be the kind of user most expected to: have limited fluency in the vocabulary of a text-collection and thus have difficulty in generating and appraising query-terms; be least able to manage effectively a query; be least able to manage effectively a set of relevant-indicated documents; have most difficulty in managing effectively multiple embedded and threaded information-needs; and be most distracted by tasks such as scrolling.

Might it be the case that the claimed advantages of the Ostensive Model and the Ostensive Browser are most apparent when exploited by such users?

Collaborative filtering

The techniques presented here used information from the *content* of the objects on a path to determine the most appropriate objects to present to a user as next-steps. Collaborative Filtering offers an alternative approach – using the correlation or co-occurrence of objects (using name only) in a population of paths captured by logging the activity of many users. Essentially, things are presented to a user that were commonly selected by other users when in a similar situation. The 'similar situation' is a high overlap in certain attributes, which can be anything from 'products bought', through 'documents regarded as relevant', to 'hyperlinks followed'. The approach has the advantage of exploiting information that amounts to prior probabilities from the collection of logged user experiences. Another advantage is that it is inherently media-neutral, as it treats each object as merely a content-less symbol. The disadvantage is that it relies upon the existence of large populations of such paths to be able to function effectively – further, those paths must be over the same objects.

Chalmers ([Chalmers98] & [Chalmers99]) has taken those techniques and applied a sophistication that is similar to the one applied in this thesis to the relevant-marked objects: He applies a temporal structure over the objects, and then looks for similar sequences in the logged paths to that of the user's current path.

The content approach presented here and the name/token approach of Chalmers are complimentary. The Ostensive Model can be seen a using dynamic and local evidence to drive the process of prediction, whereas the Chalmers model can be said to be using *a priori* and global evidence.

A comparison of the effectiveness of the two could be carried out using the Ostensive Browser to determine under which circumstances each is the more effective approach. The two path-approaches have obvious complementarity – clearly inviting the development of a hybrid of both. Work such as that of Robertson [Robertson97] suggest that evidence from the two sources might be most effective when biased towards the dynamic/local evidence of the Ostensive Model – although this might be strongly influenced by the effectiveness of the particular retrieval techniques driven by the Ostensive Model.

Dynamic Ostensive Relevance Profiles

Having determined empirically a generally suitable Ostensive Relevance Profile (through evaluations as proposed above), and given that it becomes the basis of the system's efforts to support a user, it is proposed that it be the subject of local optimisation during an information seeking session. This could take the form of either the profile being modified whilst retaining its general shape, or alternatively, the profile being completely replaced in a longer path by an optimised version:

For example, after each relevance indication by a user (i.e. a user clicks on an, as yet unexplored, next-step), the system could modify its relevance profile in such a way that it would have correctly (or at least more correctly) predicted the choice made by the user. That is, if the selected object was not the top-ranked object using the evidence from the path, the system could optimise its combination (e.g. by altering individual Ostensive Relevances) such that the object became the top-ranked. In this way, the system could build completely tailored profiles reflecting, as best it could, the development of each of the users' information-needs. This could be applied only along individual paths to objects, or it could be applied in a session-wide sense – striving to minimise the overall error made in predicting next-steps.

It may be possible to identify patterns in such optimised profiles – e.g. perhaps the optimised profiles for short paths tend to have a particular form that is distinguishable from those of longer paths.

If multiple retrieval models were available (or multiple parameterisations of a single model), the possibility exists for them to 'compete' over the length of a single path, or over all paths in a session. As a path or paths grow, documents followed-up by the user would increase the weight of those models that best predicted their selection. With each step, a model's influence in the next-steps would be proportional to its currently achieved weight. The 'influence' could be exercised either in its contribution to the score of individual documents, or it could be a particular number of the next-steps that that particular model is allowed to suggest.

Longer-term learning could be performed across individual sessions – perhaps on a per-collection, or per-user basis, or both.

Essentially, there is a wealth of opportunity for different kinds of observational evidence to be collected, different kinds of evidence combination to be applied, and different kinds of learning to be applied – all in an environment where their existence and mode of operation can be hidden completely from the user.

