Towards Nootropia : a non-linear approach to adaptive document filtering

Thesis

Towards Nootropia: a Non-Linear Approach to Adaptive Document Filtering

by Nikolaos Nanas

Diploma in Civil Engineering,
Aristotle University of Thessaloniki (1997)
M.S. in Intelligent Systems, University of Sussex (2000)

Submitted to the Knowledge Media Institute
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Artificial Intelligence

at THE OPEN UNIVERSITY

8 December 2003

Towards Nootropia: a Non-Linear Approach to Adaptive Document Filtering

by

Nikolaos Nanas

Submitted to the Knowledge Media Institute on 8 December 2003,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Artificial Intelligence

Abstract

In recent years, it has become increasingly difficult for users to find relevant information within the accessible glut. Research in Information Filtering (IF) tackles this problem through a tailored representation of the user interests, a user profile. Traditionally, IF inherits techniques from the related and more well established domains of Information Retrieval and Text Categorisation. These include, linear profile representations that exclude term dependencies and may only effectively represent a single topic of interest, and linear learning algorithms that achieve a steady profile adaptation pace. We argue that these practices are not attuned to the dynamic nature of user interests. A user may be interested in more than one topic in parallel, and both frequent variations and occasional radical changes of interests are inevitable over time. With our experimental system "Nootropia", we achieve adaptive document filtering with a single, multi-topic user profile. A hierarchical term network that takes into account topical and lexical correlations between terms and identifies topic-subtopic relations between them, is used to represent a user's multiple topics of interest and distinguish between them. A series of non-linear document evaluation functions is then established on the hierarchical network. Experiments using a variation of TREC's routing subtask to test the ability of a single profile to represent two and three topics of interest, reveal the approach's superiority over a linear profile representation. Adaptation of this single, multi-topic profile to a variety of changes in the user interests, is achieved through a process of self-organisation that constantly readjusts the profile stucturally, in response to user feedback. We used virtual users and another variation of TREC's routing subtask to test the profile on two learning and two forgetting tasks. The results clearly indicate the profile's ability to adapt to both frequent variations and radical changes in user interests.

στη Μιτσή και στον Φάδερ

to my parents

Contents

1 Introduction 1

2 Literature Review 7

2.1 Personalised Information Delivery 9

2.2 The Accessible Information Space 11

2.2.1 Dynamic Information Sources 11

2.2.2 Static Information Sources 12

2.3 The Statistical Regularities of Language 14

2.4 Document Indexing 16

2.4.1 Term Weighting for Document Indexing 20

2.5 Topic Representation 24

2.5.1 Topic Representation by Query 25

2.5.2 Topic Representation by Classifier 27

2.5.3 Term Weighting for Topic Representation 28

2.6 Document Evaluation 32

2.6.1 Document vs. Query 32

2.6.2 Document vs. Classifier 36

2.7 Term Dependence 38

2.8 The Obtained Information Space 42

2.9 Information Filtering 43

2.10 User Profile Representation 46

2.10.1 Single-Topic Representations in IF 47

2.10.2 Profile Initialisation 50

2.10.3 Document Evaluation 52

2.11 The Presented Information Space 55

2.12 The Read Information Space 57

2.13 Profile Adaptation 59

2.13.1 Profile Adaptation through Learning 60

2.13.2 Profile Adaptation through Evolution 65

2.14 Evaluating Information Filtering Systems 69

2.14.1 Evaluation Measures 71

2.14.2 TREC-2001 Filtering Track 72
2.14.3 Virtual or Synthetic Users 75
2.15 Summary and Conclusions 77

3 Building a Multi-Topic Profile 83
3.1 A User-Study on Profile Initialisation 84
3.2 Step 1: Term Weighting and Selection 86
3.2.1 Existing Term Weighting Methods 87
3.2.2 Relative Document Frequency 89
3.3 A Comparative Evaluation of Term Weighting Methods 91
3.3.1 Evaluation Methodology 92
3.3.2 Results 94
3.4 Step 2: Identifying Term Dependencies 100
3.5 Step 3: Generating a Hierarchy 102
3.6 Representing Multiple Topics of Interest 104
3.7 Summary and Conclusions 109

4 Non-Linear Document Evaluation 111
4.1 Layered Approach 113
4.1.1 Experimental Evaluation 116
4.2 Continuous Approach 121
4.2.1 Single-Topic Experiments 126
4.2.2 Two-Topic Experiments 130
4.2.3 Three-Topic Experiments 143
4.2.4 Discussion 150
4.3 Additional Evidence 153
4.4 Personalisation Services 156
4.5 Summary and Conclusions 159

5 Profile Adaptation through Self-Organisation 163
5.1 Networks, Self-Organisation and Autopoiesis 166
5.2 Self-Organisation Process 169
5.2.1 Step 1: Extract Informative Terms 170
5.2.2 Step 2: Update Profile Term Weights 172
5.2.3 Step 3: Remove Incompetent Profile Terms 174
5.2.4 Step 4: Add New Terms 175
5.2.5 Step 5: Re-establish Links 177
5.2.6 Overview 180
5.3 Experimental Evaluation 181
5.3.1 Experimental Methodology 182
5.3.2 Task α: parallel interest in two topics 186

5.3.3 Task β: a new topic of interest emerges 193
5.3.4 Task γ: forgetting a topic 201
5.3.5 Task δ: forgetting a topic with negative feedback . . . 209
5.3.6 Discussion of the Results 215
5.4 Summary and Conclusions 216

6 Conclusions and Future Research 219
6.1 Linearity in Information Filtering 219
6.1.1 Term Dependence Representation 221
6.1.2 Multi-Topic Document Filtering with a Single Profile 223
6.1.3 Profile Adaptation through Self-Organisation 224
6.1.4 Experimental evaluation 225
6.2 Future Work 227
6.2.1 Towards Nootropia 227
6.2.2 Projections 228

A Summary of Topic Codes 231

B List of Publications 233

List of Figures

2-1 Personalised Information Delivery 9

3-1 Results of experiment using summation of weights (eq. 3.2) . . 95
3-2 Results of experiment using multiplication of weights (eq. 3.3) 95
3-3 Normalised Weight distributions for IG, RelDf and Rel2DF . . 98
3-4 Associative links between profile terms. (Nodes (ellipses) represent terms and edges their inbetween links.) 101
3-5 Concept hierarchy profile representation 103
3-6 Hierarchical network for topics R6 and R21 105
3-7 Hierarchical network for topics R6 and R21 105
3-8 Link distributions for different link weight thresholds 108

4-1 Layering of hierarchical profile representation 114
4-2 Experimental Results for (a)IG and (b)RelDF. 117
4-3 Hierarchical Profile Representing Two Topics of Interest. . . . 121
4-4 Activated Profile. 122
4-5 Experimental Results for (a)IG and (b)RelDF. 127
4-6 (a)IG and (b)RelDF for topics R1/R2(I) 134
4-7 (a)IG and (b)RelDF for topics R7/R8(II) 134
4-8 (a)IG and (b)RelDF and topics R29/R68(III) 134
4-9 (a)IG and (b)RelDF for topics R10/R32(IV) 135
4-10 (a)IG and (b)RelDF and topics R6/R21(V) 135
4-11 (a)IG and (b)RelDF for topics R41/R79(VI) 135
4-12 Average number of profile terms and links. 137
4-13 Average link weight. 142
4-14 (a)IG and (b)RelDF for topics R1/R2/R3(I') 144
4-15 (a)IG and (b)RelDF for topics R7/R8/R9(II') 144
4-16 (a)IG and (b)RelDF and topics R1/R29/R68(III') 144
4-17 (a)IG and (b)RelDF for topics R10/R32/R50(IV') 145
4-18 (a)IG and (b)RelDF and topics R6/R20/R21(V') 145
4-19 (a)IG and (b)RelDF for topics R41/R58/R79(VI') 145

4-20 Average number of terms and links. 146

5-1 Step 1: Term weighting and extraction. 171
5-2 Step 2: Redistribution of term weights. 174
5-3 Step 3: Removal of incompetent terms. 175
5-4 Step 4: Adding new profile terms. 176
5-5 Step 5: Link generation and weighting 178
5-6 The effect of adaptation on the profile 180
5-7 Results for for task α.1 188
5-8 Results for for task α.2 188
5-9 Results for for task α.3 188
5-10 Results for for task α.4 189
5-11 Results for for task α.5 189
5-12 Results for for task α.6 189
5-13 Profile statistics for task α.1 190
5-14 Profile statistics for task α.2 190
5-15 Profile statistics for task α.3 190
5-16 Profile statistics for task α.4 191
5-17 Profile statistics for task α.5 191
5-18 Profile statistics for task α.6 191
5-19 Score fluctuation for task β.1 194
5-20 Score fluctuation for task β.2 194
5-21 Score fluctuation for task β.3 194
5-22 Score fluctuation for task β.4 195
5-23 Score fluctuation for task β.5 195
5-24 Score fluctuation for task β.6 195
5-25 Profile statistics for task β.1 198
5-26 Profile statistics for task β.2 198
5-27 Profile statistics for task β.3 198
5-28 Profile statistics for task β.4 199
5-29 Profile statistics for task β.5 199
5-30 Profile statistics for task β.6 199
5-31 Score fluctuation for task γ.1 202
5-32 Score fluctuation for task γ.2 202
5-33 Score fluctuation for task γ.3 202
5-34 Score fluctuation for task γ.4 203
5-35 Score fluctuation for task γ.5 203
5-36 Score fluctuation for task γ.6 203
5-37 Profile statistics for task γ.1 206
5-38 Profile statistics for task γ.2 206
5-39 Profile statistics for task γ.3 206

5-40 Profile statistics for task γ.4 207
5-41 Profile statistics for task γ.5 207
5-42 Profile statistics for task γ.6 207
5-43 Score fluctuation for task δ.1 210
5-44 Score fluctuation for task δ.2 210
5-45 Score fluctuation for task δ.3 210
5-46 Score fluctuation for task δ.4 211
5-47 Score fluctuation for task δ.5 211
5-48 Score fluctuation for task δ.6 211
5-49 Profile statistics for task δ.1 212
5-50 Profile statistics for task δ.2 212
5-51 Profile statistics for task δ.3 212
5-52 Profile statistics for task δ.4 213
5-53 Profile statistics for task δ.5 213
5-54 Profile statistics for task δ.6 213

Preface

I would like to express my gratitude towards my thesis supervisors, Dr. Victoria Uren for our daily brainstorming minutes, Pr. Anne de Roeck for her generous support and Dr. John Domingue, for his critical advices and comments. They accepted to start with me this long scientific journey, and it's with their precious collaboration that this work comes to its term. I would also like to thank my ex supervisors, Dr. Stuart Watt and Pr. Enrico Motta, for our collaboration, and also Pr. Rob Gaizauskas and Pr. Darrel Ince for accepting to read my work and be examiners.

I will never forget the catalytic and inspiring role of my friend Dimitris Vyzovitis through out these years and those every day moments with Dnyanesh Rajpathack, Bertrand Sereno, Gangmin Li and other colleagues and friends. Special thanks to the "Sofa People": Alex the Guru, Arun Saldana Pie, Ed Head, Niall Paterson, and last but not least Boat Mark, for all the MK sessions.

I thank my family and friends back in Greece for putting up with my absence. My special thanks go to the Nootropia's godfather, Mr. Kostas Xatzis, and to Lefter and Evi for their every day support. Finally I would like to express all my gratefulness to my parents. Without their support this work would have never been accomplished.

Nikolaos Nanas
November 28, 2003

Chapter 1

Introduction

In recent years, advances in digital media, network and computing technologies have caused an exponential growth of the digital information space that is accessible to individuals. Digital devices like cameras, and microphones, capture information about our physical world. Computing applications produce, or help us to produce, additional digital information. A lot of what is captured or produced can be stored in storage devices with increasing capacities. Networking allows the flow and further reproduction of information. Finally, personal devices like TVs, PCs, PDAs and cell phones provide us with access to what is available online, through various information networks like the Internet.

At the same time, we are practically and physically limited in the amount of information that we may perceive. Hence we are faced with the cumbersome task of selecting out of the glut of accessible information, information items that comply with our requirements, i.e. "relevant information". This is the problem that is usually referred to as "Information Overload" [102].

The problem of information overload has forced us to reach for tools that assist an individual with the above cumbersome task. This need for "Personalised Information Delivery" (PID) was recognised as early as 1958 when Luhn coined the term "Selective Dissemination of Information" [101]. Since then the need has been increased by the vertiginous growth of the accessible information space, resulting in intense research activity and commercial interest. Various research disciplines have emerged to tackle different aspects of the problem. Research in Information Retrieval (IR) enabled us to actively search for relevant information using queries. Text Categorisation (TC) allowed the automatic organisation of information into thematic categories and hence facilitated the above search. However, although research in these fields has been fundamental for tackling the PID problem, we will see that Information Filtering adds new dimensions to the problem and points towards alternative scientific directions.

Information Filtering (IF) depends on the ability to recognise an individual's interests and adapt to changes in them; it depends on a *profile*, a model of the individual's preferences. As we will see, on the basis of an individual's profile, an IF system can tackle most PID aspects. But so far this ability to "read someone's mind" has only been a distinguishing characteristic of the most intelligent beings on this planet, us and our close evolutionary cousins. We still don't know the exact mechanisms behind this behaviour, but we do know that "it is conjured up by the local, feedback-heavy interactions of unwitting agents, by the complex adaptive system that we call the human mind" [75]. In his book "Emergence", Steven Johnson goes on to argue that:

> *"Amazingly, this process has come full circle. Hundreds of thousands–if not millions–of years ago, our brains developed a feedback mechanism that enabled them to construct theories of other minds. Today, we are beginning to create software applications that are capable of developing a theory of our minds. All*

> *those fluid, self-organizing programs tracking our tastes and interests, and measuring them against the behavior of larger populations– these programs are the beginning of a progression that will, in a matter of years, lead to a world where we regularly interact with media that seems to know us in some fundamental way. Software will recognise our habits, anticipate our needs, adapt to our changing moods. The first generation of emergent software displayed a captivatingly organic quality; they seemed more like life-forms than the sterile instruction sets and command lines of early code. The next generation will take that organic feel one step further: the new software will use the tools of self-organisation to build models of our own mental states. These programs won't be self-aware, and they won't pass any Turing tests, but they will make the media experience we've grown accustomed to seem autistic in comparison. They will be mind readers."*

This passage encapsulates in the best possible way our motivations. The practical implications of PID and personalisation in general are extensive. According to Johnson, computer games that provide a personalised experience, music recommendation systems and personalised newspapers are only the start. Personalisation is already changing the traditional laws of advertisement and media distribution and with the increase in bandwidth it will soon bring a genuine revolution in what it means to be a media consumer. But at the same time, complex adaptive systems have triggered a revolution themselves. Their emergent, self-organising qualities change the way we conduct business, we think about democracy and add a new perspective to the quest for Artificial Intelligence. As Johnson puts it:

> *"A few decades from now, the forces unleashed by the bottom-*

> *up revolution may well dictate that we redefine intelligence itself, as computers begin to convincingly simulate the human capacity for open-ended learning. But in the next five years alone, we'll have plenty of changes to keep us busy. Our computers and television sets and refrigerators won't be thinking themselves, but they'll have a pretty good idea what we're thinking about."*

Within this context, our PhD work focused on adaptive document filtering. We developed *Nootropia*[1], an experimental IF system that uses a single profile to recognize a user's interests and adapt to changes in them. We concentrated on profile representation, document evaluation and profile adaptation, but also suggest how Nootropia may support most aspects of PID and other personalisation services.

In particular, in the next chapter we review the state-of-the-art in PID to set the foundations of our research and more importantly to identify directions that research in IF left unexplored. We argue that it traditionally inherits the incorrect term independence assumption from IR and TC: profile representations ignore correlations between terms in text. Such profiles support linear document evaluation functions and consequently, they can only effectively represent a single topic of interest. Furthermore, they are usually coupled with linear learning algorithms that can only achieve a steady adaptation pace. These practices we argue, impose a reductionist approach to profile representation and adaptation that is not well suited to the dynamic user interests. A user may be interested in more than one topic in parallel and these multiple interests change over time.

[1] Greek word for: "An individual's or a group's particular way of thinking, someone's characteristics of intellect and perception"

Can multiple user interests and various changes in them be represented with a single profile?

Our innovative approach to this problem is founded on non-linearity and self-organisation. In chapter 3 we describe a methodology for extracting a hierarchical term network out of a set of user-specified documents reflecting the user's current interests. This hierarchical profile representation tackles all three dependence dimensions that Lauren B. Doyle identified as early as 1962 [48]. The hierarchical term network comprises the most informative terms in the specified documents, measures their topical and lexical correlations in text and distinguishes topic-subtopic relations between them. The net-effect is that the hierarchical profile can represent and distinguish between multiple topics of interests.

For the profile to be used computationally for document filtering, we introduce in chapter 4 a directed spreading activation model to establish on the hierarchical term network a series of non-linear document evaluation functions. In this way, we achieve document evaluation according to multiple topics of interest.

To adapt this single, multi-topic profile to the changing user interests, we have been inspired by biological theories of self-organisation. We present in chapter 5 a process that allows the hierarchical profile to structurally self-organise in response to changes in user feedback. As a result the profile appears to adapt to a variety of changes in the user interests ranging from frequent local variations to the emergence of a new topic of interests and the loss of interest in a certain topic.

The above three components, hierarchical, multi-topic profile representation, non-linear document evaluation and profile adaptation through self-organisation, constitute Nootropia's *Adaptive Document Filtering* core. They are evaluated experimentally using appropriate variations of TREC-2001 routing subtask with positive results. Although it has not been our main research goal, our experimental methodology suggests possible directions towards the establishment of a new evaluation standard: one that does not exclude multi-topic profile representations and that can test the ability of a profile to adapt to both radical and modest drifts in the user's interests.

But our steps towards Nootropia do not stop at the essential adaptive document filtering core. In chapter 4 we suggest ways of using the profile to achieve enhanced representation of the filtering results and to support other personalisation services. Furthermore, although our PhD work has focused on textual information, it is in principle applicable to other media, like audio and image, for which descriptive features can be automatically extracted. These are all interesting directions for future research. In chapter 6, we summarise what has already been done and how it contributes to the domain of adaptive document filtering.

Chapter 2

Literature Review

> "The visibility of personal computers, individual workstations, and local area networks has focused most of the attention on *generating* information – the process of producing documents and disseminating them. It is now time to focus more attention on *receiving* information – the process of controlling and filtering information that reaches the persons who must use it."
>
> Denning, 1982

In the quest for Personalised Information Delivery (PID) various research disciplines have emerged, giving birth to a plethora of techniques, algorithms and experimental systems. In parallel, numerous commercial systems have hit the market, mainly in support of internet activities like e-commerce. Recently, reviews of the state-of-the-art have been presented based on general dimensions that allow a coherent description and/or classification of existing approaches and systems [98, 80, 134, 133, 1, 140, 119, 62]. These reviews attempt to reveal both the commonalities and the differences between current approaches. It has been recognised that at the core of any approach to personalised information delivery lies a tailored representation of an individual's interests called the "user profile". The user profile can be used both for retrieving information and for evaluating the retrieved or received information items, according to the individual's interests. The output of the evaluation allows the selection and presentation of the information items in such a way that the relevance of what is finally seen by the individual is increased. However, despite this common fundamental component to PID, existing approaches specialise in specific problem instantiations, in order to achieve optimised solutions [98]. Even though, existing reviews of the state-of-the-art reflect the division in approaches, they do not provide a critical evaluation that can lead to a more global perspective.

This study attempts to highlight the issues that have to be resolved concerning PID and IF in particular. It is structured around a problem model that enables a unifying perspective towards existing practices. The goal however is not a mere description or classification of existing systems, that is bound to be out-of-date due to the accelerating pace of development in the field. Extensive accounts of existing systems can be found in [1, 140]. A large number of systems is nevertheless used to provide evidence for the argumentation, as in the cases of [119, 62]. The focus will be on the issues that arise along the model's dimensions and in particular on those that relate to user profiling and its adaptation to changes in the individual's information interests. Although, these are fundamental aspects of IF, significant space for improvement will be identified.

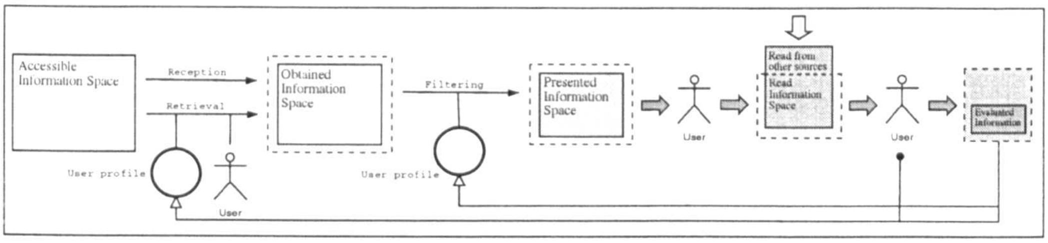

Figure 2-1: Personalised Information Delivery

2.1 Personalised Information Delivery

To provide an integrated presentation of the research domains that relate to PID we use a model that extends those in [98, 134, 133]. According to the model, PID tackles the problem of information overload with a sequence of "focusing" processes that increase the manageability and relevance of what is finally seen by the user (fig. 2.1). Initially, out of the *accessible information space* we can select between sources of information and exploit their internal organisation. Through selective reception or retrieval, we may focus on a subspace of what is accessible, the *obtained information space*, that is more likely to contain relevant information items. It is however still impossible or at least uneconomically time consuming for an individual to go through the obtained information items in search for relevant information. This process can be automated using a user profile to evaluate the retrieved or received information items. The evaluation's output can then be exploited to appropriately present the information items to the individual. So far the model resembles the one adopted by Oard in [134, 133], where a series of three processes: collection, selection and display, is used.

Clearly, the user profile is fundamental for automating the above focusing processes. An accurate enough representation of the user's interests is therefore required. To build such a representation we need some information about what is of interest to the user, or in other words *relevance informa-*

tion. It is usually defined in terms of user feedback on what has already been assimilated. More specifically, the user sees information that has either been presented by the PID system or through some other route (fig. 2.1). In that sense, a PID system is not approached as an interface to accessible information, but rather as a "personal information assistant" or "agent" [102]. The user-provided relevance information can be used for building the initial profile and successively for updating it. Based on additional relevance information the initial profile can adapt to changes in the user's information requirements. Such changes are implied by changes in the user's knowledge and environment. They are reflected in what the user considers to be relevant and hence in the relevance information that is provided. The interaction with the user is therefore a significant part of PID, and so appropriately included in the model (fig. 2.1).

In the rest of this chapter we discuss the corresponding theories, models and techniques based on the above model. Starting with the accessible information space, we follow the information in its journey towards the user and finally its use for adapting the user profile. The information spaces provide the links between the necessary focusing processes in a way similar to the use of buffers in [98]. The first part of the review focuses on IR and TC. It sets the foundations for discussing IF in the rest of the chapter. We elaborate on profile representation, initialisation, document evaluation and profile adaptation and we finally conclude with a discussion of methodologies for the evaluation of IF systems, a subject external to the PID model, but equally important.

2.2 The Accessible Information Space

PCs, PDAs and mobiles provide us with access to a vast and growing information space distributed over various sources. One can distinguish between *dynamic* and *static* information sources, based on the lifetime of the information items, i.e. the decay in the value of an item's information content in relation to time [98]. On the one side of the spectrum information items like stock market values exhibit short lifetimes while on the other side an example could be some seminal scientific paper that is relatively persistent. The spectrum is continuous, and the actual lifetime depends on an item's usage. Stock market values for example could be used retrospectively for predictions. In practice however, a specific distinction between static and dynamic information sources is the amount of preprocessing that the lifetime of their information items allows.

2.2.1 Dynamic Information Sources

In the case of dynamic information sources the information items have to reach their destination as soon as possible. There is no time to organise the information space beyond a high level. Another reason that hinders further organisation is the large volume of information that such sources typically produce [16]. Dynamic sources can be thought of as broadcasters. Users that have tuned their applications to an information source or some of its channels, receive everything that is transmitted by the source, through the specified channels. The side-effect is that there is no prior meta-information about an information item's actual content, that may relate it to the content of the rest of the items in the source. An item's relevance can only be assessed after it has actually been obtained. So despite the ability to select the information source or some of its channels, the user is usually confronted with a large number of information items to review in order to find what is of relevance. Examples of dynamic information sources include internet newswires like Usenet, mailing lists, online newspapers and even email in general.

Another information source that is appropriate to be classified as dynamic, is other users. Network technology has not only connected people to online information sources but also to other people [79]. Human experts can be a valuable source of information, but finding the expert or experts that are likely to provide the required information is not straightforward.

2.2.2 Static Information Sources

Static information sources on the other hand contain information items with a longer lifetime. Instead of broadcasting new information items on the fly, static sources maintain the information items for future access. It is therefore possible to appropriately structure static information sources and also to provide surrogates of the content of individual information items [16]. Of course, the organisation of information far preceded the digital information era. Libraries, like the Library of Congress, organise information items, in this case mainly books, into manually constructed subject hierarchies that define thematic categories and their hierarchical relationships. Subject hierarchies provide an intuitive overview of the topic structure of static information sources [163]. Professional indexers are usually employed to populate the subject hierarchy, i.e. to assign information items to the categories that the hierarchy defines. An information item's surrogate in this case consists of the identifiers of the categories that the item is assigned to. The hierarchy allows users to browse the information space while the indexing of information items enables their retrieval [129]. In general, the organisation of static information sources provides evidence of an information item's content, prior to obtaining the actual item.

The above practices have been applied to the way digital information is organised. Manually constructed and populated subject hierarchies have been used for example by the ACM Digital Library[1] and the Reuters news archive [152]. But the accelerating increase in digital information renders its manual organisation ineffective [199]. Various disadvantages of manual indexing have been identified [34], while manually constructed hierarchies are usually too general to facilitate the search for relevant information [93].

The solution is to automate or at least facilitate the organisation of static information sources. This requires the ability to represent information content in a form amenable to processing by computers. Abstract representations of information content can be constructed as combinations of appropriate descriptive features. In the case of textual information, textual features like words or phrases, are a natural choice. Automatic feature extraction is not as straightforward for audio and visual information. Nevertheless, recent advances indicate that it is indeed possible for both kinds of media [186, 69, 207, 182]. So, although in the rest of this thesis the focus will be on textual information this possibility is not going to be neglected. In the following sections we initially discuss those characteristics of language that invite the application of statistical processing. We then elaborate on statistical techniques for the construction of content representations and their application for the automated organisation of static collections of documents, and especially the retrieval and categorisation of documents from such sources.

[1] see: http://www.acm.org/class/1998/overview.html

2.3 The Statistical Regularities of Language

Every human language can express a vast variety of meanings and ideas with a limited repertoire of words. This is achieved by appropriate combinations of words to form units of different semantic levels. These units range from compound terms, phrases and in written language, paragraphs, documents and sets of documents. The ability to communicate, which can be grounded on Shannon's information theory [169], implies the emergence of patterns of word usage and not an arbitrary combination of words. Syntactic rules are an example of such patterns [104]. As a result, statistical regularities can be observed in the usage of words in language which are also reflected in textual information. One example is Zipf's law, which states that the frequency of a word decays as a power function of its frequency rank [208]. Statistical correlations exist between terms in text and between terms and larger semantic units like documents or document classes. Doyle has argued that these correlations are "a natural consequence of the way people think and communicate" [48]. He has identified two basic phenomena that cause statistical dependencies between terms. *Language redundancy* refers to the habitual use of lexical compounds as semantic units. *Reality redundancy* on the other hand relates to the pattern of reference to various aspects of the topic which is being discussed. In the rest of this thesis, we will refer to the term correlations caused by these two phenomena as *lexical correlations* and *topical correlations* respectively. The two phenomena are not distinct; lexical compounds can be formed as part of a topic's discourse and in that sense, they can comprise terms which are topically correlated. Nevertheless, a distinction can be made between terms that usually appear close to each other and terms that appear together frequently in the context of a broader semantic unit (e.g. sentence or paragraph). Finally, Doyle attributes a third phenomenon called *documentation redundancy*, to document series like progress reports and newsletters, that repeatedly and periodically refer to the same spectrum of topics. We will adopt the same term to more generally refer to those language characteristics that cause correlations between terms and documents or document classes.

Recently, language has been treated as a network of terms (nodes) that are linked to related terms. Increased interest to such complex networks has been triggered by the groundbreaking work of Watts and Strogatz [188] and of Barabási [13]. They have demonstrated that many biological, technological and social networks exhibit common important statistical characteristics. According to [67, 126], language networks of the above type share these characteristics. These observations do not only "reflect the evolutionary and social history of lexicons and the origins of their flexibility and combinatorial nature" [67], but also their importance for the study of language and cognitive science [126]. Network representations of textual information will be of particular interest henceforth.

In general, the statistical regularities of language make it possible for machine readable representations of textual information to be automatically derived. *Information Retrieval* (IR) and *Text Categorisation* (TC) are two very well established disciplines that exploit this ability. In particular, IR research focuses on the development of algorithms and models for the retrieval of documents from static collections [159]. TC on the other hand, is concerned with the problem of automatically assigning a class label or subject descriptor to documents that belong to the same topic [122]. TC facilitates the indexing of documents according to a set of pre-defined and relatively static topic categories and therefore the subsequent search for relevant information. Despite this difference, IR and TC share the same three, higher level components:

a) a representation of each document's content (sec. 2.4),

b) a representation of the topic (or class) of interest (sec. 2.5) and

c) a way of comparing the previous two (sec. 2.6).

These three components are fundamental to personalised information delivery and so it is appropriate to dwell on how they are instantiated in the context of IR and TC.

2.4 Document Indexing

The goal of document indexing is to produce a set of features that represent the content or topics of a document [34]. Typically a document can be treated:

a) as the set of letters that appear in the document, in the order they appear;

b) as the set of unique terms[2] that appear in the document; and

c) as the set of terms that appear in the document in the order that they appear.

In the first case, if n is an integer number, then a document is treated as the set of all possible sequences of n letters, n-grams, that can be constructed by the letters in the document, in the order that they appear [104]. The document can be then represented as a vector in the space of all possible n-grams. The dimensionality of the space increases exponentially with n and

[2] In the rest of this thesis a "term" is considered to be a single word.

so usually no more than 3 letters are used [178]. This kind of representation excludes term semantics.

In the second case, each document is treated as a "bag of words". The order in which terms appear in the document is not taken into account. If the document is part of a collection, then it can be represented as a vector in a t-dimensional space, where t is the number of unique terms in the collection. This is the essence of the *Vector Space Model* [159]. According to the vector space model, absence of a term is indicated by 0 while presence of a term is indicated either by 1 (binary vector) or a numerical weight (weighted vector). Vector representations of documents have been widely used in IR and Text Categorisation. *Term Weighting* has been shown to produce weighted vectors that improve retrieval performance over binary vectors [155, 161] and helps reduce the problematic high dimensionality of the native feature space [203]. Even moderate-sized text collections can include tens or hundreds of thousands of unique terms. Term weighting is an important technique for identifying the most informative words to be used for document indexing, which we will discuss further in the next section (sec. 2.4.1).

Dimensionality reduction can also be achieved via *stop word removal* and *stemming*. In stop word removal, non-informative terms from a stoplist of grammatical or function words like *the*, *is* and *for*, are excluded from consideration during document indexing. Although in [145], it has been argued that such function words could be used to create more effective indexing terms, stop word removal has been a common practice. Stop-lists are usually constructed manually, but the automatic generation of domain specific stoplists has also been proposed [204]. Another strategy is to use stemming algorithms, like those developed by Porter [138] and Lovins [100], which truncate words into their stems. For example, words like *laughing* and *laughter* are both truncated into their common stem *laugh-*. Church has studied the use of stemming and although he agrees with previous experimental results showing that stemming does not affect retrieval performance (at least in 'English'), he concludes that the use of stemming is justified for informative terms [35].

A fundamental assumption of the vector space model is that document terms are stochastically independent - terms are not recognised as being related to each other [108]. This assumption, although demonstrably false, facilitates the use of certain IR models by minimising the number of parameters that have to be estimated [99]. In reality, lexical and topical correlations cause stochastic dependencies between terms in documents. Challenging the term independence assumption is a major theme of this thesis.

Latent Semantic Indexing uses *singular value decomposition* that exploits the stochastic dependencies to project the initial feature space to a "latent" semantic space with far fewer dimensions [45]. Lexical correlations can be captured if phrases are used instead of single terms for document indexing. This requires that the order of terms in the text is taken into account. In this third case of document analysis, phrases can be identified using, for example, statistical and/or syntactical parsers [51, 95]. Their respective merits in terms of retrieval are controversial. In general, indexing methods that are based on syntactic components have not been proved to be more effective in terms of retrieval performance [156]. One reason could be that the produced index phrases are fixed and have to be used self-same for retrieval [104].

The latter is a general problem with content-based indexing: a document can only be retrieved on the basis of the assigned features (terms or phrases). This can negatively affect retrieval performance due to the *vocabulary problem* [57]. In particular, a word can have multiple meanings depending on its context (*polysemy*) while different words can refer to the same concept (*synonymy*). The problem can be alleviated using *thesauri* that express term relations. In addition to the terms included in a document, related terms can thus be identified and used for its indexing. Traditionally, thesauri have been constructed manually. The most ambitious project is WordNet [113]. But manually constructed thesauri are expensive to build and maintain. Automatic thesauri construction attempts to tackle this problem by extracting term relations mechanically [135, 31, 34, 41, 14].

The vector space model ignores topical correlations as well. The assumption is often made in IR that the documents of interest are part of a single, homogeneous and unstructured collection [176]. An *inverted index* is usually employed that just lists for each word in the collection all documents that contain it. In contrast, recent research has focused on the automatic extraction of concept hierarchies from document sets [163, 5]. The extracted hierarchies express topic-subtopic relations between terms and can be used in a way similar to manually constructed subject hierarchies for automatic indexing, multi-document summarisation and interactive access to information (sec. 2.2.2).

Other approaches to automatic document indexing include the use of *inference networks* [183, 181, 28], *neural networks* [199, 88] and *semantic networks* [15, 40]. According to these connectionist approaches, documents and index terms are represented by nodes. Terms that are contained in a document are linked to the corresponding node. Links between terms and between documents are not used and therefore term correlations are ignored. As we will discuss in more detail in section 2.6, connectionist approaches to document indexing can support flexible retrieval strategies. Nevertheless, as has been noted by Kwok [88], more interesting representations can be formulated if term correlations are explicitly taken into account. In section 2.7 we discuss in detail how lexical correlations and topical correlations, or the stochastic term dependencies that they cause, are currently represented by networks, whereas in the case of the networks described in section 2.3, terms are represented as nodes and their correlations as links.

2.4.1 Term Weighting for Document Indexing

Term weighting has been an essential technique for document indexing. The goal is to weight the unique terms in a document and then, based on the assigned weights, select the most informative terms for representing the document's content. A representation of a document's content that reflects accurately and in depth its various topics is characterised *exhaustive*. On the other hand, *term specificity* relates to the level of detail at which an individual term represents a given topic [161, 175]. One of the primary objectives of term weighting is to tackle the trade-off between exhaustive document representations and the specificity of index terms [63].

These qualitative measures are usually quantified on the basis of three statistics:

Term frequency (*tf*) is the number of times a term appears in an individual document. Luhn was first to recognise that *tf* furnishes a useful measurement of a term's ability to describe a document's content [101]. The underlying idea is that a term that relates to a document's topic will appear more frequently in that document than most non-related terms.

Document Frequency (*df*) is the number of documents in the collection that contain a term. It reflects the distribution of terms within the collection, which provides evidence of their specificity.

Collection Frequency (*cf*) is the number of times a term appears in the complete collection. Its importance is based on the observation that very frequent terms are more likely to be function words while very rare terms are of limited retrieval importance.

Numerous term weighting methods have been introduced that use the above statistics or combinations of them. Depending on which statistics are being used a distinction can be made between methods that assign *document-specific* weights and those that result in *collection-wide* term weights. The following paragraphs present some of the most well established term weighting methods that are used for document indexing. A more extensive review can be found in [77].

Term Frequency (TF)

The frequency tf_t of a term t, can be used as the term's document-specific weight. Typically, logarithmic smoothing is applied to dampen the effect of *tf* on the term's weight [63] (equation 2.1). Although, *tf* is a measure of a term's significance within a document, it does not bear any information about the term's specificity. For this, a term's collection statistics are required. Nevertheless, since *tf* is not dependent on the existence of collection statistics, it can be readily applied for the weighting of terms in documents that have been received from dynamic information sources. Another drawback of *tf* is that its absolute value depends on the total number of terms in the document. If comparison between the term weights of different documents is required then *tf* is usually normalised to the number of terms in the document or to the maximum number of times a term appears in the document [122]. Such normalisation however, can have a negative effect on retrieval performance [172].

$$TF_t = log_2 tf_t \tag{2.1}$$

Relative Frequency Technique (RFT)

RFT has been suggested by Edmundson and Wyllys [49]. Based on the assumption that special or technical words are more rare in general usage than in documents about the corresponding subjects, they introduced four different ways for assessing the relative frequency of a term within a document and a general corpus. In contrast to pure TF, RFT exploits a term's corpus statistics. As a result, this document-specific weight incorporates implicit information about a term's specificity. Although, Doyle suggests RFT as a solution to the documentation redundancy phenomenon [48], the technique has not been widely adopted. Nevertheless, we present here the first of the four proposed measures, because it has been influential in the development of our new term weighting method (sec. 3.2.2). More specifically, if T_d is the total number of terms in the document and T_c the number of terms in a general document collection then equation 2.2 represents the first RFT measure.

$$RFT_t = \frac{tf_t}{T_d} - \frac{cf_t}{T_c} \tag{2.2}$$

Inverse Document Frequency (IDF)

IDF uses *df* to estimate the specificity of terms in a document collection. It is based on the idea that a semantically focused word will appear in only a few documents, while a semantically unfocused word is spread out homogeneously over the collection. With N being the total number of documents in the collection, Sparck Jones proposed equation 2.3 as a way of calculating the IDF weight of a term t and showed that its usage significantly improves retrieval performance compared to unweighted retrieval [175]. Further studies have confirmed this finding [206, 63]. IDF is a collection-wide weighting method that assigns a single weight to each term in the collection irrespective of document. One of its advantages is that it can be updated online if new documents are added to the static information source [187].

$$IDF_t = log_2 \frac{N}{df_t} \tag{2.3}$$

Term Frequency Inverse Document Frequency (TFIDF)

Term frequency can be used to identify terms that collectively provide an exhaustive description of a document's content. DF on the other hand is a measure of a term's specificity. To tackle the trade-off between exhaustivity and specificity combinations of these statistics have been proposed [63]. The most widely adopted and documented combination is TFIDF [175]. TFIDF assigns a document-specific weight to a term t with term frequency tf_t, usually according to equation 2.4. Different variants of TFIDF, which incorporate normalisation and/or logarithmic smoothing of the effect of the two parameters (tf and df), have been presented in [157]. TFIDF has been shown to have a positive effect on retrieval performance [39]. However, one major problem with TFIDF is its batch nature. Document term weights have to be recalculated every time new documents are added to the collection [187].

$$TFIDF_t = tf_t \times log_2 \frac{N}{df_t} \tag{2.4}$$

Residual Inverse Document Frequency (RIDF)

RIDF is a variation of IDF that assigns collection-specific weights to terms according to the difference between the logs of the actual IDF and its prediction by a Poisson model [35]. According to Manning and Schütze [104], the most common way of calculating the RIDF of terms is given by equation 2.5, where $\lambda_t = cf_t/N$ is the average number of occurrences of term t per document and $1 - p(0; \lambda_t)$ is the Poisson probability that t appears at least once in a document.

$$RIDF_t = idf_t + log_2(1 - p(0; \lambda_t)) \tag{2.5}$$

The Poisson distribution has also been the base of the 2-Poisson weighting schema that is described in [21, 64, 65, 22]. The latter has itself been the basis for other term weighting methods [150]. Other term weighting methods used for document indexing include, *term strength* [204], *term precision* [160, 205] and *term discrimination value* [162].

2.5 Topic Representation

We have already discussed how representations of document content can be automatically extracted. However, the automation of processes like document retrieval and categorisation requires the ability to represent textual content that relates to a topic of interest. Although document representation is usually common to both IR and TC, there are significant differences in the way the topic of interest is represented in these two domains. These we explore in the following sections.

2.5.1 Topic Representation by Query

The classic problem in IR is *ad-hoc* retrieval. In ad-hoc retrieval an anomalous state of knowledge or short-term information need prompts the user to engage in active information-seeking behaviour [16]. The user identifies the particular information need in terms of a request consisting of one or several free language terms, i.e. a *query*. Boolean operators between the query terms can also be used (*boolean query*). The query is then used to efficiently retrieve documents from a static collection based on the corresponding inverted index.