References

Agosti94

Agosti M., Melucci M. and Crestani F. (1994) TACHIR: a tool for the automatic construction of hypertexts for Information Retrieval. In: *Proc. of RIAO 94*, Rockerfeller University, USA, pp 338-357.

Ayer36

Ayer A.J. (1936) *Language Truth and Logic.* Penguin Books, London.

Ayer56

Ayer A.J. (1956) *The Problem of Knowledge.* Penguin Books, London.

Bates89

Bates M.J. (1989) The design of browsing and berrypicking techniques for the online search interface, *Online Review* **13:5** pp 407-424

Belkin82

Belkin N.J., Oddy R., and Brooks H. (1982). ASK for information retrieval: Part I. *Journal of Documentation* **38** pp 61-71

Belkin90

Belkin, N.J. (1990). The cognitive viewpoint in information science. Journal of Information Science, (16), 11-15.w

Borlund97

Borlund P. and Ingwersen P. (1997). The development of a method for the evaluation of interactive information retrieval systems. *Journal of Documentation* **53:3** pp 225-250

Borlund98

Borlund P. and Ingwersen P. (1998) Measures of relative relevance and ranked half-life: Performance indicators for interactive IR. In: *Proceedings of ACM SIGIR '98*, Melbourne pp. 324-331

Borlund99

Borlund P. and Ingwersen P. (1999) The application of work tasks in connection with the evaluation of interactive information retrieval systems. *Final MIRA Conference on information retrieval*, Glasgow pp. 29-46

Callan92

Callan J.P., Croft W.B., and Harding S.M. The INQUIRY retrieval system. In. *Proc. of 3rd Int. Conference on Database and Expert Systems Applications* pp.78-83

Campbell95

Campbell I. (1995) Supporting information-needs by ostensive definition in an adaptive information space. In: *MIRO'95. Electronic Workshops in Computing,* Ed. Ian Ruthven, Springer Verlag

Campbell96

Campbell I. and van Rijsbergen C. J. (1996) The Ostensive Model of developing information-needs. In: *Information Science: Integration in Perspective, Proceedings of the CoLIS-2 conference*, Eds. Ingwersen P. and Pors N. O., Copenhagen, 1996.

Campbell00

Campbell I. (2000) Interactive evaluation of the Ostensive Model, using a new test-collection of mages with multiple relevance assessments. *Journal of Information Retrieval,* **II:1**, pp 87-114

Chalmers92

Chalmers M., and Chitson P. (1992) Bead: explorations in information visualisation. In: *Proc. of ACIM-SIG IR* pp. 330-337

Chalmers98

Chalmers M, Rodden K, Brodbeck D. (1997) The order of things: Activity-Centred Information Access. In: Proceedings of WWW7, Brisbane. pp. 359-367.

Chalmers99

Chalmers M. (1999) Comparing Information Access Approaches. Journal of American Society for Information Science (JASIS) 50^{th} Anniversary Issue, 50(12), pp 1108-1118.

Chiaramella97

Chiaramella Y. and Mechkour M. (1997) Indexing an Image Test Collection, Technical report. *Chapter of Deliverable D10, FERMI ESPRIT Basic Research Action* 81034

Cohen99

Cohen L.J. (1999), Introduction to the philosophy of induction and probability, *Clarendon Press*

Cornelius96

Cornelius I. (1996) Information and interpretation. In: *Information Science: Integration in Perspective, Proceedings of the CoLIS-2 conference.* Eds. Ingwersen P. and Pors N. O., Copenhagen

Cosijn00

Cosijn E. and Ingwersen P. (2000) Dimensions of Relevance. *Information Processing and Management,* **36:4** pp. 533-550

Crestani98

Crestani F, Lalmas M, van Rijsbergen C J, and Campbell I. (1998) "Is this document relevant?"...probably. A survey of probabilistic models in Information Retrieval. *ACM Computing Surveys* **30:4** pp. 528-552

Croft79

Croft, W.B. and Harper D.J. (1979) Using probabilistic models of document retrieval without relevance information. *Journal of Documentation* **35:4** pp. 285-295

Croft92

CroftW.B., and Harper D., (1992) Using probabilistic models of document retrieval without relevance information. In *Proc. of ACIM-SIGIR* pp. 330-337