The query formulation process is essentially analogous to document indexing. In the same way that terms are selected, manually or automatically, to describe a document's topics, the user specifies terms that describe the current topic of interest [105]. Both representations can be expressed in the context of the vector space model as multidimensional vectors. The basic difference is that while document indexing results in fairly permanent representations of documents, a query is a temporary content representation that is discarded after the end of the information-seeking episode. It has also been acknowledged that for cognitive reasons a query is an imperfect representation of the actual information need [118]. In addition and as in the case of document indexing, practical problems arise due to the vocabulary problem. A query can only retrieve documents that are indexed by the query terms and not their potential synonyms.

This problem can be alleviated with *query expansion*. As in the case of document indexing, thesauri can be employed to expand the initial query terms with associated terms [135, 30]. Query expansion by synonyms of the query terms has been shown to increase retrieval performance [158]. However, although document indexing employs global thesauri that represent the relations between terms in the complete collection vocabulary, it has been shown that thesauri based on local analysis of documents is advantageous for query expansion [201]. The latter local thesauri are produced from analysis of documents that the user has deemed relevant to the current query. In a similar approach to query expansion, these documents are used to generate a *dependence tree* that expresses the stochastic correlations between terms [18].

Query enhancement can also be achieved if weights are assigned to query terms [149]. The weights reflect the association between the query terms and the topic of interest. Information about the relevance of retrieved documents, i.e. *relevance information*, is therefore required. The result is a weighted vector representation of the topic of interest. Query term weighting is further discussed in the next section.

So far, we have described user formulated queries of free language terms. The focus on terms is strongly related to the dominance of the vector space model and of inverted indexes. Amongst others, connectionist approaches to document indexing can allow the use of a complete indexed document for the retrieval of related documents [15, 88]. Although this kind of *document-based* retrieval is limited to documents that are already part of the indexed collection, it overcomes to some extent the above query disadvantages. A document can provide more information about what the user is looking for than a set of query terms. Finally, the dependence on user specified queries has been implied by the fact that IR is mainly concerned with short-term information needs. No information about what the user is looking for is available prior to the submission of the query and none remains after the completion of the information-seeking episode. Any automation of the retrieval process however, would require automatic, or at least semi-automatic, query formulation. We will come back to document-based retrieval and automatic query formulation later on.

2.5.2 Topic Representation by Classifier

In contrast to IR, which is concerned with short-term information needs, TC tackles the problem of assigning documents to one or more of a number of predefined and fixed topic categories. This requirement has two serious implications that distinguish TC from IR. Firstly, there is usually extensive relevance information in the form of a large set of training documents that have been pre-classified according to the existing categorisation schema [122]. Secondly, in contrast to IR, topic categories in TC are fairly static.

Based on the training documents, machine learning algorithms can automatically generate an elaborate and relatively permanent representation, a *classifier* for each topic of interest [167]. We'll come back to machine learning algorithms in section 2.13. A variety of classifiers have been applied and evaluated, like decision trees [96], naive Bayes [96, 131], rule-based [38], nearest-neighbour [203], neural networks [130, 198] and Support Vector Machines [74]. Typically, the topic categories are assumed to be unrelated. Recently, research in TC has focused on ways to exploit the hierarchical relations between topic categories [154, 192]. This trend was triggered by the need to cope with very large document sets. Hierarchical categorisation tackles this problem by decomposing the categorisation, according to the existing hierarchy.

The vector space model is usually adopted to represent both the training documents and those that have to be classified [73, 96]. Machine learning algorithms are therefore confronted with the high dimensionality of the feature space [203]. Although, LSI has been employed for dimensionality reduction [166], the common practice is to weight the terms in the training documents. As in the case of query term weighting, the assigned weights represent the statistical association between the terms and each topic of interest. The next section presents a number of widely adopted term weighting methods that accomplish this task.

Finally, the vector space model imposes the term independence assumption, resulting in a majority of *linear* classifiers. Non-linear classifiers have also been investigated, that either use phrases to index the documents [38], or employ non-linear neural networks [166, 192, 193]. We discuss linear and non-linear classifiers further, in section 2.6.2.

2.5.3 Term Weighting for Topic Representation

Section 2.4.1 summarised a number of term weighting methods that are used for document indexing. These are methods that estimate how closely a term is related to a document's content or how specific it is in regard to the complete document collection. Here we are interested in those term weighting methods that estimate the association between terms and a topic of interest. These methods are based on differences in the distribution of terms between the complete collection, a set of documents that is relevant to the topic of interest, and, in some cases, a set of documents that is non-relevant to that topic. They assign a *topic-specific* weight to terms in the relevant set. The contingency table (table 2.1) summarises a term's distribution within these document sets [185]. In the following paragraphs we use the table's notation to present some of the most well established approaches, that, as already mentioned, have been used for query term weighting and dimensionality reduction in TC.

Table 2.1: Contingency table

		Documents		
		Relevant	Non-relevant	Collection
Term	+	r (A)	$n-r$ (B)	n
	-	$R-r$ (C)	$N-R-n+r$ (D)	$N-n$
		R	$N-R$	N

where

r is the number of relevant documents that contain the term
n is the total number of documents in the collection that contain the term
R is the total number of relevant documents
N is the number of documents in the collection

Query Term Weighting

Weighting of query terms can be accomplished if relevance information is acquired through user feedback on the documents retrieved so far. This implies a broader information seeking episode that comprises more than one query about the same topic of interest. Robertson and Sparck Jones proposed four methods (F1 to F4) for the probabilistic weighting of search terms, based on the *binary independence retrieval model* [149]. The four methods correspond to the combinations between two independence assumptions and two ordering principles. Robertson has emphasised the difference between problems where complete relevance information is available (*retrospective*) and problems where estimations based on incomplete information are required (*predictive*). For predictive problems, he suggested a variation of the simple, retrospective version of the methods. Out of the four proposed methods here we focus on the retrospective version of the first (eq. 2.6) and the predictive version of the fourth (eq. 2.7)[3]. The latter was shown to be the best performing approach. Although Robertson and Sparck Jones report increased retrieval performance over unweighted queries, contradictory results

[3] These are the methods that we evaluate in section 3.3.

have been presented in the case of weighted boolean queries [55].

$$F1_t = log\frac{r/R}{n/N} = log\frac{r}{R} - log\frac{n}{N} \tag{2.6}$$

$$F4_t = log\frac{(r+0.50)/(R-r+0.5)}{(n-r+0.5)/(N-n-R+r+0.5)} \tag{2.7}$$

Relevant Document Frequency (RDF)

RDF is the simplest method that exploits relevance information. Given a set of documents that are relevant to a topic, RDF measures the number of documents that contain the term (eq. 2.8). The assumption here is that those terms that appear in the majority of the documents are more strongly associated to the documents' topic than terms that occur less. However, RDF can mistakenly identify terms that appear in a lot of the documents because they appear frequently in general (e.g. function words).

$$RDF_t = r \tag{2.8}$$

Information Gain (IG)

IG is an information-theoretic metric that measures the expected reduction in entropy of category prediction that is caused by partitioning the relevant documents according to a specific term [117]. IG is a general metric that has also been applied for measuring term correlations [41]. Equation 2.9 defines the binary version (one topic) of the metric [96]. The m-ary version can be found in [203].

$$\begin{aligned} IG_t = & -Pr(rel)logPr(rel) + Pr(t)Pr(rel|t)logPr(rel|t) + \\ & + Pr(\neg t)Pr(rel|\neg t)logPr(rel|\neg t) = \\ & -\frac{R}{N} \cdot log\frac{R}{N} + \frac{r}{N} \cdot log\frac{r}{n} + \frac{R-r}{N} \cdot log\frac{R-r}{N-n} \end{aligned} \tag{2.9}$$

Mutual Information (MI)

MI is another measure derived from information theory. It gauges the reduction in uncertainty of one random variable when we know about another. The metric is commonly applied for identifying term correlations [176, 61, 36, 198]. In a similar way, it can be used for measuring the association between a term and a specific topic of interest (eq. 2.10) [154]. We should note that the equation is identical to F1 (eq. 2.6).

$$F1/MI_t \approx log\frac{A \times N}{(A+C) \times (A+B)} = log\frac{r/R}{n/N} \tag{2.10}$$

χ^2 chi square (CHI)

CHI is similar to MI in that it measures the lack of independence between two variables. It calculates the difference between the observed frequencies in the contingency table and the frequencies expected under the independence assumption. If the difference is large, then we can treat the variables as not independent. For the problem at hand, χ^2 is applied to measure the lack of independence between a term and the user-specified topic of interest (eq. 2.11). A variation of χ^2 has been proposed in [130] and in [132] χ^2 is evaluated in the context of text classification. A more analytical presentation of IG, MI and CHI can be found in [203], while a general review of term weighting methods employed in TC is presented in [167].

$$\begin{aligned}\chi^2 &= \frac{N \cdot (AD - CB)^2}{(A+C)\cdot(B+D)\cdot(A+B)\cdot(C+D)} = \\ &= \frac{N \cdot (rN - nR)^2}{R \cdot n \cdot (N-R) \cdot (N-n)} \qquad (2.11)\end{aligned}$$

2.6 Document Evaluation

There would not be any reason for automatically generating document and topic representations if they were not complemented by appropriate measures for comparing them. The actual similarity measures are dependent, at least to some extent, on the corresponding representations. The following sections discuss how IR and TC tackle document evaluation by query and classifier, respectively.

2.6.1 Document vs. Query

There are two basic approaches for the retrieval of documents. According to the *exact match* principle, all documents that contain the combination of words or phrases in the query are retrieved. No distinction between the retrieved documents is made. On the other hand, the *best match* approach to IR does not only result in the retrieval of documents but also in their ordering according to some estimation of the documents' relevance. This can be achieved by both the vector space and probabilistic retrieval models [159].

In the vector space model both documents and queries can be represented as a vector in a multi-dimensional space. This spatial metaphor triggers the application of trigonometric measures of similarity. Jones and Furnas have performed a coherent geometric analysis of vector-based similarity measures [76]. Their conclusions are valuable but concentrate on how differences in the document representations affect their evaluation by the discussed measures. This is justified in the context of IR where a lot depends on the document indexing process. Symmetrically, here we will concentrate on some of the most well established document evaluation approaches and how changes in the query formulation can affect them. This short analysis will be illuminating for our discussion henceforth.

The most basic, but at the same time fundamental, vector-space similarity measure is the *inner product*. It calculates the relevance $R_{D,Q}$ of a document D to query Q, using equation 2.12, where w_i and dw_i are respectively the weights of a term t_i in the query and in the document and n is the dimensionality of the space. Inner product's obvious linearity lends it the following characteristics. A document's relevance is monotonic to the angle between the query and the document and to the length of the query vector. A query's direction depends on the relative importance of terms within the query and provides an indication of the represented topic. The query's length in turn, indicates the intensity of the represented topic in relation to other vector representations [76]. The first kind of monotonicity is sufficiently explained by Jones and Furnas. For the second case, it is enough to remember that, according to the vector space model, the terms are assumed to be independent. The space's dimensions are therefore orthogonal. So, everything else being the same, we can indefinitely increase a document's score by either increasing the weight of a query term or the cardinality of the set of common terms between the query and the document. In other words the inner product is monotonic along any of the space's dimensions which means that theoretically it does not have an upper bound.

$$R_{D,Q} = \sum_{i=1}^{n} w_i * dw_i \tag{2.12}$$

It is this latter characteristic of inner product that prompted the adoption of the *cosine similarity* (equation 2.13). It is easy to recognise that its numerator is actually the inner product. The critical difference is that both the query and document vectors are now normalised so that they have a length of one. The effect of this normalisation is that the document's relevance is now monotonic only to the angle between the two vectors. Any document with a vector representation that shares the same direction with the query vector, receives a maximum relevance of one. So there exists an upper bound that corresponds to more than one document. The weight of individual query terms or the cardinality of the common term set does not affect the document's relevance. Whether this is a beneficial characteristic of the cosine similarity measure is going to be questioned. The inner product measure has been the basis for a number of other measures that are analytically presented in [76].

$$R_{P,D} = \frac{\sum_{i=1}^{n} w_i * dw_i}{\sqrt{\sum_{i=1}^{n}(w_i)^2} * \sqrt{\sum_{i=1}^{n}(dw_i)^2}} \tag{2.13}$$

Probabilistic IR models are based on the *probability ranking principle*, which states that given the available evidence, an IR system performs optimally if the documents are ranked according to decreasing probability of relevance [148]. The history of probabilistic IR models can be traced back to the seminal paper by Maron and Kuhns [105]. The goal is to estimate the probabilities $P(Rel|D)$ and $P(\neg Rel|D)$. The simple decision rule $P(Rel|D) - P(\neg Rel|D) > 0$ can then be used to minimise the probability of error. Other more general loss functions that associate a different cost for different decision errors can also be used. The estimation is enabled by application of Bayes' theorem to the above probabilities (eq. 2.14 and 2.15). Since $P(Rel)$ and $P(\neg Rel)$ are common for all documents, the crucial parameters that have now to be estimated are $P(D|Rel)$ and $P(D|\neg Rel)$.

$$\begin{aligned} P(Rel|D) &= \frac{P(D|Rel)P(Rel)}{P(D)} && (2.14) \\ P(\neg Rel|D) &= \frac{P(D|\neg Rel)P(\neg Rel)}{P(D)} && (2.15) \\ \text{where } P(D) &= P(D|Rel)P(Rel) + P(D|\neg Rel)P(\neg Rel) \end{aligned}$$

If T is the set of query terms that appear in a document, and $p_i = P(t_i|Rel)$ ($q_i = P(t_i|\neg Rel)$) is the probability that a term t_i appears in a relevant (non-relevant) document, then by assuming independence between terms we can express $P(D|Rel)$ using equation 2.16 and $P(D|\neg Rel)$ using equation 2.17. Incorporating term dependence in these equations would involve calculating the joint probability $P(t_i|t_j)$ for every possible pair of terms t_i and t_j. Therefore, a prohibitively large number of parameters would have to be estimated. A solution to this problem is going to be discussed in section 2.7. The second product in these equations is based on the assumption that absence of a term in a document provides evidence for the document's relevance and non-relevance respectively. This is necessary in order to allow the direct comparison between documents that contain different numbers of query terms. Using equations 2.16 and 2.17 and after appropriate monotonic transformations, the initial decision rule or loss function can be transformed into a *linear discriminant function* for evaluating documents [149, 184]. The score that this function assigns to a document should ideally be in the range [0,1], but due to the applied monotonic transformations the maximum possible score usually exceeds one, and its exact value depends on each query. Another approach within the framework of the probabilistic retrieval model is logistic regression [54, 56, 39]. Probabilistic indexing is also the approach underlying the use of inference networks for document indexing [183, 181, 28].

$$P(D|Rel) = \prod_{t_i \in T} p_i \prod_{t_i \in Q-T} (1 - q_i) \tag{2.16}$$

$$P(D|\neg Rel) = \prod_{t_i \in T} q_i \prod_{t_i \in Q-T} (1 - pi) \tag{2.17}$$

Both the above categories of similarity measures can rank documents according to their estimated relevance. The same can be achieved in the case of connectionist approaches to document indexing using *spreading activation* functions [199, 88, 40, 15, 76]. According to this retrieval model, a query is represented by a node that is linked to the nodes representing the query terms. An initial energy is assigned to a query and is subsequently liked through the network and towards the document nodes. A document's relevance is calculated as the final amount of energy that the document received. The overall effect is again a linear evaluation function with characteristics that render it advantageous over the inner product and cosine similarity measures [76]. Since the initial amount of energy is the same for all queries, the number of query terms does not affect document scoring. As already mentioned, spreading activation approaches enable document-based retrieval. Jones and Furnas argue that spreading activation retrieval models deserve further attention, but the increased cost associated with connectionist document indexing still hinders their wide adoption.

2.6.2 Document vs. Classifier

The above similarity measures may also by applied in TC. Probabilistic measures have been applied in the case of Bayes classifiers [73, 131, 122] and distance-based measures can be used for nearest-neighbour algorithms [104]. Typically however, similarity measures for TC depend on the particular classifier adopted. Our goal here is not to review existing approaches, but rather to make a general and important observation about current practices.

Due to the inherent term independence assumption, most classifiers support a linear evaluation function. Such classifiers can only perform binary categorisation of documents. A classifier represents a single topic and is used to evaluate documents according to that topic. Most text categorisation problems however, comprise a fixed number $k > 1$ of topic categories. The common approach to multi-topic categorisation problems is to break the task into disjoint binary categorisation problems. A separate binary classifier is built for each of the k topic categories. The problem is then to appropriately combine the classifiers' output into a k-dimensional vector. Typically, a single topic category is assigned to each document. The output vector in this case has the value one in only one of its dimesions [193]. In a lot of application domains however, a document can belong to more than one topic category. The output vector in this case is either a binary vector [164], with one denoting that a document belongs to the corresponding category, or a weighted vector that can be used to order the topic categories for each document [164, 192]. The caveat with this approach is that it ignores correlations between different topic categories [164].

Recently, there have been attempts to build a single, non-linear classifier that can perform multi-topic categorisation. They adopt neural networks that take as input a document's representation and produce the required k-dimensional vector as output [193, 192]. Hidden layers capture implicitly the stochastic dependencies between terms. Although these single, multi-topic classifiers represent exceptions to the common linear approach to text classification, we will see that their disadvantage in terms of IF is the fixed dimensionality of both the input and output vectors.

In conclusion, for both IR and TC, the dominant term independence assumption has resulted in linear document evaluation functions. Therefore, the corresponding content representations can only effectively represent a single topic of interest. In IR this choice is partially justified by the fact that we can confidently assume that a query expresses a current short-term interest in a specific topic. In TC, the pre-defined and fixed number of topic categories allows the employment of a single classifier for each of the topics. Although, single, multi-topic classifiers have been investigated they can only be applied to categorisation problems for which the topic categories are predefined and fixed. As we will further argue in section 2.10, although non-linearity is necessary for building multi-topic representations, it is not a sufficient factor. Non-linear neural networks have been employed for binary categorisation problems [166].

2.7 Term Dependence

The term independence assumption has been dominant in IR and TC. This is due to the fact that the vector space model does not explicitly represent term dependencies and to the difficulty with which they can be modeled in probabilistic retrieval. Doyle, however, argues, that in building a retrieval system one has to take into account all three dimensions of term dependence [48] (section 2.3). While term weighting can tackle documentation redundancy, the other two phenomena require a different treatment. To deal with language redundancy, Doyle suggests joining terms with strong adjacent, lexical correlation into a single compound term. For reality redundancy on the other hand, he proposes expressing the non-adjacent, topical correlations with an hybrid structure that combines the characteristics of both hierarchical and associative graphs. He states that:

> "The respective disadvantages of either pure-hierarchical or purely associative structures seem to require a compromise. ... we probably need more pathways than are provided by a pure hierarchy, but many fewer than are provided by a coordination system. ... Hybrid arrangements are conceivable: one can have either hierarchies with associational crosslinkages, or association maps with arrows pointing towards subcategories.".

In other words, Doyle points towards the use of appropriately constructed connectionist networks that express both the hierarchical and associative relations between terms, a major theme of this thesis. As we have already discussed in section 2.3, connectionist models of language that explicitly represent term associations are receiving increased attention. Here we will concentrate on how such network structures have been applied to represent term dependence in the general IR context.

As already mentioned, in order to identify term correlations a term's context has to be taken into account. It is usually defined as a span of contiguous words, called "window", that surrounds the word. The size of the window defines the kind of associations that we can identify [94]. A small window of a few words is usually called "local context" and is appropriate for identifying adjacent, lexical correlations. Topical correlations between terms that do not appear close enough are therefore ignored. "Topical context" on the other hand, is defined by a larger window that incorporates from several sentences up to the complete document [68]. We will refer to the latter, special case, as "document context". Typically, when topical context is adopted the order of words within the window is ignored. The goal of topical context is to identify semantic relations between terms that co-occur in documents discussing a certain topic. In that sense, the problem with document context is that it ignores any differences in the semantic content of different document sections. Two terms within the same document are associated even if they appear in sections about possibly unrelated topics. We should also note that the term dependencies that can be captured by topical context are not only caused by topical correlations, but also by lexical. The dependencies that are therefore captured by topical context are in a sense stochastic. A distinction between lexical and topical correlations cannot be made on the basis of statistical evidence provided by topical context alone.

In order to incorporate term dependence in probabilistic IR, van Rijsbergen proposes the use of an appropriately constructed "dependence tree" [184]. The dependence tree is derived as the maximum spanning tree (MST) of the associative graph that represents the statistical dependence between every possible pair of terms. The MST allows the computation to focus on the most significant dependencies. van Rijsbergen suggests the use of the dependence tree for query expansion. Bhatia has adopted this idea to provide personalised query expansion based on a user profile represented by a dependence tree [18]. Query expansion has also been investigated by Park et. al. [135], who propose the automatic construction of a thesaurus using a term similarity measure on a "collocation map". A collocation map is a sigmoid Bayesian network that encodes the statistical associations between terms on a given document collection [61]. A similar approach has been described by Chung et. al. for automatic subject indexing [34]. All of the above approaches use document context and hence they are based on stochastic dependencies between terms.

In addition to the above associative graphs, approaches for the automatic construction of hierarchical networks that capture topic-subtopic relations between terms, have also been proposed. These networks are usually referred to as "Concept", "Topic" or "Subject" hierarchies, and as already mentioned, can be applied for the organisation, summarisation and interactive access to information. One method for the automatic construction of a concept hierarchy is through the use of subsumption associations between terms ("Subsumption Hierarchies") [163]. A term t_i is said to subsume t_j if the documents that contain t_j are a subset of the documents containing t_i. In this way, subsumption hierarchies exploit term cooccurences within the document context and combine them with evidence from the complete document set, in order to identify topic-subtopic relations between terms. Lexical correlations are not explicitly taken into account.

On the contrary, lexical correlations are the basis for the construction of the so called "Lexical Hierarchies" [5, 129]. They are founded on the "lexical dispersion hypothesis" which states that "a word's lexical dispersion – the number of different compounds that a word appears in within a given document set – can be used as a diagnostic for automatically identifying key concepts of that document set". Local context is used to identify lexical compounds, which are then combined with statistics provided by the complete set of extracted compounds, in order to structure the terms' concept hierarchy. The more compounds a term appears in, the higher in the hierarchy it is set. Lexical hierarchies exclude any topical correlations between terms that do not appear close enough in the text. Both subsumption and lexical hierarchies employ some term weighting mechanism for selecting the terms that are going to be used as building blocks of the hierarchy.

In addition to the above two approaches, Lawrie et. al. have investigated the generation of a concept hierarchy using a combination of a graph theoretic algorithm and a language model [94]. The approach has been shown to perform as well as both subsumption and lexical hierarchies. Finally, an evaluation that indicates that subsumption hierarchies are advantageous compared to lexical hierarchies has been conducted [93].

In summary, significant attempts have been made to capture term dependencies. Traditionally, document context has been adopted, resulting in associative graphs that capture the stochastic correlations between terms. On the other hand, concept hierarchies have the potential to represent topic-subtopic relations. However, while subsumption hierarchies do not explicitly take into account the lexical correlations between terms, lexical hierarchies are only based on such correlations. A content representation structure that captures both topical and lexical correlations, while at the same time tackling documentation redundancy, is still lacking.

2.8 The Obtained Information Space

We have discussed how users can access information online, which is either broadcast from dynamic sources or stored and organised in static sources. In the first case the user receives information passively, from the subscribed channels. Documents are received on the fly and no meta-information about their actual content is available. In the second case, the user actively searches for relevant information, either by submitting a query or by exploiting the organisation of static information sources for browsing. Once received or retrieved the actual content of the obtained documents is made available, but despite this initial focusing process (refer back to figure 2.1), it is still cumbersome for the user to go through the obtained information space in order to find the required information. Both the retrieval and the evaluation of the relevance of obtained documents can be automated on the basis of a user profile that represents the user's interests.

2.9 Information Filtering

The research discipline mainly concerned with the application of user profiles for document evaluation is Information Filtering (IF) – a term coined by Denning in 1982 [46]. Since then IF has received increased interest. For its historical development see [133]. In a seminal paper [103], Malone et. al. introduced alongside *cognitive* or *content-based* filtering, *social* filtering, which is usually referred to as *collaborative* filtering [59].

According to Malone, content-based filtering "characterises the contents of the message and the information needs of potential message recipients, and then uses these representations to intelligently match messages to recipients". The analogies to IR and TC are obvious. Content-based filtering is based on content representations of the obtained documents, the profile representation of the user interests and a way of using the profile to evaluate documents. IR and IF have been described as "two sides of the same coin" [16]. On the other hand, although TC is not a user-oriented problem, IF has been approached as a binary classification problem [167], with the IF task cast as classifying incoming documents as either relevant or non-relevant to the user. Because of these perceived similarities, content-based IF research has been dominated by approaches inherited from IR or TC. It is also the most popular approach since it is easy to implement on machine-readable information items like documents.

Collaborative filtering, on the other hand, has been defined by Malone as: "filtering that works by supporting the personal and organisational interrelationships of individuals in a community". In collaborative filtering the information items are not characterised by their actual content but on the basis of ratings received by users in a community. A user profile in this case measures the correlations between a user's ratings and those of other users. Based on these correlations a user's trust in the ratings of other users can be approximated and used to evaluate information items that have already been rated. One of the major advantages of collaborative filtering over content-based filtering is that it is not constrained by the information media. However, it requires a substantial volume of ratings which makes its application to dynamic information sources troublesome [102, 143]. One solution to the problem of sparse ratings is to employ virtual users that automatically rate items [170]. A more promising solution may rely on the integration of content-based and collaborative filtering. Their synergy has been pinpointed as a promising research direction [133], and some first attempts have already been made [59, 12].

Numerous research systems have employed content-based and/or collaborative filtering for a variety of personalisation applications. Table 2.2 presents a summary of existing IF systems, their application area and filtering approach. This kind of IF systems has also been referred to as "Intelligent Information Assistants" or "Agents" [102, 84]. Traditionally, IF has been considered in the context of dynamic information sources like netnews (e.g. Usenet) [16, 52], where the task is to maintain only the most relevant items out of a large stream of incoming documents. However, other application areas emerged. Email filtering and more specifically spam filtering, is similar to netnews filtering since the goal is to remove unwanted items. Browsing

Table 2.2: Application Areas and IF model

System Name	Application Area	Content	Collaborative
ProFile [4, 3]	Netnews	√	
Alipes [195, 197]	Netnews	√	
NewsWeeder [91]	Netnews	√	
NewsDude [20]	Netnews	√	
SIFT [202]	Netnews	√	
PSUN [108, 173, 107]	Netnews	√	
INFOrmer [174]	Netnews	√	
SCISOR [70]	Netnews	√	
NewT [171]	Netnews	√	
PEA [200]	Email	√	
SIFTER [125]	Email	√	
WebWatcher [7]	Browsing	√	
Personal WebWatcher [120]	Browsing	√	
ARACHNID [109, 110]	Browsing	√	
Letizia [97]	Browsing	√	
Syskill & Webert [137]	Browsing & Searching	√	
Amalthaea [127]	Searching	√	
WebMate [32]	Searching	√	
OYSTER [121]	Searching	√	
Watson [25]	Searching	√	
InfoFinder [87]	Searching	√	
Remembrance Agent [144]	Searching	√	
ANATAGONOMY [78]	Personalised Newspaper	√	
Siteseer [153]	Recommendations		√
webCobra [44]	Recommendations		√
Ringo [170]	Recommendations		√
Referral Web [60]	Expert Finding		√
GroupLens [86]	NetNews		√
PHOAKS [179]	Netnews		√
TAPESTRY [59]	Email	√	√
FAB [10, 12]	Recommendation	√	√
Personal TANGO [37]	Personalised Newspaper	√	√

assistants on the other hand suggest links on a web page that may lead to relevant pages. Automatic query formulation underlies the development of search assistants. In the case of personalised online newspapers, an additional step is taken to synthesise the retrieved documents into a personalised news page. Most collaborative filtering systems deal with recommendations of information items like music tracks or movies. Finally, expert finding is an interesting application area that focuses on locating knowledgable peers that can be a useful source of information (section 2.2.1).

Regardless of the particularities of each of the above application areas, a user profile resides at the core of all these services. In the following section, we will concentrate on user profiling for content-based document filtering, but we will avoid committing ourselves to a specific application area. The issue of integrating content-based and collaborative filtering will also receive some attention. We also discuss how a profile can provide other personalisation services like query formulation and expert finding.

2.10 User Profile Representation

Though there are similarities, there is a fundamental difference between IF and, IR and TC. As discussed in section 2.5, in IR the user requirements are assumed to be short-term. TC on the other hand, is concerned with a fixed number of predefined and relatively static topic categories. In contrast, IF concentrates on long-term user requirements. This has two significant implications for user profiling:

1. Although in IR we can confidently assume that the user is currently interested and actively searching for documents about a specific topic, this assumption cannot be made for IF. The user may be interested

in more than one topic in parallel. Even if a single, general topic of interest exists, it may consist of related subtopics. Therefore, a user profile should be able to represent multiple topics of interest.

2. User interests inevitably change over time, driven by changes in the user's environment or the user's knowledge. Unlike TC, the topic categories of interest should be considered dynamic in the context of IF. It is therefore necessary that a user profile can be *adapted* to a variety of changes in the user's interests.

Consequently, IF should be treated as a dynamic, multi-topic representation and document evaluation problem. This view of IF differs significantly from IR and TC. It has requirements that existing IR and TC models cannot fullfill. However, as already mentioned, IF research has been dominated by approaches from IR and TC. Their inheritance has been significant for progress in IF, but has left interesting research directions unexplored. In this section we concentrate on profile representation, initialisation and document evaluation. Adaptation to changes in the user interests is discussed in section 2.13.

2.10.1 Single-Topic Representations in IF

In section 2.5, we argued that term independence is a fundamental assumption in both IR and TC research. In IR, the dominant vector space model does not explicitly represent term correlations. The same is true for probabilistic IR models. Even in the case of connectionist approaches to IR, links between terms are ignored. This leads to linear, document evaluation functions, like the inner product, the cosine similarity measure and, in connectionist approaches, spreading activation (sec. 2.6). Due to their linearity,

Table 2.3: Multi-topic strategies using single-topic profile representations

System Name	Profile Representation	Multi-topic Strategy
ProFile [4, 3]	vector	1-1
WebMate [32]	vector	1-1
Mercure [24]	vector	1-1
FAB [10, 12]	vector	1-1
Alipes [195, 197]	vector	1-1
SIFT [202]	vector	1-1
Syskill & Webert [137]	bayes classifier	1-1
Personal WebWatcher [120]	bayes classifier	1-1
InfoFinder [87]	decision trees	1-1
NewsDude [20]	vector	clustering
SIFTER [125]	vector	clustering
NewsWeeder [91]	vector	clustering
ARACHNID [109, 110]	vector	population
Amalthaea [127]	vector	population
NewT [171]	vector	population
PEA [200]	vector	1-1 & population

document evaluation functions in IR can only estimate the relevance of a document to a single topic of interest.

The same is true for linear TC classifiers. Since, in TC, topic categories are predefined and fixed, the common approach is to break multi-topic categorisation tasks into disjoint binary categorisation problems. A separate binary classifier is built for each of the topic categories.

Inherited from IR and TC, IF systems have traditionally adopted linear, single-topic representations, including both vector representations and linear classifiers. Table 2.3 summarises the representational approach of some well established IF systems. However, as we argued, the user may be interested in more than one topic in parallel. In order to tackle multiple topics of interest, these systems employ three different strategies (table 2.3). Typically, as in multi-topic TC, a different profile is built for each topic of interest. According to this "1-1" strategy, the user is required to define the topic of interests and provide relevant documents for each one of them [4]. Alternatively, online clustering algorithms can be employed to incrementally identify document classes. The number of classes is either predefined [91] or is determined by a fixed relevance threshold [125, 20]. A similar approach maintains a population of profiles which is evolved using Genetic Algorithms (GAs). We'll come back to GAs in section 2.13.

Due to the underlying term independence assumption, such combinations of linear representations can only yield partial solutions to multi-topic profile representation. The topics of interest are assumed to be independent. Neither their relative importance nor their topic-subtopic relations are represented. Using multiple single-topic profiles implies a large number of parameters, like number of terms in each profile and, as we will see, learning coefficients, that have to be optimised. Finally, multiple profiles are more difficult to maintain by the user.

Nevertheless, taking into account term dependence is in itself not sufficient for building a single multi-topic representation. Heuristic phrase extraction has been employed by the InfoFinder system [87], which as in the case of phrase-based indexing results in fixed phrases. Recently, connectionist profile representations employed by the INFOrmer [174] and PSUN [108] systems, adopt an associative term network to represent the user interests. For constructing the network, terms are considered to be associated if they appear in the same phrase. Despite this more flexible way of representing the lexical correlations between terms, both systems reside on the "1-1" strategy for representing multiple topics.

To our knowledge, no existing IF system has used a single profile to represent multiple topics of interest and their interrelations. In accordance with Doyle [48], we argue that this requires the construction of a user profile that tackles all three dimensions of term dependence and recognises topic-subtopic relations between terms. Despite some first efforts (section 2.7), term dependence is still, especially in IF, an unresolved issue and hence a major theme in this thesis.

2.10.2 Profile Initialisation

As in the case of query formulation, in constructing a user profile, the user is the only source of information about what is of interest. In contrast to a query, a user profile is a long-term representation that is initialised once and has then to be adapted to changes in the user interests. It is therefore feasible to ask the user to provide more than a set of keywords for profile initialisation. Yet, IR-oriented IF systems have relied on user specified keywords for the formulation of query-like profiles [27, 202, 171, 127]. This approach to profile initialisation can suffer from the same drawbacks as query formulation (section 2.5) and it can be problematic in general [102].

Foltz and Dumais have shown that in IF, it is more effective and easier for users to express their interests in terms of a small set of documents than as lists of terms and/or phrases [52]. In the case of the "1-1" and population strategies for multi-topic interest representation, the user is usually required to specify a set of documents for each topic of interest. This extra classification burden is avoided in the case of the clustering approaches. Documents can be explicitly specified all at once or progressively collected while the user browses the web or reads emails.

Whatever the case, a set of user-specified documents provides both the pool of candidate profile terms and the necessary information for their weighting and selection. For this task IF systems have traditionally adopted term weighting methods from IR and TC (table 2.4). As already discussed in section 2.4.1, most IR term weighting methods do not exploit relevance information. They are based on document-specific and/or collection-wide statistics. Nevertheless, when profile terms are selected out of the unique terms in the user specified documents, relevance information is implicitly taken into account. In that sense, when a vector representation of a user profile is built out of terms extracted from relevant documents correlations between terms are implicitly taken into account.

The problem with methods from IR is that they are dependent on the existence of a document collection that, in the case of dynamic sources of information, has to be generated dynamically from received documents [71]. One drawback of this approach is the initial sparseness of the collection that can be alleviated using some normalisation factor [196]. Alternatively, one may employ an initial auxiliary collection that is then updated incrementally [2] or an existing controlled vocabulary [90, 10]. It is also preferable in the case of dynamically generated collections, that term weights can be updated online. Note that this rules out the widely adopted TFIDF method. TFIDF term weights, do not explicitly take into account relevance information and also have to be recalculated every time a new document is added.

Query term weighting and methods from TC can exploit the relevance information provided by the user specified documents to measure the specificity of terms to the underlying topic. A term that is specific to a topic can distinguish relevant documents from non-relevant. Therefore, specific terms are of particular importance when building a user profile. In contrast to TC, the user neither has the time nor the inclination to specify a large number of initialisation documents. Furthermore, while in TC the topic categories of interest are known in advance, in IF we can neither predefine the number of interesting topics nor their topical proximity. Finally, we argue that it is easier for the user to specify relevant than non-relevant initialisation documents. The space of non-relevant documents is considerably larger than the space of relevant documents. Therefore, a small number of relevant initialisation documents for each topic of interest is a better and more general expectation of a real situation. It would also be advantageous if the user could submit a single set of initialisation documents. The documents would not have to be pre-classified and instead, the underlying topics are induced as part of the initialisation process.

Although the above characteristics of the user specified initialisation documents can affect the performance of existing term weighting methods, IF systems adopt them on the basis of their successful application in IR and TC. An evaluation of existing term weighting methods and the exploration of new possible solutions in the context of IF could facilitate IF research.

2.10.3 Document Evaluation

Term weighting allows the identification of the most competent terms for building a user profile. If term independence between terms is assumed then the weighted profile terms can be used to evaluate documents on the basis of existing IR and TC models. Table 2.4 summarises the document evaluation approaches adopted by some existing IF systems. The cosine similarity measure has been the most popular approach. However, in addition to its linearity, the cosine similarity measure does not take into account the number of profile terms that appear in a document [76]. It is therefore assumed that the ability to represent a topic is independent of the number of profile terms. In other words, the different topics of interest are treated

Table 2.4: Term weighting and document evaluation

System Name	Term Weighting	Document Evaluation
WebMate [32]	TFIDF	cosine similarity
Alipes [195, 197]	TFIDF	cosine similarity
NewsDude [20]	TFIDF	cosine similarity
SIFTER [125]	TFIDF	cosine similarity
Amalthaea [127]	TFIDF	cosine similarity
NewT [171]	TFIDF	cosine similarity
FAB [10, 12]	TFIDF	cosine similarity variant
NewsWeeder [91]	TFIDF and IG	cosine similarity
ARACHNID [109, 110]	TFIDF variant	linear neural network
Syskill & Webert [137]	IG	various
Personal WebWatcher [120]	IG and others	–
ProFile [4, 3]	F4	inner product
SIFT [202]	frequency	inner product
Mercure [24]	other	spreading activation
InfoFinder [87]	heuristic	–
PEA [200]	WIDF	Dice similarity coefficient

as having the same semantic depth. Usually, a fixed number of terms is used to construct each separate topic-specific profile [196]. Of course, this assumption is wrong. As a user maintains interest and develops expertise in a topic, related subtopics attract the user's attention and therefore more terms are required to represent the initial, general topic of interest.

Since a fixed amount of energy is deposited on a query, irrespective of the number of query terms, the same disadvantage is shared by spreading activation retrieval functions. Another problem with connectionist IR approaches is that they require the existence of an expensive network of terms and documents that prohibits their application to dynamic information sources. This problem is avoided in the case of INFOrmer [174] and PSUN [108] that represent user interests as a network of terms only. Nevertheless, these networks lack direction and so the spreading activation that is employed by INFOrmer is iterative and hence computationally expensive.

The inner product measure does not make the above assumption. It is monotonic to the number of profile terms that appear in a document [76]. Its drawback is that it can overestimate the relevance of a document that contains many profile terms, even if they are not informative or related to a specific topic.

Document evaluation using a TC approach varies for different classifiers. In general, the linearity of existing classifiers and IR models leads to document evaluation functions that measure the relevance of a document to a specific topic with a single relevance score. In contrast, non-linear classifiers can produce a k-dimensional vector that measures the relevance of a document to k separate, but pre-defined and fixed topic categories [193, 192]. Such classifiers cannot deal with changing interests.

The evaluation of documents according to multiple and dynamic topics of interest poses an interesting and challenging research problem that requires a novel approach. A relevance score is a quantitative measure of the user's interest in the document, but it does not provide any evidence of the document's aboutness. Such evidence is necessary for multi-topic IF. The user should be able to select documents based on the topics they discuss and not only their score.

Finally, we should remember at this point, that document evaluation is not the only personalisation service that a user profile can support. As already mentioned, search assistants employ user profiles to automate the formulation of queries that are then submitted to existing search engines. The goal of expert finding on the other hand, is to match peers to someone's information need. Expert finding could be cast as a problem of matching a query or a document to the profiles of other users within a community.

Finally, collaborative filtering is based on ways of assessing the similarity in the interests of different users in a community. Collaborative filtering could therefore be facilitated if such similarities can be measured by comparing the content-based profiles of different users. In conclusion, user profiles can support a variety of personalisation services that significantly increase the spectrum of their applicability. A demonstrative example is the application of IF in the domain of Knowledge Management on the basis of the above services [42]. The provision of such services should not be neglected in IF research.

2.11 The Presented Information Space

The whole point of document evaluation is to increase the probability that the documents that the user finally sees are relevant. The filtering results must be presented to the user appropriately (fig. 2.1). We can distinguish between two different presentation practices that exploit the relevance scoring of documents. In the case of dynamic information sources like netnews, it is traditionally required that a single accept–reject decision has to be made for each individual document. An accepted document is immediately presented to the user. To make this decision based on a document's relevance score, an IF system must employ a decision threshold which has an absolute value for all documents. The threshold can be defined by the user [71, 108], but this requires that the range of relevance values that a profile can assign to a document is known in advance. This is not always true for adaptive profiles because of changes in the number and/or the weights of profile terms. Adaptive threshold calibration through learning is one possible solution [147, 27]. Yet, a single threshold would be problematic if a single, multi-topic profile is used. A document's score is then dependent on the importance of its underlying topic or topics within the profile. Documents about the most interesting topic can receive larger relevance scores than document about less interesting topics. Therefore, a single threshold would inevitably favor documents about the most dominant topic. We believe that thresholding strategies which take into account additional evidence of a document's aboutness are required in this case. Although, we deal with the problem of providing such additional evidence, thresholding is not yet part of our research.