Cutting92

Cutting D.R., Pedersen J.O., Karger D.R. and Tukey J.W. (1992) Scatter/gather: a cluster based approach to browsing large document collections. In: *Proceedings of the 15th annual International ACM-SIGIR conference on research and development in information retrieval* pp 318-329

Cutting93

Cutting D.R., Karger D.R., and Pedersen J.O. (1993) Constant Interaction-time scatter/gather browsing of very large document collections. In *Proceedings of the 16th annual International ACM-SIGIR conference on research and development in information retrieval* pp126134

Ellis94

Ellis D, Furner-Hines J, and Willett P (1994). On the measurement of inter-linker consistency and retrieval effectiveness in hypertext databases. In: *Proceedings of ACM SIGIR '94*, Dublin pp. 51-60

Fermi94

The FERMI project (1994). *ESPRIT Basic Research Action 81034. Documentation and Technical reports web server.* "http://www.dcs.gla.ac.uk/fermi".

deFinetti74

de Finnetti B. (1974) Theory of probability: a critical introductory treatment. Volumes 1 & 2. Translated by Antonio Machi and Adrian Smith. Wiley series in probability and mathematical statistics.

Furnas86

Furnas G. W. (1986) Generalised Fisheye views. In: Proceedings of ACM CHI 86 Conference on Human Factors in Computing Systems. Boston, Massachusetts April 13-17. 16-23

Harman92

Harman D. (1992) Relevance Feedback Revisited. *Proceedings of the 15th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Copenhagen, 1-10

Hartigan75

Hartigan J. A. (1975) Clustering Algorithms. In: *Wiley Series in Probability and Mathematical Statistics*

IBM72

IBM (1972) Storage and Information Retrieval system (STAIRS). *General information manual, 2nd edition, IBM Germany, Stuttgart.*

Ingwersen84

Ingwersen, P. (1984). A cognitive view of three selected online search facilities. Online Review, (8), 465-492.

Ingwersen92

Ingwersen P. (1992) *Information Retrieval Interaction.* Pub. Taylor Graham, London.

Ingwersen94

Ingwersen P. (1994) Polyrepresentation of information-needs and semantic entities. In: *Proceedings of 17th ACM-SIGIR, Dublin,* pp. 51-60.

Ingwersen96

Ingwersen P. (1996) Cognitive perspectives of information retrieval interaction. *Journal of Documentation,* **52:1** pp. 3-59

Jeffrey65

Jeffrey R.C. (1965) *The logic of decision* Pub. McGraw-Hill

Lamping94

Lamping J. and Rao R. (1994) Laying out and visualising large trees using hyperbolic space. In: *Proc. of ACM-UIST '94* pp. 13-14

Lehman94

Lehman J.W. and Reid C.A. (1994) Knowledge-based searching with TOPIC. In: *Proc. of The second Text REtrieval Conference (TREC-2),* pp. 209-221

Lesk69

Lesk M. E. and Salton G. Relevance assessments and retrieval system evaluation. *Information storage and retrieval,* **4** pp.343-359

Ling72

Ling R. F. (1972) On the theory and construction of k-clusters. *Computer Journal,* **15** pp.326-332

MacKay69

MacKay D.M. (1969) *Information mechanism and meaning.* The M.I.T. Press.

Magennis97

Magennis M. (1997) Potential and actual effectiveness of interactive query expansion. *Ph.D Thesis. University of Glasgow.*

Maron60

Maron M. E. and Kuhns J. L. (1960) On relevance, probabilistic indexing and information retrieval. *JACM* **7** pp. 216-244

Mizzaro97

Mizzaro S. (1997) Relevance: The whole history. *Journal of the American Society for Information Science*, 48(9):810-832

deMey77

de Mey, M. (1977). The cognitive viewpoint: its development and its scope. In: de Mey, M., Pinxten, R., Poriau, M., and Vandamme, F., eds. International Workshop on the Cognitive Viewpoint. Ghent: University of Ghent, xvi-xxxii.

deMey80

de Mey, M. (1980). The relevance of the cognitive paradigm for information science, In: Harbo, O. and Kajberg, L., eds. Theory and application of information research. Proceedings of the 2nd International Research forum on Information Science. London: Mansell, 49-61.