Ranking the filtered documents according to decreasing relevance is not only more generally applicable but also advantageous. Even in the case of dynamic information sources, we can assume with some confidence, that there is enough time between two information seeking episodes to collect the received documents into a batch that is presented as an ordered list whenever required. In addition, the relevance score can be considered an order preserving approximation of the documents' probability of relevance. So, in accordance to the probability ranking principle, ranking of documents according to relevance score is an optimum strategy. By allowing the user to adaptively choose to terminate the information seeking activity, a synergy between human and machine is achieved [133]. Simply put, in interactive applications an imperfectly ranked list can be superior to an imperfectly selected set of documents.

Despite the advantages of document ranking, its application is challenging in the case of documents filtered by a multi-topic profile. For the same reasons as above, the top of the list can be occupied by documents about the most dominant topic in the profile. In addition, if a document's position in the list is the only relevance indication, then the user cannot easily distinguish between documents about different topics. Additional evidence is again necessary. Such evidence could support the automatic generation of a document's summary [94], or enhanced visualisations of the results, as in the case of concept hierarchies [163, 5].

Arguably, the appropriate presentation of the filtering results to the user, especially in the case of multi-topic profiles, poses challenging issues that we will attempt to address. We will however concentrate on document evaluation as a fundamental problem in IF.

2.12 The Read Information Space

The user chooses to read some of the presented documents based on current information interests. One misleading assumption that is usually made by a lot of IF systems is that the presented documents constrain the space of what the user might finally read. In other words the IF system is treated as an interface to the accessible information space. On the contrary we agree with the view of an IF system as an intelligent information assistant [102]. According to this view, the user is not constrained to the documents that the IF system presents. Other document routes are also possible. The documents that a user chooses to read comprise the "Read Information Space" (fig. 2.1).

These documents can be a valuable source of relevance information, if their actual relevance is specified through relevance feedback. In the case of *explicit* relevance feedback, the user evaluates the read documents and indicates their relevance either on a binary or a numeric scale. Profile initialisation is a special case of explicit relevance feedback (sec. 2.10.2), where none of the user-specified documents has been presented by the IF system. Explicit feedback can also be provided on parts of a document [171]. However, explicit feedback has two serious drawbacks [134]:

1. It increases the cognitive load on the user which can counterbalance the reduced cognitive load that results from a presented information space more closely aligned to the user interests.

2. This problem is augmented by the observation that numeric scales may not be well suited to describing the reactions humans have to documents.

These difficulties motivate the study of *implicit* relevance feedback mechanisms which attempt to induce users' reactions by "looking over their shoulder" to observe their reading behaviour. A simplistic way to implement implicit feedback is to assume that everything that is presented to the user is relevant unless the user explicitly defines it as non-relevant [72]. More appropriate measures can however be employed. For example, a strong positive correlation between reading time and explicit feedback provided by the user on a four-level scale has been observed [123]. In the case of email filtering, sources of implicit evidence about the user's interest in each message may include: whether the message was read or ignored, whether it was saved or deleted, and whether it was replied to or not [177]. Finally, one can imagine future systems that would use the user's facial expression and body language to induce the reaction to a read document [136]. Solutions to the drawbacks of explicit feedback are therefore possible.

The relevance information that feedback provides is crucial for the profile's adaptation to changes in the user interests. Not only does it allow the changes to be identified but also provides the means for the adaptation itself. In the rest of this thesis we assume that such relevance information exists for at least some of the documents that the user has read (fig. 2.1).

2.13 Profile Adaptation

Changes in the user interests are caused by changes in the user's environment and knowledge. In a professional environment, for example, changes may occur due to a change in the job assignment or the initiation of a new project. On the other hand, new knowledge is acquired through interaction with other users in a community and due to the information that the user has already seen. The environment imposes specific goals that the user has to fulfill. If the current user knowledge is not substantial, new information needs emerge. Also, the user's current knowledge frames what the user can comprehend and therefore the space of potentially useful information. These interactions can generate a large variety of changes that is difficult to explicitly categorise. In general however we can distinguish between *long-term interests* and *short-term needs*.

Long-term interests correspond to higher level subject areas or topics that define the user's general preferences or expertise. These interests are formed gradually and the time it takes for them to change could be proportional to the time it takes to build them [196]. They are in that sense persistent and are usually coupled with interests in related, more specific subtopics. The user environment affects the choice of related subtopics that attract the user's attention.

Short-term information needs trigger the user to explore specific aspects of the general topics of interest. A short-term need can progressively evolve into a general long-term interest or become obsolete once the user's goals are fulfilled. In the long run a decay of interest in a general topic is also possible. Of course these distinctions are not discrete but continuous.

Changes in the user interests are therefore dynamic. A combination of parameters causes a variety of changes. Fast changes in the current information needs contribute to progressive changes in the user's long-term interests and vice versa. The pace of change varies accordingly and it is definitely not constant. The dynamic nature of the changes in the user information interests renders profile adaptation a challenging and fascinating research area that is the focus of *Adaptive Information Filtering* (AIF).

Adaptive information filtering has recently received increased interest, as reflected in the incorporation of the adaptive filtering task as part of 7th Text REtrieval Conference (TREC-7) and all subsequent TRECs. However, as we will discuss in more detail in section 2.14, the TREC adaptive filtering task and related research, have concentrated on changes in the content of incoming documents from a dynamic information source [92, 194]. Although such changes can be statistically symmetrical to changes in a user's short-term needs, they don't account for the actual problem.

In adaptive information filtering, changes in the user interests must be appropriately reflected by the user profile. This implies that changes in both the long-term interests and short-term needs must be tackled. Consequently, adaptation at adjustable speeds is required. However, the inheritance of profile representations from IR and TC has been coupled by adaptation mechanisms that assume a steady pace of change in the user interests [196].

2.13.1 Profile Adaptation through Learning

Recently there has been a tendency to seek an adequate solution to the problem of profile adaptation in learning algorithms. In IR the most well studied learning algorithm has been Rocchio's [151, 159]. Rocchio's algorithm is applicable to the vector space model and its goal is to readjust the weight of query terms according to the user's feedback. More specifically given an initial query vector Q a new vector Q' is generated using equation 2.18, where D^R and D^N are the vector representations of relevant and non-relevant documents. The parameters α, β and γ determine respectively how much the initial query and the relevant and non-relevant documents contribute to the formulation of the updated query. These feedback parameters are usually determined experimentally and remain fixed. The original Rocchio's algorithm instantiates the parameters as $\alpha = 1$, $\beta = 2$ and $\gamma = 0.5$. Rocchio's algorithm updates the query weights linearly at a rate that depends on the feedback parameters.

$$Q' = \alpha Q + \frac{1}{n_R}\beta\sum_{i=1}^{n_R} D_i^R - \frac{1}{n_N}\gamma\sum_{i=1}^{n_N} D_i^N \tag{2.18}$$

In IF the adoption of the vector space model was accompanied with research on profile adaptation using Rocchio's algorithm. One of the problems encountered is that Rocchio's algorithm is a batch algorithm. A set of relevant and preferably non-relevant documents is required for the algorithm to be effective. This is not however the case for dynamic information sources, where adaptation should be achievable on a per document basis. Although it was demonstrated that a few feedback documents per adaptation cycle can produce relatively good results, the best performance is still achieved when the complete training set is available [2]. Alternatively, online algorithms like the Exponential Gradient (EG) have been proposed and evaluated [27]. Nevertheless, EG is also linear since the pace of learning is defined by a learning coefficient.

Rocchio's algorithm, variations on it and other linear learning algorithms have been adopted by many IF systems [90, 165, 26, 66]. The feedback parameters or some other learning coefficient define the adaptation pace. Therefore, the profile cannot be adapted flexibly to the dynamic changes in the user interests. If a large learning coefficient is selected then the profile is adapted rapidly to changes in the short-term needs, which may lead to over-specialisation to the most recent documents. On the other hand a small learning coefficient can cause high profile inertia, which hinders the profile's responsiveness. One solution is to appropriately adjust the learning coefficient over time. This is the basic idea behind *reinforcement learning* that has been employed by [168, 109, 24, 7]. Usually, the learning coefficient is progressively reduced as more documents become available to be learned from, so that what has already been learned is preserved. The goal is to optimise a profile to a specific topic.

Optimality is also the essence of machine learning algorithms like rule induction, instance-based learning, statistical classification, regression, neural networks and genetic algorithms [117]. As we have already discussed, a lot of IF systems employ machine learning algorithms for learning a user profile for each topic of interest. The underlying assumption is that more training documents lead to improved predictive performance and make the profile more insensitive to noise. However, given the dynamic nature of user interests, a profile built from a large number of training documents that accurately reflect the user's past interests, might perform substantially worse than a classifier limited to more recent documents. Therefore, a good text classification learning algorithm is not necessarily a useful profile learning algorithm. Optimality is a long-term objective that conflicts with the low inertia required for adapting quickly to changes in short term needs [90]. One solution is to assign less importance to older observations [190], but there is evidence that this approach is not always effective [189]. In the context of IF, such windowing techniques have been suggested by [194, 83], that maintain an adjustable window of past feedback documents. This approach is analogous to reinforcement learning in that the window size implies the learning coefficient. Nevertheless, the ability of machine learning algorithms to adapt to radical drifts in the user interests has been questioned [191].

Another solution to the trade-off between fast adaptation to short-term needs and stable convergence on long term interests, is the use of dual profile representations. For the Alipes IF system [196, 195], a three-descriptor architecture is used to represent each topic of interest. It comprises a long-term descriptor and two short-term descriptors (positive and negative). Each descriptor consists of a weighted vector representation that is updated using a linear learning function. The basic difference resides on the learning coefficient (see p. 60). For short-term descriptors the learning coefficient is defined by the user through scaled relevance feedback. On the other hand, the learning coefficient for the long-term descriptor is inversely analogous to the number of relevant documents that have been processed so far. As more documents are made available for learning, the contribution of the more recent feedback documents decreases, in a reinforcement learning fashion, so that what has already been learned is preserved. Dual profiles have also been adopted by clustering approaches to IF [33, 19, 50]. Two different levels of clustering are used to account for long-term interests and short-term needs. Short-term clusters are built from recent feedback documents and long-term clusters from a larger set that contains past documents. When documents cannot be confidently classified by the short-term clusters, classification is delegated to the long-term clusters.

All of the above learning algorithms target single-topic representations. A profile or a cluster representing a specific topic of interest is adapted to modest, local changes of interest in the topic. This is reflected by the fact that learning is based only on feedback documents that were close enough to a profile or cluster [20]. Such learning approaches however, have two significant disadvantages. First, the algorithms do not account for changes in the relevant importance of different topics. In Alipes [195], the suggested solution was to assign different weights to each of the three descriptors representing each topic. These weights define the relative importance of descriptors when filtering documents and are adjusted based on the user feedback using linear learning algorithms. A similar approach is adopted by the SIFTER IF system [90, 125], that expresses the relative importance of individual profiles with two vectors of dimensions equal to the number of topics of interest. The first represents the estimated relevance probability of different topics and the second their "action" probability, i.e. the probability that the documents corresponding to a specific topic will be presented first in the ordered list of filtering results. Reinforcement learning is used to appropriately modify these vectors. The problem is that these learning techniques are external to the learning of the actual vector representation, thus significantly increasing the involved adaptation parameters.

A second disadvantage of learning algorithms is that there is no explicit mechanism for adding or removing terms from a profile representation. The profile is therefore anchored to the topic area that can be represented by alternative configurations of the profile terms' weights. Depending on the learning coefficient, the weights can either be modified rapidly to account for sudden changes in the short-term needs or be progressively optimised for a relatively stable, general interest in the corresponding topic. Yet, a single-topic profile cannot be adapted to radical changes in the user interests, like loss of interest in a topic or the emergence of a new topic of interest. Typically, a new profile is generated when a feedback document is not close enough to existing profiles [195] or clusters [26]. Removing clusters that have not participated long enough in filtering documents has also been proposed [26]. Adding and removing terms to a profile can be achieved using Genetic Algorithms which we discuss in the next section.

2.13.2 Profile Adaptation through Evolution

As already mentioned, IF systems which employ Genetic Algorithms (GAs) maintain a population of profiles or agents that collectively represent the user interests. The population evolves using the genetic operations of *crossover* and/or *mutation*. The fitness of individual profiles in the population is assessed on the basis of an appropriate evaluation function that depends on user feedback. The fittest individuals are selected to "mate" using crossover, in order to produce hopefully fitter offspring, which at some small rate are randomly modified by mutation. Therefore the individuals in the population compete for the ability to reproduce. The overall effect is a random, but at the same time directed exploration of the information space.

The Amalthaea IF system uses an artificial multi-agent ecosystem of evolving agents that cooperate and compete [127, 128]. The ecosystem consists of two general species of agents, namely Information Filtering Agents (IFAs) and Information Discovery Agents (IDAs). Competition takes place among agents of the same species, while cooperation is achieved between agents of different species. Each IFA maintains a weighted vector that represents a topic of interest. Each IDA on the other hand acts parasitically on an existing search engine. It specialises in using IFAs' keywords to formulate queries that are submitted to the corresponding search engines. The retrieved documents are filtered by the IFAs. This is a representative example of the ability of profiles to support both the retrieval and evaluation of documents. The evaluation of individual agents is based on an economic model that derives ideas from agoric open systems [114]. IFAs receive positive feedback in terms of credit which they share with the IDAs that they cooperated with. All agents pay some of their credit as "rent" to inhabit the ecosystem. As a result agents that do not perform well, run out of credit and are purged from the profile. Fit individuals on the other hand have the opportunity to reproduce using crossover. During crossover, portions of the parent vectors are exchanged and thus agents with new combinations of keywords are born. This simple mechanism allows new areas of the information space to be explored. When a new topic of interest emerges, existing agents that may cover it become fitter, while agents that correspond to lapsed topics, loose fitness and are eventually purged from the ecosystem. The problem with Amalthaea is that individual IFAs do not have the ability to learn. Their vector representation remains the same through out their life cycle.

Individual learning agents within evolving populations have been adopted by NewT [171], the IF system described in [8, 9] and InfoSpiders [111] . NewT maintains a population of filtering agents that target Usenet. Each agent corresponds to a weighted keyword vector. When a document that an agent has presented to the user receives positive feedback, the agent's vector is linearly moved towards the document's vector. A similar function is used to update the agent's fitness. Informative terms in the document not already in the agent's vector are added to the vector. In [8], each agent corresponds either to a single term-value pair or conjunctions of such pairs. Each agent is alloted a bid in the range between [-1,1] which is appropriately adjusted based on user feedback using constrained Hebbian learning. An economic model similar to the one used in Amalthaea defines the fitness of individual agents. Finally InfoSpiders maintains a population of agents that autonomously search the web on behalf of the user. The population is initialised by assigning to each agent a starting web page, an initial amount of energy and a query, which can be the same for all the agents. Each agent browses the web by estimating the relevance of each outgoing link from the current document. The agent consumes energy both to visit a new document and to send a relevant document to the user. Additional energy is gained either based on the relevance of the document to the query or through direct user feedback. Q-learning, a variation of reinforcement learning, is used to train for each agent a neural network based on the difference between the relevance of the current document and the relevance of the link that led to it, and the corresponding change in energy. Agents with energy over a certain threshold are selected for reproduction. In addition to the above approaches combinations of GAs with clustering techniques have been proposed [200, 178]. Other systems based on GAs include CIFS [180] and IntellAgent [47].

GAs constitute an interesting solution to the problem of profile adaptation. Motivated by natural evolution the genetic operations allow the exploration of the information space for new areas of interest. Due to the crossover of profile vectors and the addition of new terms, what is represented is not constrained by the current profile terms. Collectively the profiles or agents not only represent the current interests, but can progressively evolve in order to cover a new, potentially remote, area. Furthermore, a profile's fitness is an external indication of the relative importance of the corresponding topic of interest. The problem however is that the evolutionary process requires a substantial number of generations. It was demonstrated experimentally that adaptation is relatively slow [171]. GAs are therefore appropriate for profile adaptation to progressive, but possibly radical changes in the general user interests.

The hybridisation of GAs with learning alleviates this problem. This type of GAs is usually referred to by its disciples as *Memetic Algorithms* (MAs). MAs are founded on Lamarckian evolution and more recently on *memetic theory* [43]. The basic idea is that individuals learn during their lives and what is learned is passed to their offspring. In the context of IF, the ability of individuals to learn allows them to quickly adapt to modest, local changes in the user's current needs. At the same time, the genetic operations enable progressive evolution towards more remote areas of interest. For more details on MAs, the interested reader is pointed to [116]. However, GAs in general and MAs in particular suffer from high computational cost [196]. A diverse enough population of profiles has to be maintained and adapted in parallel. In addition, the relative importance of topics represented by individual profiles is reflected by their fitness and not by the representation itself. Typically, each profile includes the same, fixed number of terms.

In general, all of the profile adaptation approaches that we have reviewed so far, concentrate on profile representations that exclude term dependencies. On the contrary, the evolutionary IF system described in [8, 9], employs agents that represent conjunctions of terms, while the connectionist IF systems, INFOrmer [174] and PSUN [108, 173, 107], that we described in section 2.10.1, use term networks to more flexibly represent phrases. In INFOrmer, adaptation involves the linear update of term and link weights, based on feedback documents. Terms not included in the profile are added to the profile. PSUN's adaptation on the other hand, focuses on the appropriate learning of link weights. It adopts a combination of unconstrained and constrained Hebbian learning. Links that do not appear in relevant documents progressively loose their weight and are forgotten. Nevertheless, in all of the above cases single-topic profiles are being adapted.

In summary, the need for adaptive user profiles is evident. Nevertheless, there has been a tendency to seek an adequate solution in linear learning algorithms that can only achieve a steady adaptation pace. Such algorithms cannot resolve the trade-off between frequent changes in the current information needs and progressive changes in the long term interests. Reinforcement learning and dual profile representation have been suggested to solve this problem. However, adjustable learning coefficients or combinations of them represent only a partial solution. The algorithms neither express the relative importance of different topics of interest, nor is there an explicit mechanism for forgetting old topics or learning new ones. These issues can be resolved using GAs and preferably MAs for evolving a population of learning individual profiles. Despite their motivating background however, these algorithms are computationally expensive. Finally, existing adaptation processes target profile representations that ignore term dependencies. Despite some exceptions to this rule, there is still no process for adapting a single, multi-topic profile representation.

2.14 Evaluating Information Filtering Systems

Personalised information delivery systems are by nature interactive. They don't only provide the user with relevant information, but also require the user's involvement for both profile initialisation and adaptation. Therefore, a successful PID system has to be accepted by individual users and become ingrained in their daily practices. This implies that the system's performance is satisfactory enough to attract the user's involvement. Evaluation of the performance of PID systems is hence required prior to their deployment in a real situation, or before they hit the market. This is a challenging task that has occupied researchers in all related domains. Here we concentrate on the evaluation of IF systems and especially of adaptive IF systems. A thorough review of current evaluation practices can be found in [62], where the authors make a distinction between evaluation by experimentation, evaluation by simulation and analytical evaluation.

Evaluation by experimentation refers to the actual evaluation of the IF system by a sample of users. The reliability of such user studies depends on the number of participants. One of the most extensive field studies have been conducted for the Grouplens collaborative filtering system [86], while a more controlled study involving a smaller number of subjects was performed in the case of FAB [12]. User studies provide a good insight into the human related issues that IF systems have to resolve. Nevertheless, the heterogeneity of users and the difficulties in controlling the experimental parameters render this kind of study difficult to reproduce.

One solution is to simulate users. The simulation involves the use of a document collection for which the relevance of documents to specific topic categories is known in advance. *Virtual* or *synthetic* users with specific interests in one or more of these categories can therefore be used. Simulated experiments can be reproduced accurately. Different systems or system configurations can hence be compared. The latter is of particular importance at early stages of development when fine tuning of a system's parameters is required before its actual use by users. The comparison between different systems implies the agreement on a common and reproducible evaluation standard. This is the motivation behind the Text REtrieval Conference (TREC) which has been held annually since 1992.

Ideally, IF systems should be evaluated analytically so that the system's behaviour is explicitly decoded. The increasing complexity of IF systems however, renders such analytical approaches extremely difficult. Here we concentrate on simulated experimentation which forms the basis of our own system evaluation. The following sections discuss some existing evaluation measures, the TREC-2001 filtering track and the use of virtual users for simulated evaluation by existing systems.

2.14.1 Evaluation Measures

The evaluation of IF systems has benefited by the long experience in the evaluation of IR systems. IR systems have been traditionally evaluated on the basis of *precision* and *recall*. Precision is the percentage of retrieved documents that are relevant and recall is the percentage of relevant documents that have been retrieved. Although it is straightforward to calculate precision, recall requires that the complete set of relevant documents in the collection is known in advance. This is however not true in the case of dynamic information sources.

To accommodate the evaluation of IF systems that have to decide on an item's relevance online, alternative measures have been suggested. The TREC conference adopts the *Utility* measure *T10U* with a credit of 2 for each relevant document (R^+) retrieved and a debit of 1 for each non-relevant document (N^+) retrieved (i.e. $T10U = 2R^+ - N^+$). This corresponds to the learning rule: retrieve if $P(rel) > .33$. The advantage of the utility measure is that it can incorporate the diverse characteristics of individual users. Nevertheless, as we have already argued, online filtering of documents represents only a specialisation of filtering that produces an ordered list of documents. The only difference is the application of an appropriate threshold that may be defined irrespective of the actual document evaluation process.

For the evaluation of IF systems that produce an ordered list of documents, measures that combine precision and recall have been suggested. For example, the *F-beta* measure combines precision and recall with some free parameter which determines their relative weighting [185]. Although numerous other combinations exist, a measure that does not suffer from known drawbacks has not yet been developed [66]. Nevertheless, for our experiments we have adopted the well established *Average Uninterpolated Precision* (AUP) measure (see footnote 3). The AUP is defined as the sum of the precision value at each point in an ordered list where a relevant document appears, divided by the total number of relevant documents. For example, if the first 5 out of a list of 10 documents are relevant to a specific topic and there are a total of 100 relevant documents, then the AUP score of this list is $AUP = (1/1 + 2/2 + 3/3 + 4/4 + 5/5)/100 = 0.05$. If the last 5 documents in the list are relevant the corresponding AUP score becomes $AUP = (1/6+2/7+3/8+4/9+5/10)/100 = 0.0177$. If, on the other hand, the total number of relevant documents is larger, e.g. 200, the above scores are halved. The AUP measure is therefore a combination of precision and recall with an absolute value that depends on the total number of relevant documents.

2.14.2 TREC-2001 Filtering Track

TREC-2001 adopts the Reuters Corpus Volume 1 (RCV1). The latter is an archive of 806,791 English language news stories that recently has been made freely available for research purposes[2]. The stories have been manually categorised according to topic, region, and industry sector [152]. The TREC-2001 filtering track is based on 84 out of the 103 RCV1 topic categories. Furthermore, it divides RCV1 into 23,864 training stories and a test set

[2] http://about.reuters.com/researchandstandards/corpus/index.asp

comprising the rest of the stories[3].

The filtering track is further divided into three subtasks: routing, batch filtering and adaptive filtering. For all subtasks a different profile is built for each of the 84 topics. According to the routing and batch filtering subtasks profile initialisation can be based on the complete set of training documents for the corresponding topic, whereas, for the adaptive filtering subtask, only the first two training documents per topic are allowed. The order of documents corresponds to their chronological order of publication. Given the large number of documents in the training set we can argue that the routing and batch filtering subtasks have been influenced by the TC view of IF. A large number of pre-classified documents are available for training the initial profile. However, as discussed in section 2.10.2 it is more realistic to assume that a user will provide a much smaller number of initialisation documents, which on the other hand can be much larger than just two documents per topic of interest. Thereafter, the adaptive filtering subtask adopts an IR-oriented view of IF, that assumes very limited initial relevance information. Finally, all three subtasks allow the use of any non-relevance related information from the training set. In other words, the training set provides the collection statistics of terms. For this purpose other sources outside the RCV1 could also be used.

The constructed profiles are tested against the complete test set. The output of the routing task is a ranked list of the best scoring 1000 documents. Systems are evaluated by calculating the AUP of this list. According to the adaptive and batch filtering tasks on the other hand, systems have to select a subset of the test set by evaluating documents in their chronological order.

[3] For more details on the TREC 2001 filtering track see:
http://trec.nist.gov/data/t10_filtering/T10filter_guide.htm

This implies the use of thresholding for making the binary decision between selecting or discarding each document. Systems are evaluated by calculating the precision, recall, Utility and F-beta measure of the unordered, output set. The difference is that for the batch filtering subtask the initial profile and threshold remain constant. The adaptive filtering subtask though, tests the ability of systems to adapt to changes in the content of a topic's test documents, over time. Each accepted document is immediately judged for relevance, and this information can be used by the system to adaptively update the filtering profile and/or adjust the threshold. The assumption is made that this is the only available relevance information during a profile's life cycle. The possibility of external sources of relevance information is excluded. So, not only is the ability of systems to adapt evaluated on the basis of relatively local and loosely controlled changes in content, but also a system's performance depends on appropriate threshold calibrations that are not necessarily an unbreakable part of profile adaptation.

In conclusion, the TREC conferences represent a serious attempt towards the standardisation of IR system evaluation. In terms of IF however, the guidelines are still influenced by the traditional view of IF as a specialisation of IR or TC, that focuses on dynamic information sources, where the value of documents decays rapidly with time. The allowed number of initialisation documents and the dependence on thresholding, reflect this trend. In addition, the adaptive filtering task does not test the ability of systems to adapt to radical changes in a user's general topics of interest, like losing interest in a specific topic or the emergence of a new topic of interest. It has been acknowledged that it is difficult to reach an agreement regarding the evaluation of AIF systems [66]. Another recognised drawback is the large number of test documents per topic in RCV1, which does not always reflect a realistic situation [146]. For these and other considerations, the filtering track has been removed from TREC-2003. There is clearly space for improvement in the way IF systems, including adaptive, are being evaluated. The TREC filtering guidelines focus on systems that use a separate profile for each topic of interest. The evaluation of single multi-topic profiles has been characteristically ignored. To our knowledge no existing evaluation standard considers such profile representations.

2.14.3 Virtual or Synthetic Users

Radical changes in a user's long-term interests can be simulated using virtual users. Given a document collection that has been pre-classified according to a number of topic categories, a virtual user's current interests can be defined as a small subset of the existing topics. Interest changes can then be simulated by modifying this subset. Loosing interest in a topic is simulated by removing the topic from the subset. Similarly, the emergence of a new topic of interest can by simulated by adding a new topic to the subset.

This approach has been adopted for the evaluation of Alipes [196], SIFTER [90] and NewT [171]. In the case of Alipes, experiments were performed using the Reuters-21578 1.0 test collection with the ModApte split, that divides the collection into 9603 training documents and 3299 test documents. Five topics with at least 100 documents in the collection were selected. 100 documents for each topic were then divided into 80 training documents and 20 test documents. A virtual user's current interests were reflected by a combination of some of the five topics. Interest changes were simulated by negating one or more of the topics in the current group and adding additional topics. The training documents that corresponded to negated topics were used as negative feedback. The system was evaluated against a series of such simulated changes. In contrast to Alipes, where a pre-classified collection of documents was used, the evaluation of SIFTER and NewT were performed on a dynamic information source. Therefore, the relevance of incoming documents was not known in advance. Virtual users were simulated as a list of keywords for each of a set of predefined categories. A document was assumed to be relevant to a topic if it included the corresponding keyword(s). Changes in the virtual user's interests were simulated by modifying this list. Although this latter approach allows the adoption of virtual users for evaluation of systems against dynamic information sources, it suffers by the fact that the virtual users assess the relevance of incoming documents on more loose evidence than the actual profile. Nevertheless, it has been claimed that experiments with simulated users were more conclusive than experiments with real users [171]. Virtual users are also adopted by [11, 112]

In general, virtual users represent an interesting evaluation approach for AIF systems. Changes in the user's interests can be simulated in a controlled way, instead of being cast to changes in the content of incoming documents, as in the case of the TREC adaptive filtering subtask. Systems can therefore be tested against radical drifts in the topics of interest. However, existing instantiations of this approach have been relatively ad-hoc. Furthermore, in the case of Alipes, the number of test documents is substantially small, smaller than the number of training documents. In a real situation an IF system might have to identify relevant documents within a collection containing thousands of documents. Further steps towards evaluation standards that employ virtual users for the testing of AIF systems should be made.

2.15 Summary and Conclusions

In this chapter we have reviewed models and techniques involved in personalised information delivery. The discussion was structured along a model of PID that facilitated an integrated presentation of related research domains (fig. 2.1). Starting with the accessible information space, we distinguished between dynamic and static information sources. We moved on to describe how, in the case of static information sources, automatic indexing of documents enables their categorisation and subsequent retrieval. Usually, document indexing involves the weighting and selection of terms using methods that exploit the statistical characteristics of language.

TC and IR have been the focus of the first part of this review. The discussion included both the way a topic category of interest is represented in these two domains and how it is used for document evaluation. Term weighting is employed when building a topic representation. However, we made the observation that the dominant term independence assumption leads to topic representations that support linear document evaluation functions. Such representations, although to some extent justified in the context of IR and TC, can only represent a single topic of interest. It is however acknowledged that the term independence assumption is wrong. Term dependencies are caused by both topical and lexical correlations between terms. Efforts have been made to take into account term dependencies for term weighting, query expansion and for extracting concept hierarchies from a set of documents. The latter kind of representation has been highlighted for its ability to represent topic-subtopic relations between terms. This first part of the review has set the technical foundations of document representation and evaluation for the rest of the discussion. It concludes with a description of the characteristics of the obtained information space, which includes all information items that have been either received from a dynamic information source or retrieved from a static one. It was then noted that both the retrieval and evaluation of obtained documents can be automated on the basis of a user profile.

Arguably, the research area that is founded on user profiling is IF. After describing the two main approaches to IF, namely content-based and collaborative filtering, we presented a summary of its main application areas and a list of corresponding research systems. The rest of the discussion concentrated on content-based filtering, without however committing ourselves to a specific application area. It was then stressed that despite their higher-level similarities the long-term nature of user interests significantly diffentiates IF from IR and TC. Not only it is reasonable that the user may be interested in more than one topic in parallel, but also that inevitably the user's interests will change over time. These characteristics of the user interests have significant implications for user profiling. Nevertheless, due to the perceived similarities, IF has been dominated by methods inherited from IR and TC. The subsequent discussion established this argument. Initially, we have described current approaches to profile representation. We then looked more closely into the processes of profile initialisation, and document evaluation. The latter allows the appropriate presentation of the filtering results to the user. The goal is to increase the probability that what is finally read by the user is relevant. Despite the additional effort, it is then required that the user provides feedback about at least some of the read documents. This additional relevance information can be used for adapting the profile to changes in the user interests. In addition to the above processes, which are incorporated in the initial PID model (fig. 2.1), we finally discussed current practices for the evaluation of IF systems prior to their actual deployment in a real situation.

This review of the state-of-the-art in research related to PID, has revealed a number of research directions that have been left unexplored. More specifically:

1. The term independence assumption has been common both to IR and TC research. Despite recent attempts to incorporate term dependencies into content representation structures, no existing approach tackles all three dimensions of term dependence.

2. IF has been approached as a specialisation of IR or TC. This has led to profile representations that inherit the term independence assumption, leading to single-topic profiles. But, in contrast to IR and TC, in IF, a user may be interested in more than one topic in parallel. Despite recent efforts to incorporate term dependencies, representing multiple user interests with a single profile has not yet been researched.

3. In building a user profile IF systems adopt term weighting methods based on their successful application in the context of IR and TC. Given the differences in the kind of relevance information that is usually available for profile initialisation, an evaluation of existing term weighting methods and the investigation of new ones in the context of IF, should be pursued.

4. Document evaluation according to multiple topics of interest poses another interesting research issue. Apparently the quantitative relevance score, that a document evaluation function assigns to each document, is not sufficient. Additional evidence of a document's aboutness should be provided. Along the same lines the user profile could be used for supporting additional personalisation services that can broaden its scope.

5. Providing additional evidence of a document's aboutness should be coupled with appropriate presentation of the results. Ordering documents

according to their relevance score is not enough.

6. A more significant observation was made for the current profile adaptation practices. There has been a tendency to seek a solution in linear learning algorithms that can only achieve a steady adaptation pace. Such algorithms cannot cope with the dynamic nature of the user interests. Despite some efforts to overcome this disadvantage using reinforcement learning and dual profile representations, the achieved solutions are only partial. Alternatively, GAs and MAs have been adopted that suffer from a high computational cost. Furthermore, as a consequence of the dominance of single-topic profile representations, the adaptation of multi-topic profiles has been neglected.

7. The influence of IR and TC is also evident in the way IF has been evaluated. The well established TREC conference and more specifically TREC-2001 reflects this attitude, both in the way profiles are initialised and in the dependence on thresholding. In addition TREC's adaptive filtering subtask simulates changes in the user interests as changes in the content of incoming documents. Changes in the user interests can be simulated in a more controlled way using virtual users, but their application has not been yet standardised.

The above research directions require further investigation. In the next chapter we present a methodology that generates, out of a set of relevant documents, a hierarchical term network representation of a user profile, through a series of processes that take into account, document, language and reality redundancy. In other words, we describe the initialisation of a profile representation that tackles all three term dependence dimensions. The first of the three processes involves the weighting and selection of the most informative terms in the initilisation documents. In this context, we introduce a novel term weighting method and we evaluate it together with a number of existing methods. This first step is complemented by a step for identifying term correlations and a final step that generates the hierarchical network. We argue that the generated profile can represent more than one topic of interest.

The question that we answer in chapter 4, is how can the hierarchical profile be used for multi-topic filtering. This includes both the quantitative evaluation of a document's relevance and how it can be complemented with additional evidence of the document's aboutness. We introduce a series of document evaluation functions and we evaluate them with both single-topic and multi-topic filtering experiments. We also propose ways of using the profile to support personalisation services like automatic query formulation and expert finding.

The most challenging and fascinating aspect of IF, profile adaption, is the theme of chapter 5. Inspired by biological theories of self-organisation, we introduce a process that allows a single, multi-topic profile to adapt to a wide variety of changes in the user's interests. Our experimental evaluation using virtual users has satisfied our expectations.

In general, all our experiments were based on the TREC-2001 routing subtask and therefore the need for thresholding was avoided. For each experiment, appropriate modifications to the standard routing guidelines were introduced to account for the particularities of IF. So although evaluation of IF systems has not been the main focus of our research, we point towards possible directions for a standardised methodology.

Chapter 3

Building a Multi-Topic Profile

> "The statistically dependent placement of words in text is a natural consequence of the way people think and communicate"
>
> Doyle, 1962

According to Doyle, the language redundancy phenomenon causes lexical correlations between terms. Reality redundancy on the other hand results in topical correlations [48]. By documentation redundancy, we refer to the phenomenon that causes correlations between terms and larger semantic units, like documents or document classes. Although it is natural for such correlations to occur the term independence assumption has been common in IR, TC and subsequently, in IF research. Exceptions include associative graphs, employed for query expansion, that express stochastic dependencies between terms [184, 135, 34]. More recently, concept hierarchies, that identify topic-subtopic relations between terms, have been extracted from document sets and applied for their visualisation or automatic summarisation [163, 5, 129]. However, subsumption hierarchies do not explicitly take into account lexical correlations. They don't take into account how close terms appear to each other. On the other hand, lexical hierarchies are based only on correlations between adjacent terms in text (section 2.7). In IF, connectionist approaches that only represent the lexical correlation between terms have also been suggested [174, 108]. In conclusion, a content representation that tackles all three dependence dimensions and recognises topic-subtopic between terms, is missing. This is the goal of this chapter.

We present a methodology for generating a hierarchical term network representation out of a set of user specified documents. The set of documents may discuss multiple topics of interest. In other words, we describe the initialisation of a single, multi-topic profile. Its application for document evaluation is the subject of the next chapter. As we have discussed in section 2.10.2, the number of initialisation documents that the user provides is another distinguishing IF factor. It is further investigated in the next section. The methodology involves three steps. Initially, we investigate methods for weighting and selecting the most competent terms in the user specified documents (sec. 3.2). We then identify and measure dependencies between terms that are caused by both lexical and topical correlations (sec. 3.4). This second step results in an associative graph that is finally transformed into a hierarchy in step three (sec. 3.5).

3.1 A User-Study on Profile Initialisation

In section 2.10.2, we have argued that for profile initialisation the user can specify a number of relevant documents for each of the topics of interest. However, the number of specified documents is expected to be smaller than the hundreds of documents that are usually available for the training of classifiers in TC. To test this hypothesis, we have conducted a limited user study. Seven PhD students, a representative example of individuals that need information as part of their daily activities, were asked to provide relevant and non-relevant documents for their hypothetical profile initialisation. The subjects were provided with a simple web-based interface where they could define topics of interest or not, and submit documents for each one of them. They were prompted to provide as many documents as possible, since this could be beneficial for their profile's performance.

The study provided an indication of both the number of initialisation documents and the number of topics that users may specify in a real situation. Table 3.1 summarises the number of topics and the number of documents per topic that each of the seven students specified. Overall, the students specified an average of two topics of interest, ranging from 1 to 4 topics. Five out of the seven students specified more than one topic of interest, which indicates that they can distinguish between several subject area that they are currently interested in. Interestingly, none of the students has specified non-interesting topics. As expected, it was more straightforward for the students to select initialisation documents out of the limited space of relevant documents than out of the much larger non-relevant space. For each topic of interest, the students specified an average of 23.125 documents, ranging from 1 to 123 documents per topic. We expect that in a real situation, extra motivation and improved interfaces may increase this number. It was also noted that students in the first year of their PhD process (students D, F and G) specified fewer topics and a smaller number of documents per topic than students in the second (student E) and third year (students A, B and C). This may be illustrative of how interests evolve over time.

Although our sample was not statistically significant, this study supports our argument and our experimental choices henceforth. More specifically:

Table 3.1: User Study Results ((T)opics, (D)ocuments)

Student													
A		B		C		D		E		F		G	
T	D	T	D	T	D	T	D	T	D	T	D	T	D
I	3	I	123	I	10	I	10	I	18	I	7	I	1
II	5	II	34			II	8	II	13			II	2
III	31	III	78									III	4
IV	23												

- It indicates that users may be interested in more than one topic in parallel.
- It supports our intuition that it is more general to expect that the user will specify only relevant documents.
- It supports the argument, that although users are unlikely to provide hundreds of documents for each topic of interest, yet, they are likely to provide more than 2 documents per topic.

3.2 Step 1: Term Weighting and Selection

In the first step of the methodology, the goal is to extract out of the user specified documents those terms that are more specific to the underlying topic (or topics). Such terms can distinguish documents about that topic from other obtained documents. Initially, stop word removal and stemming, using Porter's algorithm, is applied to reduce the space of unique terms in the documents. [1] The specificity of the remaining terms within the collection, or in particular their specificity to the underlying topic, can then be

[1] Although in the future we intend to evaluate the necessity of these dimensionality reduction techniques, for our current research we have decided to comply with these practices.

measured using existing term weighting methods (sections 2.4.1 and 2.5.3). The applicability of existing term weighting methods from IR and TC for the problem at hand is further discussed in the next section. Complementary to these IR- and TC-oriented approaches, we introduce a new term weighting method, called *Relative Document Frequency*, that has been devised so that it complies with the particularities of IF (section 3.2.2). The assigned weights can then be used to extract an absolute number of the most specific terms or those with weights over a certain threshold. The extracted terms are used to populate the profile. If we ignore term dependencies, this unconnected profile version can be applied for document evaluation, using, for example, the inner product between the profile's and each document's vector representations.

3.2.1 Existing Term Weighting Methods

In section 2.4.1 we presented a number of term weighting methods that are used for automatic indexing. Such methods do not explicitly take into account relevance information: they are only based on document-specific and/or collection-wide statistics. Methods that take into account collection-wide statistics, like *df*, can be used to select out of the user-specified documents, those terms that are in general specific within the collection. By constraining the choice of profile terms to the user specified documents, we implicitly take into account relevance information.

As we have discussed in section 2.10.2, one problem with methods that exploit collection-wide statistics is their dependence on the existence of a general document collection. However, although it can be problematic to compile such a collection in the case of dynamic information sources, an auxiliary collection may overcome this problem. More serious drawbacks can be identified in the case of methods that assign document-specific weights.

These methods incorporate the *tf* of terms and therefore the absolute value of the assigned weights depends on the total number of terms in a document. Weight normalisations have been applied to alleviate this problem. In addition, such methods are usually batch, in that the weight of terms in the documents has to be recalculated every time a new document is added to the collection. This is the case for the widely adopted TFIDF method. Finally, to calculate a term's weight in the profile, its document-specific weights in the user-specified documents have to be appropriately combined into a single weight that expresses the term's correlation to the documents' underlying topic. This is an additional computational step which can be avoided using methods that exploit relevance information. Taking these considerations into account, we have chosen to exclude methods that assign document-specific weights, including TFIDF.