Nelson94

Nelson P.E. (1994) The ConQuest system. In: *Proc. of The second Text Retrieval Conference (TREC-2)*, pp 265-270

Ounis98a

Ounis I. and Pasca M. (1998) Modelling, indexing and retrieving images using conceptual graphs. In: *Proceedings of the 9th International Conference on Database and Expert Systems Applications (DEXA '98), Vienna* pp. 24-28

Ounis98b

Ounis, I. (1998) Un modèle d'indexation rationnel pour les graphes conceptuels fondé sur une interprétation logique. *PhD Thesis. Université Joseph Fourier, Grenoble*

Oxford93

Oxford English Dictionary (1993) CD-ROM, 2nd Edition, Oxford University Press

Porter80

Porter M. F. (1980). An algorithm for suffix stripping. *Program*, **14:3** pp.130-137

Quine53

Quine W vO (1953). *From a logical point of view.* Harvard University Press.

Quine60

Quine W vO (1960) *Word and Object.* The MIT press, Cambridge, Massachusetts

Quine69

Quine W vO (1969) *Ontological relativity and other essays.* Columbia University Press.

Rennison94

Rennison E. (1994) Galaxy of news. An approach to visualising and understanding expansive news Landscapes. In: *Proc. of ACM-UIST '94*

vanRijsbergen79

van Rijsbergen C.J. (1979) *Information Retrieval.* Butterworths, London.

Robertson76

Robertson S. and Spärck-Jones K. (1976) Relevance weighting of search terms. *JASIS*, **27** pp. 129-146

Robertson93

Robertson, *et al.* (1993) *Okapi at TREC.* In: *Proc. of The first Text Retrieval Conference (TREC-1)* pp. 21-30

Robertson97

Robertson S. (1997) On relevance weights with little relevance information. In: *Proceedings of ACM-SIGIR, Philadelphia*, pp. 16-24.

Salton71

Salton (1971) *The SMART retrieval system – experiments in automatic document processing*, Prentice-Hall.

Sanderson91

Sanderson M. and van Rijsbergen C.J. (1991) *NRT: news retrieval tool. Electronic Publishing origination dissemination and design,* **4:4** pp. 205-217

Sanderson94

Sanderson M. (1994). Word Sense Disambiguation and Information Retrieval. In: *Proceedings of ACM SIGIR conference.* Eds. Croft W.B. and van Rijsbergen C. J. pp. 142-151

Saracevic96

Saracevic T. (1996) Relevance reconsidered '96. In: *Information Science: Integration in Perspective, Proceedings of the CoLIS-2 conference* Eds. Ingwersen P. and Pors N. O., Copenhagen, pp. 201-218.

Schaffer96

Schaffer D, Zuo Z, Greenberg S, Bartam L, Dill J, Dubs S, and Roseman M. (1996) Navigating Hierarchically Clustered Networks through Fisheye and Full-Zoom Methods. ACM Transactions on Computer-Human Interaction, Vol. 3, No. 2, June 1996, 162-188.

Shannon49

Shannon C. and Weaver W. (1949) *The mathematical theory of information.* University of Illinois press.

Shepherd91

Shepherd M.A. and Watters C. (1991) Virtual structures for hypertext. In: *Proc. of ACM SIGIR / ISS International Conference of Multimedia Information Systems,* pp. 201-219

Sibson73

Sibson R. (1973). SLINK: an optimally efficient algorithm for the single link cluster method, *Computer Journal,* **16** pp.30-34

Stein91

Stein R.M. (1991) Browsing through the terabytes. *Byte*, pp. 157-164

Thompson89

Thompson R.H. (1989) The design and implementation of an intelligent interface for Information Retrieval. *Ph.D., University of Massachusetts*

Voorhees98

Voorhees E.M. (1998) Variations in relevance judgements and the measurement of retrieval effectiveness. In: *Proceedings of ACM SIGIR '98, Melbourne*, pp. 315-323.

Willett80

Willett P. (1980) Document Clustering using an inverted file approach. *Journal of Information Science,* **2** pp. 223-231