Term weighting methods that have been applied for query term weighting, or in the context of TC, take into account relevance information in order to explicitly measure the specificity of terms to the topic underlying a set of user-specified documents. We argue that these methods are more appropriate for the problem at hand. However, in the case of TC, it is assumed that a large set of pre-classified documents are usually available. For each of the topic categories, thousands of training documents may exist. In addition, the classification is usually performed by more than one human indexer and the topic categories are coarse enough to facilitate the classification of thousands of documents. Therefore, we can, with some confidence, treat any documents that have not been assigned to a specific topic as non-relevant to that topic. However, as our user study further supports, in IF, we can neither predefine the number of topics that the user is going to specify nor their topical proximity. As a consequence we can not confidently make the above assumption.

Finally, it would be preferable for users to be able to just specify a single set of relevant documents thus avoiding the classification effort. For these reasons we propose in the next section a novel term weighting method and we then evaluate it together with a subset of the methods discussed in the previous chapter.

3.2.2 Relative Document Frequency

Relative document frequency (RelDF) is a measure of the relative importance of terms within the user specified documents and a general collection of documents. The method appears in a paper by Porter [139], but lacks further usage. The essence behind the approach is also analogous to the *relative frequency technique* that has been suggested by Edmundson and Wyllys [49] (hence the adopted name). Based on the assumption that special or technical words are more rare in general usage than in documents about the corresponding subjects, they presented a number of ways for assessing the relative frequency of terms within a document and a general collection (section 2.4.1).

In a similar way, we assume that terms pertaining to the topic of interest to the user will appear in a larger percentage of the user specified documents than in the general collection. The goal is to identify a user-specific vocabulary that distinguishes the documents of interest from the rest of the collection. The method assigns to each term, a weight in the interval (-1,1), according to the difference between the term's probabilities of appearance in the user specified documents and in the general collection. Using the notation of the contingency table (table 2.1) we define RelDF using equation 3.1. While the first part of the equation ($\frac{r}{R}$) favours those terms that exhaustively describe the user specified documents and therefore the underlying topic of interest, the second part $(-\frac{n}{N})$ biases the weighting towards terms that are specific within the general collection.

$$RelDF = \frac{r}{R} - \frac{n}{N} \tag{3.1}$$

The involved statistics are the same as in the case of Robertson and Sparck Jone's first Formula F1 (eq. 2.6). However, as we will see, the logarithms used in F1 result in different weighting behaviour. Recently, it was also brought to our attention that RelDF may be derived from Rocchio's algorithm if certain assumptions are made [6]. If in equation 2.18, $\alpha = 0$, $\beta = \gamma = 1$, and we assume, binary indexing of documents and that the complete collection is non-relevant, then the equation calculates a weighted vector with each weight equal to the individual term weights that RelDF calculates.

RelDF has a number of theoretical advantages. Firstly, it is does not require non-relevant documents. Furthermore, it uses probabilities of appearance, which make accurate estimations possible even in the case of a small number of initialisation documents. The involved statistics can be updated online and therefore the method is applicable in the case of dynamically compiled document collections. Finally, RelDF is not dependent on the number R of initialisation documents. Although, a large R provides statistical confidence, it does not exclude the application of RelDF even in the case of $R = 1$. In other words RelDF can be applied both in batch and an online mode. We'll come back to this latter case in chapter 5. The only requirement for the application of RelDF is the existence of a general collection of documents which as we have already mentioned is possible even in the case of dynamic information sources.

Table 3.2: Term weighting methods

Abbreviation for method	Abbreviation	Equation
Information Gain	IG	2.9
Relative Document Frequency	RelDF	3.1
Relevant Document Frequency	RDF	2.8
χ^2 (chi square)	CHI	2.11
Robertson's 4th Formula (predictive)	F4	2.7
Robertson's 1st Formula (retrospective) & Mutual Information	F1/MI	2.6
Inverse Document Frequency	IDF	2.3
Residual Inverse Document Frequency	RIDF	2.5

3.3 A Comparative Evaluation of Term Weighting Methods

In order to assess the specificity of the unique terms in the user-specified documents, one has to make a choice from a variety of existing term weighting methods. As already mentioned, existing systems usually adopt a term weighting method based on its successful application in IR or TC. However, in IF, the availability of limited relevance information and the potential lack of non-relevance information for profile initialisation, may affect their effectiveness. This has motivated us to conduct a comparative evaluation of existing term weighting methods and of the novel RelDF, in such a way that the above particularities of IF are taken into account.

We have experimented with those existing methods that comply with the requirements set in section 3.2.1 above. Table 3.2 summarises the evaluated methods. These methods can be used to assign a topic-specific, or collection-wide weight to the unique terms in the user-specified documents. With the exception of RDF, the only requirement is the existence of collection statistics from a base collection. RIDF is the only method that uses the within document frequency of terms, as part of the underlying Poisson distribution.

3.3.1 Evaluation Methodology

We evaluated the term weighting methods using a slight variation of the TREC-2001 routing subtask. Our goal was to comply with an existing and well established evaluation standard as much as possible, while at the same time to take into account the small number of user-specified relevant documents. The choice of the routing subtask was made to avoid the need for thresholding. As already discussed, according to the TREC-2001 routing subtask, systems are allowed to use the complete relevance information and any non-relevance-related information from the training set. Systems are evaluated on the basis of the best 1000 scoring test documents, using the AUP measure.

We have deviated from the routing guidelines in the following ways. To reduce the time needed for each experiment we have only used the first 10 out of the 84 TREC topics (R1-R10). Furthermore, in order to more realistically reflect the number of relevant documents that a user may provide for each topic of interest, systems were allowed to use only the first 10, 20, 30 and 40 relevant documents per topic – far less than the hundreds provided for most of the topics by the training set.

The training documents for each of the 10 topics, were preprocessed by stop word removal and stemming using Porter's algorithm. The remaining terms were weighted by each method and a topic specific profile was constructed using the most competent terms. In order to evaluate the effect of the number k of profile terms on the profile's filtering performance, different profiles were constructed for each $k \in \{2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 40, 60, 80, 100\}$. More results were produced for profiles with a small number of terms, to recognise term weighting methods that could identify the most informative terms in only a small number of extracted terms. In summary, a different topic-specific profile was constructed for each possible combination of term weighting method, topic, number of relevant documents and number of profile terms. In total, 1120 profiles were evaluated for each of the term weighting methods.

The profiles were then used to assess the relevance of the documents in the test set. Independence between profile terms and binary indexing of documents were assumed. Stemming of terms was again applied to the test documents. For each profile P and document D, two different evaluation functions were adopted. In the first case, documents were evaluated according to the inner product (equation 2.12). Since binary indexing was assumed, the inner product can be simplified to equation 3.2, in which dw_i becomes 1. In that sense a document's relevance is calculated as the sum of the weights of profile terms that it contains. For both functions the score was normalised to the number NT of terms in the document, to smooth its effect on the document's score.

$$S0_D = R_{P,D} = \frac{\sum_{t \in D} w_t \cdot 1}{log(NT)} \tag{3.2}$$

We have also experimented with evaluating a document by the product of the weights of profile terms that it contains. In this case a document's relevance R was calculated by equation 3.3. This relatively ad-hoc approach, was derived from the joint probability of independent features (equation 2.16) by removing the second product. Our goal was to find another way of uniformly comparing the term weighting methods. This multiplication approach is applicable as long as term weights are greater than one. Only then does the product of term weights increase with the number of terms. Thereafter, the weights of profile terms have been scaled so that no weight is less than one.

As in the case of the inner product [76], the drawback of the multiplication approach is that it can overestimate the relevance of a document containing too many profile terms, even if these terms are not the most informative terms in the profile.

$$R_{P,D} = \frac{\prod_{t \in D} w_t}{log(NT)} \tag{3.3}$$

3.3.2 Results

Each profile was used to evaluate the documents in the test set. The AUPs of the profiles corresponding to each method were averaged over the different topics (R1-R10) and the different numbers of relevant documents (10, 20, 30, 40). Figures 3-1 and 3-2 respectively present the results using summation of weights and multiplication of weights. In these graphs, the x-axis corresponds to the number of profile terms and the y-axis to the average AUP score. A different line has been plotted for each term weighting method. Finally, in table 3.3 each method's score for different numbers of profile terms has been averaged to a single overall score value.

The results reveal a significant difference in the performance levels of IG, RelDF, RDF and CHI in comparison to F4, F1/MI, IDF and RIDF. In other words methods from TC that explicitly take into account relevance information, appear to perform better than methods from IR. These first four methods are those biased towards the information provided by the user. They favour terms that appear in a lot of relevant documents over those appearing in only a few. In contrast, although F4 and F1/MI also exploit relevance information, the smoothing effect of logarithm in combination with the small number of user specified documents and the substantially larger number of documents in the collection, biases F4 and F1/MI towards information acquired from the collection (eq. 2.6 and 2.7). Large differences in

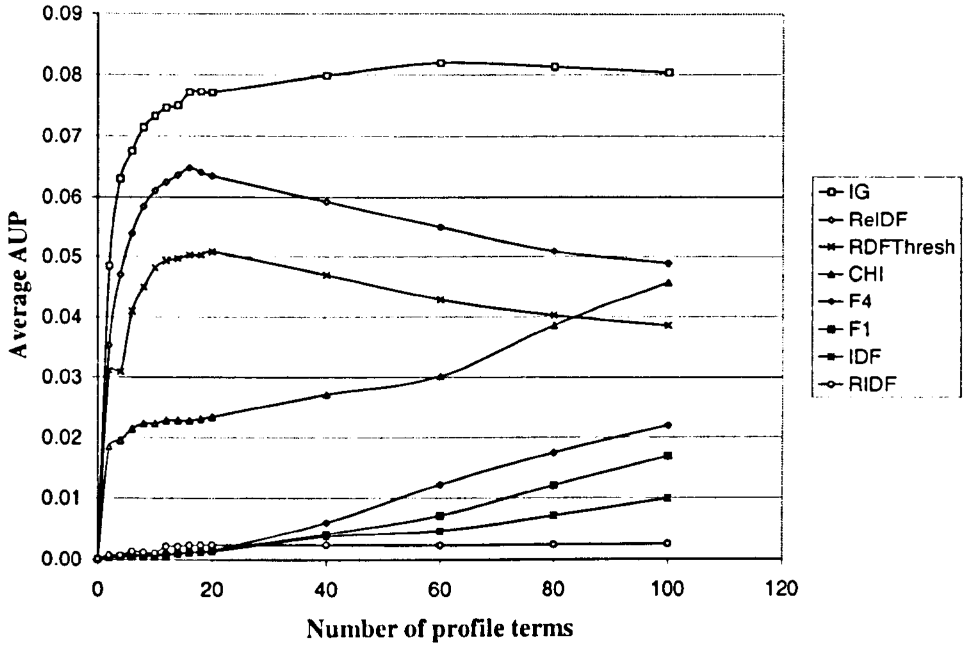

Figure 3-1: Results of experiment using summation of weights (eq. 3.2)

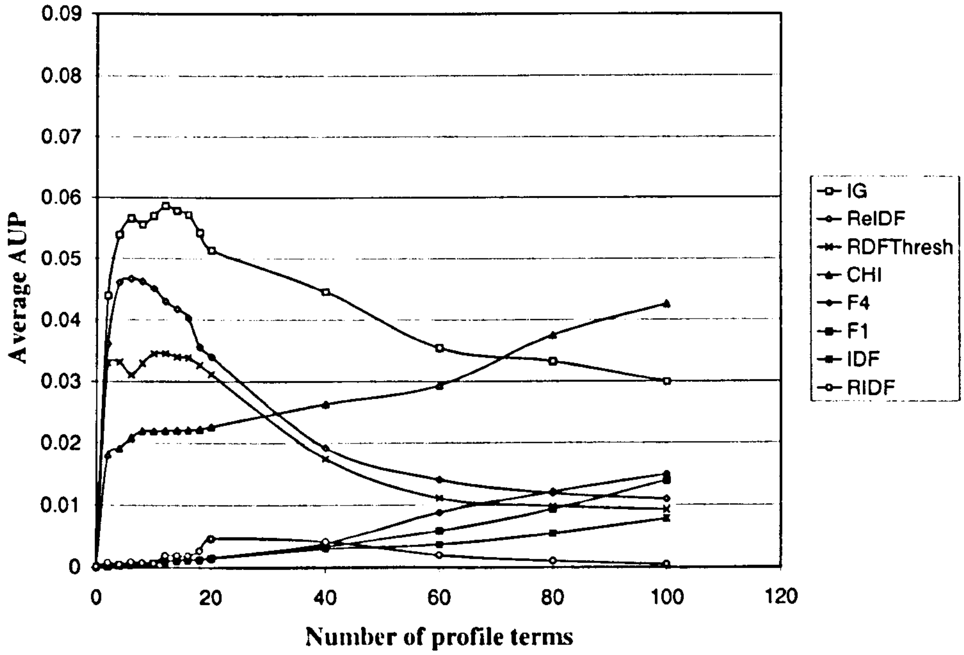

Figure 3-2: Results of experiment using multiplication of weights (eq. 3.3)

Table 3.3: Overall Score

	Evaluation Function	
Method	Summation (eq. 3.2)	Multiplication (eq. 3.3)
IG	0.07346	0.04926
RelDF	0.05629	0.03366
RDF	0.04392	0.02707
CHI	0.02574	0.0249
F4	0.00482	0.00346
F1/MI	0.00352	0.00285
IDF	0.0024	0.00197
RIDF	0.00186	0.00168

the document frequency of terms are more strongly taken into account than small differences in their relevant document frequency. This negative effect of algorithmic smoothing is evident in the difference between the performance of RelDF and F1/MI. Although both methods use the same statistics, the application of logarithms results in reduced performance for F1/MI. The importance of the user specified information is also highlighted by the poor performance of IDF and RIDF that do not take into account the relevant document frequency of terms. However, the information provided by the user is not sufficient for optimum performance. Despite the fact that RDF performs substantially better than IDF and RIDF, RelDF performs even better. The difference in their performance is due to the collection statistics that RelDF takes into account (second fraction of equation 3.1).

Apparently, the way a term weighting method combines information provided by the user and information acquired from the collection is a significant performance factor. While both kinds of information should be taken into account, what the user provides is of increased importance. This finding is not only supported by the higher overall score of the first four methods of table 3.3, but also by the increased performance of the first three of them for small numbers of profile terms. IG, RelDF and RDF have the ability to identify the most informative terms within only a small number of extracted terms. The opposite happens in the case of F4, Fl/MI, IDF and RIDF. CHI appears to behave in a way intermediate to these two extremes.

Table 3.3 presents the evaluated methods by decreasing order of overall score. IG is the best performing approach while RelDF represents a promising alternative. It appears that IG is affected less by the number of profile terms. Figure 3-3 presents an example distribution of normalised IG and RelDf weights. The weights have been normalised to the maximum value for each method so that a direct comparison is allowed. Note that the actual order of terms is not necessarily the same for both methods. Nevertheless, the graph indicates that the relative weights that IG assigns to the best 10 terms are larger than those RelDF assigns. With a smaller difference, the opposite appears to be happening for about 40 subsequent terms. Therefore, given the large number of test documents per topic[2], the change in their content over time and the small number of initialisation documents with a limited vocabulary to choose from, it is advantageous to favour, as IG does, the most general terms which exhaustively describe the test documents. On the other hand, term weighting using RelDF, may overestimate less exhaustive terms that are specific to the current temporal content of the training documents, leading to over-specialisation. Nevertheless, although, given the above characteristics of the training set, over-specialisation may be disadvantageous, we believe that it could be an advantage in the case of adaptive information filtering and for domains where the target documents are only a small subset of the available set. Adaptation can allow a profile to constantly specialise in current subtopics of interest while it maintains the representation of the general topic (chapter 5). Finally, in addition to

[2] The large number of documents per topic is an acknowledged drawback of RCV1 [146].

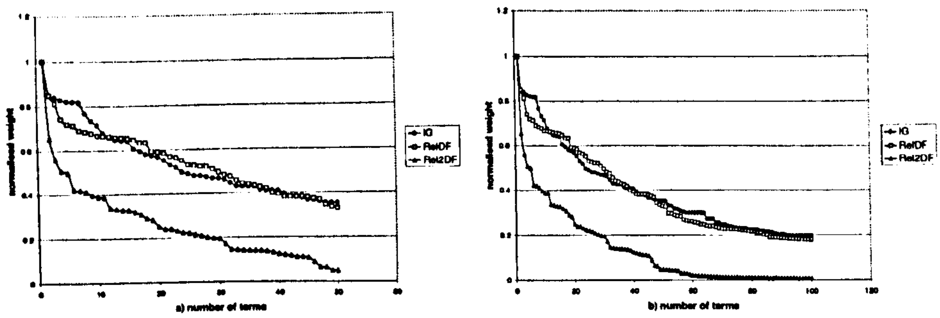

Figure 3-3: Normalised Weight distributions for IG, RelDf and Rel2DF:
a) best 50 terms b) best 100 terms

its competitive performance, ad-hoc variations of RelDF can be formulated. Variations of the form, $a \cdot \frac{r}{R} - b \cdot \frac{n}{N}$ or $(\frac{r}{R})^a - (\frac{n}{N})^b$ may be applied, where a and b define the relative weighting of the two fractions and thus of the corresponding kind of information. Small scale experiments using $(\frac{r}{R})^2 - (\frac{n}{N})^1$, produced some improved results. As depicted in figure 3-3, this variation of RelDF (called Rel2DF) has a much steeper distribution which might be advantageous given the characteristics of the test set. Further research may involve the optimisation of the a and b parameters for specific collections or user characteristics.

Despite its simplicity, the competitive performance of RDF is not surprising. RDF takes into account the important user-provided information. In addition, its results are analogous to those presented by [203], for its m-ary counterpart, document frequency. It is the performance of CHI that is unexpected. CHI is the worst of the four methods from TC. This is possibly due to CHI's m-ary nature. While in an m-ary classification problem it is usually secure to treat documents not pertaining to a certain topic as non-relevant to that topic, we have already noted that in our case not all of the documents in the training set that pertain to a certain topic are used for the construction of the corresponding profile.

Out of the methods from IR, F4 is the best performing one. Its superior performance over F1 confirms the results presented by [149]. IDF and RIDF are the worst performing approaches because they do not explicitly take into account relevance information. It is however interesting to note that RIDF performs slightly better than the rest of the IR methods for small number of extracted terms. This characteristic of RIDF can be attributed to its Poisson distribution component that takes into account the frequency of occurrence of terms in the user specified documents. As a result the user provided information influences to some extent the weighting of terms.

As expected the results using multiplication of weights for document evaluation are worse than those using summation. Nevertheless, in both cases the behaviour of the evaluated methods is analogous both in terms of relative performance and in terms of performance trend. Therefore, both document evaluation approaches confirm the above findings. This is reasonable since both measures are monotonic to the weight of individual terms and to the number of profile terms that appear in a document.

In conclusion, the presented comparative evaluation followed an alternative to TREC's routing subtask that reflects the expected small number of initialisation documents. The results indicate that methods from TC are more appropriate for IF than methods from IR. These methods favour relevance information provided by the user, over information derived from a general collection. Nevertheless, both kinds of statistical information are important, but an appropriate balance between the two is necessary. IG is the best performing approach, while RelDF appears to be a promising and flexible alternative. The results can be used as evidence for the appropriate choice of a term weighting method by IF systems. For the work presented henceforth we have chosen to concentrate on IG and RelDF. In addition the easy reproduction of the experimental setup and the basic document evaluation functions that have been adopted, allow the use of the results as a baseline for comparison with more elaborate IF approaches. In our case, the results have been used in the next chapter as a baseline for evaluating the performance of the hierarchical profile that we are going to built in the rest of this chapter.

3.4 Step 2: Identifying Term Dependencies

Having extracted the most specific terms from the user specified documents, the next step is to appropriately associate them. To identify term associations the context of terms in a document has to be taken into account. As we have described in section 2.7, a term's context can be defined as a span of contiguous words that surrounds the term, called a window. The size of the window defines the kind of associations that we can identify. In contrast to the INFOrmer [174] and PSUN [108] filtering systems that only associate adjacent terms and IR approaches that adopt the complete document context [184, 135, 34, 163], for the current work we have chosen a window of size 10[3] that is larger than the typical size of local context. This topical context allows the identification of term dependencies that are caused by both topical and lexical correlations and that can be expressed by a weighted associative link.

Topical correlations between extracted terms that appear within the window are measured using a formula similar to the one adopted by [135]. In addition, the formula has been extended to measure the lexical correlations

[3] Two different extracted terms can be associated if no more than 9 terms intervene between them. An extracted term can be associated with other extracted terms on either side in text.

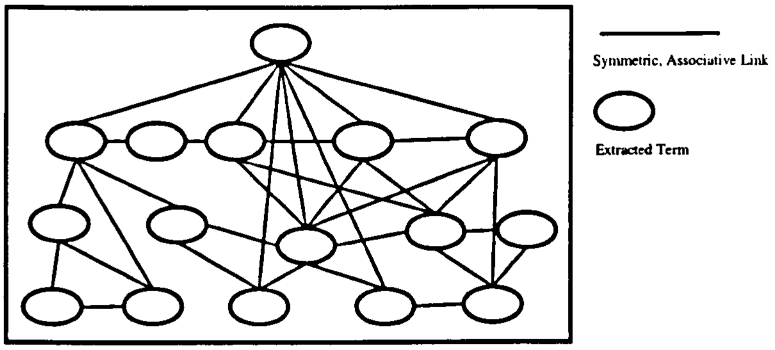

Figure 3-4: Associative links between profile terms. (Nodes (ellipses) represent terms and edges their inbetween links.)

between terms by their average distance in text. More specifically, a weight w_{ij} in the range (0,1] is assigned to the link between two extracted terms t_i and t_j ($t_i \neq t_j$) using the following formula:

$$w_{ij} = \frac{fr_{ij}^2}{fr_i \cdot fr_j} \cdot \frac{1}{d_{ij}} \tag{3.4}$$

In equation 3.4, fr_{ij} is the number of times t_i and t_j appear within the sliding window, fr_i and fr_j are respectively the number of occurrences of t_i and t_j in the user specified documents and d_{ij} is the average distance between the two linked terms. Two extracted terms that appear next to each other have a distance of 1, while if n words intervene between them the distance is $n+1$.

These first two steps result in a symmetric associative graph, like the one in figure 3-4, where nodes represent extracted terms and links their associations. The first fraction of equation 3.4 measures the likelihood that the two extracted terms will appear within the sliding window. The second fraction on other hand is a measure of how close the two terms usually appear. The significance of degree of proximity has been also identified by Luhn, who argued that "ideas most closely associated intellectually are found to be implemented by words most closely associated physically" [101]. As a result of the above formula, a link's weight is a combined measure of the statistical dependencies caused by both lexical and topical correlations. Extracted terms that appear frequently within each other's topical context and/or appear frequently close to each other, are linked with large weights. We should also note that a link weight is not a function of the weights of the constituent terms. Although, such weighting strategies have been investigated, they did not produce uniform results [95].

Finally, an alternative solution could be to assign non-symmetric weights to links using the following equation. Although this weighting strategy could be advantageous and will be considered as part of our future research, for the current work we have focused on symmetric links for computational efficiency.

$$w_{ij} = \frac{fr_{ij}}{fr_j} \cdot \frac{1}{d} \tag{3.5}$$

3.5 Step 3: Generating a Hierarchy

In order to extract a hierarchy out of the associative graph of figure 3-4, a way of identifying topic-subtopic relations between terms is required. Towards this end, we have investigated two alternative approaches. Forsyth and Rada have hypothesised that the more documents a term appears in, the more general the term is assumed to be [53]. In other words, some of the profile terms will broadly define the underlying topic, while others co-occur with a general term and provide its attributes, specialisations and related concepts [93]. According to this hypothesis, terms are ordered according to decreasing relevant document frequency (RDF). The higher a term's rank the more general the term is assumed to be. The problem with this approach is that it does not take into account collection-wide statistics in the ordering .

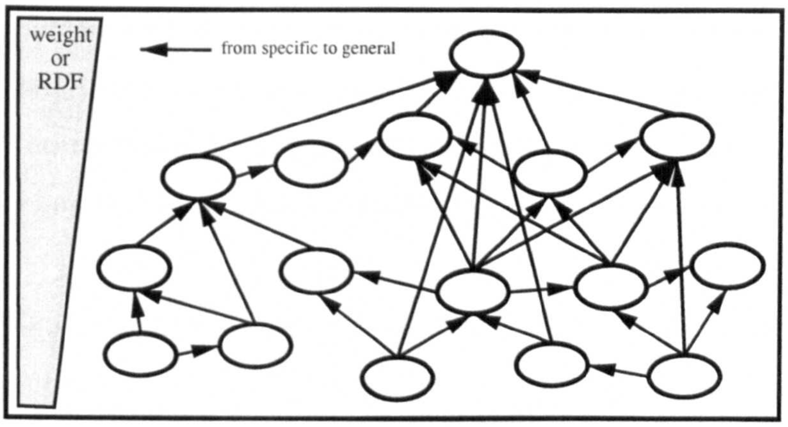

Figure 3-5: Concept hierarchy profile representation

As a result, extracted terms that are frequent in general may be placed at a high rank, although they are not specific to the underlying topic. Alternatively, terms can be ordered according to the weights assigned by either IG or RelDF. This ordering takes into account both the generality of terms within the user specified documents and their specificity within the general collection. If two terms have the same RDF or weight then they are ordered alphabetically. Therefore, there is always a difference between the rank of different terms.

The above process transforms the associative graph of figure 3-4 into a cyclic, hierarchical term network (figure 3-5). Terms at the top of the hierarchy are more specific to the user interests. They correspond to concepts that relate to the general topic of interest. Less specific terms appear in the middle of the hierarchy. These are concepts that relate to subtopics of interest. Finally, at the lowest levels of the hierarchy appear terms that comprise the subvocabulary used when the topic is discussed. If a strong associative link exists between two terms of different rank, then we may refer to such a relation as *topic-subtopic*. In our case, such a relation between terms is not strictly semantic, but rather statistical. Nevertheless, although it is not an issue that we tackle in this thesis, it is possible to show that the generated hierarchy complies with most of the design principles set by Sanderson and Croft for the generation of a concept hierarchy using subsumption [163], and hence it could be applied for the organisation, summarisation and interactive access to information.

However, in contrast to subsumption hierarchies, where the links are generated based on the co-occurrence of terms within the complete document context, the adopted link generation and weighting process combines co-occurrence of terms within topical context, with distance between terms. Overall, we can hence argue that the presented methodology generates a hierarchical representation of the user's interests that tackles all three dependence dimensions. The first step tackles documentation redundancy by employing term weighting to identify those unique terms in the user-specified documents that are strongly correlated to the underlying topic. The second step associates the extracted terms with weighted links that reflect both the topical and lexical correlations between terms. We then distinguish between associations that express topic-subtopic relations by ordering the terms according to their generality in the user specified documents. Therefore, the last two steps in combination, tackle both reality and language redundancy.

3.6 Representing Multiple Topics of Interest

In the previous sections we described the proposed methodology for generating a concept hierarchy out of user-specified documents about a single topic of interest. Nevertheless, the same process can be applied on a single set of documents that relates to multiple topics of interest. As in the case of clustering approaches, the user does not have to categorise the documents according

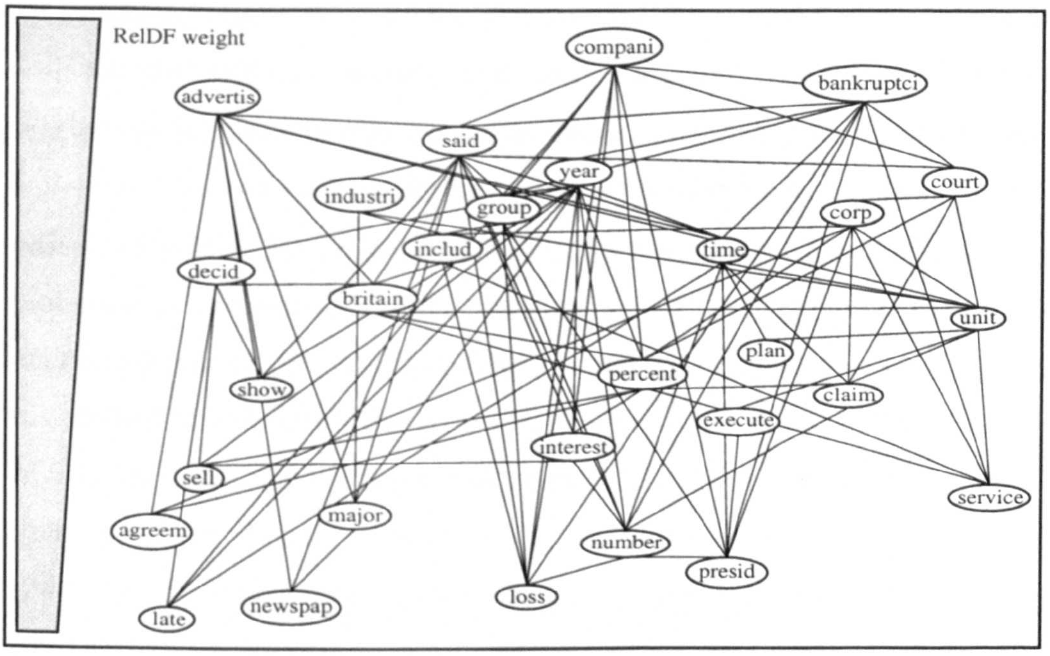

Figure 3-6: Hierarchical network for topics R6 and R21(link weights > 0.01)

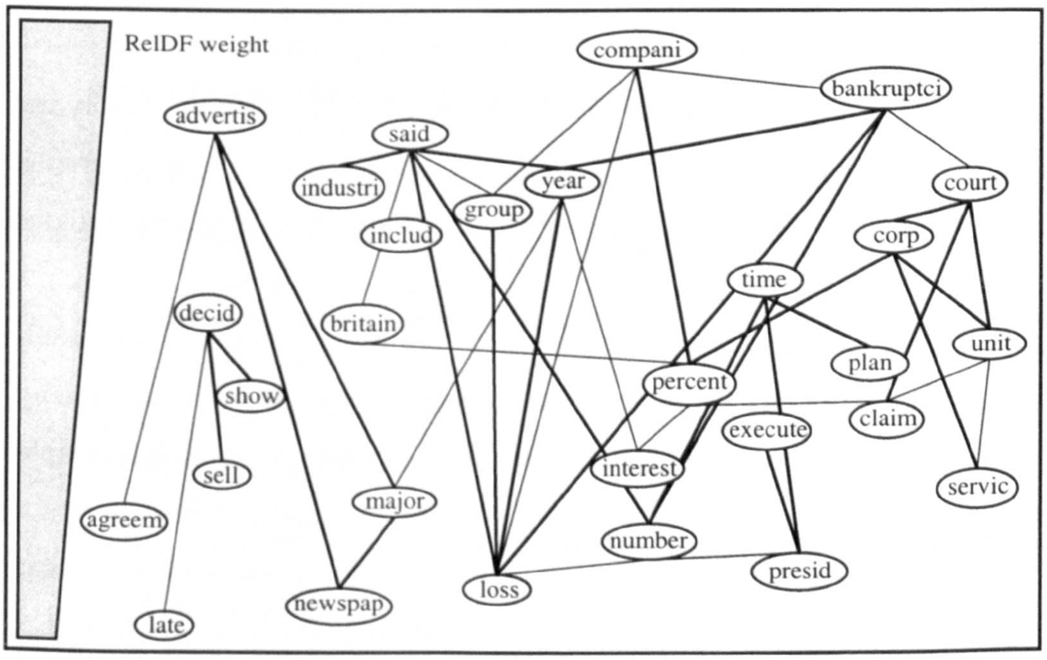

Figure 3-7: Hierarchical network for topics R6 and R21 (link weights > 0.05 & highlighted links weights > 0.1)

to the underlying topics. By following the above three steps, a separate hierarchy is formulated for each general topic discussed in the user specified documents. Terms that are specific to a topic are not related strongly, if related at all, to terms that are specific to a different topic.

Figures 3-6 and 3-7 depict a hierarchical network that was constructed from a set of 60 documents comprising 30 training documents corresponding to RCV1 topic R6 (INSOLVENCY/LIQUIDITY) and 30 training documents corresponding to topic R21 (ADVERTISING/PROMOTION)[4]. The terms in the network are stemmed. For visualisation reasons only links with weight over 0.01 are depicted in figure 3-6. The network is densely interconnected, but it is still possible to recognise the general terms that relate to the underlying topics. Terms *compani* and *bankruptci* are obviously related to topic R6 and term *advertis* to topic R21. Terms *compani* and *advertis* are only linked to terms lower in the hierarchy. Such "dominant" terms can be used to identify the profile's "breadth", i.e. the number of general topics represented.

Figure 3-7 focuses on links with weight over 0.05. We can now identify two separate hierarchies for each one of the topics of interest. Term *advertis* is strongly linked to terms *major*, *newspap* and *agreem*. A much more populated hierarchy is rooted to term *compani*. The number of terms that comprise each hierarchy, defines the hierarchy's "size" which can be used as a measure of the corresponding topic's importance in the profile. Indeed, experiments performed in the next chapter confirm this hypothesis. In this example, the hierarchy that corresponds to topic R6 has a significantly greater size than the hierarchy corresponding to topic R21. Since the same number of documents have been used for each topic, this difference is obviously due to

[4] Appendix A includes a table which summarises the thematic categories and topic codes of all topics involved in our experiments

the characteristics of the training documents.

If we further focus on links with weight over 0.1 (fig. 3-7: highlighted links), strong topical or lexical correlations can now be recognised. Strong topical correlations, like the one between terms *bankrupti* and *loss*, link terms with significant difference in weight. On the other hand, strong lexical correlations link terms of about the same weight level. Examples of this case, include the topical correlations between terms *major* and *newspap*, or terms *execute* and *presid*. The following are examples of phrase fragments extracted from the training documents that demonstrate the validity of what is being represented.

... *full-page **advertis**ements in five **major** Japanese **newspap**ers.*

... *consumer **group** takes ... to **court**.*

... ***said** that the **company** ...*

... *wide **interest**, with some people **advertis**ing in local **newspap**ers ...*

... ***said** Thursday that **advertis**ing revenues for its **Newspap**er Publishing **Group** ... rose 4.3 **percent** from a **year** earlier ...*

... *a 15 **percent** rise in **bankrupt**cies compared to the previous **year***

... ***said**, adding the **group** had expected to have several million dollars of **loss**es in the first four **years** of operation.*

... *Construction **Corp** applied for **court** receivership ... a **court** official **said** on Tuesday*

... *number of **bankrupt**cies among Japanese jewellery firms ...*

... *approval by **bankrupt**cy **court***

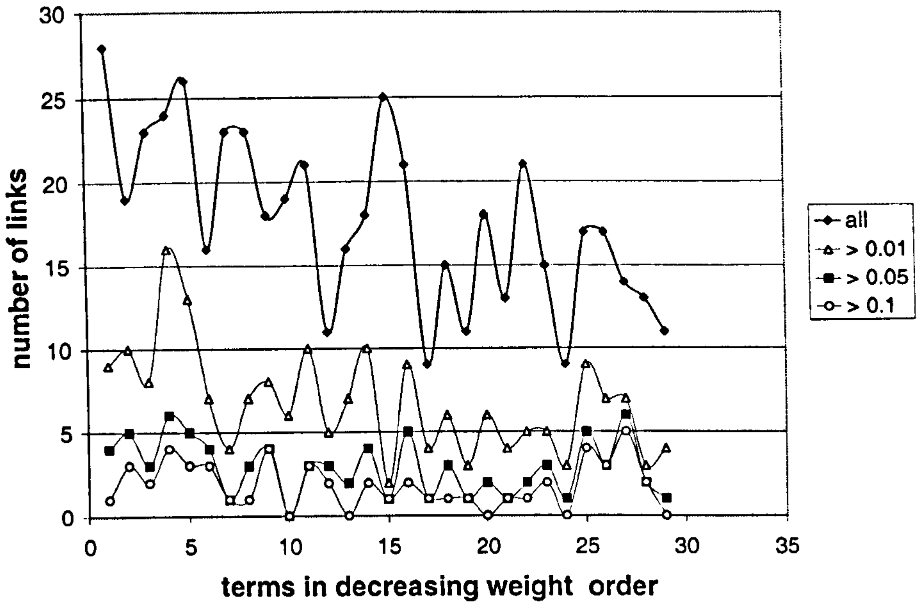

Figure 3-8: Link distributions for different link weight thresholds

Additional and potentially useful conclusions may be derived from the distribution of the number of links over different terms. Figure 3-8 depicts the distribution of links for terms with decreasing weight. When all links are taken into account then there is an obvious overall decrease in the number of links as the weight of terms decreases. Therefore, there is a correlation between weight and number of links. The same is true for links with weights over 0.01. However, the number of links with large weights is more evenly distributed, because strong lexical correlations may exist between terms of the same weight level. Link weights are apparently an ingrained feature of the network's topology. Motivated by the recent interest in the study of networks [13] and their application for language modelling [67, 126], we intend, as part of our future research, to further explore the topological characteristics of the generated hierarchical profile. This may involve experimentation with profiles for which we maintain links with weights over different thresholds. Nevertheless, for the current work we don't apply such thresholds and instead we take all generated links into account.

In conclusion, the proposed methodology tackles all three dependence dimensions to generate a hierarchical network that may represent more than one topic of interest. What is represented is reflected in the network's topology, which can be mined to extract profile characteristics, like the profile's breadth and size. As we will discuss in the next chapter, similar mining can be performed to provide additional evidence of a document's aboutness, which can be used for document evaluation and to support enhanced presentation of the filtering results.

3.7 Summary and Conclusions

In this chapter we have presented a novel methodology for generating a hierarchical network representation of the user's interests from a set of user-specified documents. The methodology consists of a series of three processes that tackle documentation, language and reality redundancy. Initially, term weighting is applied to identify the most specific terms in the specified documents. A large number of existing methods can accomplish such weighting. Nevertheless, existing methods that have been introduced in the context of IR and TC can be affected by the limited availability of relevance information and the potential lack of non-relevance information for profile initialisation. In contrast to the current practices, according to which IF systems choose a term weighting method based on its successful application in IR and TC, we have introduced a new term weighting method, called relative document frequency (RelDF), and we conducted a comparative evaluation that takes into account the above particularities of IF. The evaluation methodology itself is an alternative to the TREC-2001 routing subtask that more accurately reflects the profile initialisation process. The experimental results indicated that methods from TC outperform methods from IR. It was realised that relevance information is of particular importance, but has to be appropriately combined with statistical information derived from a general document collection. IG and RelDF were the best performing approaches and therefore, we will use them henceforth. The results will also be used as a baseline for evaluating our novel IF approach in the next chapter. Similar use of the results can be made by other IF research systems.

Having identified the most appropriate term weighting methods for the task at hand, the next step in the methodology involved the identification and weighting of the dependencies between extracted terms. We have avoided document context and instead adopted a topical context that allows the identification of associations that are caused by both topical and lexical correlations. We introduced a novel link weighting function that is a combined measure of the statistical dependencies that are caused by both types of correlations. This process has resulted in an associative graph that is finally transformed into a hierarchy by ordering the terms according to decreasing generality in the user-specified documents. In this way, we were able to identify topic-subtopic relations between terms and we have therefore argued that the generated hierarchy can be applied in the same way as a concept hierarchy. We finally described how more than one topic of interest can be represented by the proposed hierarchical profile, given a single set of documents about these topics. We now turn to the issue of using the hierarchical profile for non-linear document evaluation.

Chapter 4

Non-Linear Document Evaluation

IF has traditionally been approached as a specialisation of IR and TC. The most serious consequence of this tendency is that most IF systems adopt profile representations that ignore term dependencies (section 2.10.1). Both the vector space model and linear classifiers have been popular. Such representations however, can only support linear evaluation functions which can estimate a document's relevance to a single topic of interest. To represent multiple topics of interest, a separate profile is usually built for each individual topic. Finally, in the case of evolutionary IF systems a population of linear profiles is maintained that collectively represent the multiple user interests. The connectionist profile representations that have been recently proposed as part of the INFOrmer [174] and PSUN [108] filtering systems, are exceptions to the above rule in that lexical correlations between terms are being represented. Nevertheless, both systems use a separate profile for each topic of interest. We argue that multi-topic representation requires a profile that tackles all three dependence dimensions and that distinguishes topic-subtopic relations between terms.

In the previous chapter, we introduced a methodology for building a hierarchical profile representation that complies with the above requirement. The generated profile could be used as a concept hierarchy for the organisation, summarisation and interactive access to information. We have also described how multiple topics of interest may be represented and how evidence of what is being represented can be derived.

In the current chapter, we address how to use this profile representation for document evaluation. Our challenge was to establish a non-linear evaluation function that is able to assess the relevance of documents according to a multi-topic profile representation. We have drawn ideas from the application of neural networks [88, 199] and semantic networks [40, 15, 76] to IR. As we have already described (section 2.6.1), according to these connectionist approaches, documents, queries and index terms are represented by nodes. Terms that are contained in a document or a query are linked to the corresponding nodes. Links between terms and between documents are ignored, which leads to a linear evaluation function through dissemination of an initial query energy towards the documents. A document's relevance is finally estimated as the total amount of energy that a document received. Spreading activation functions have also been applied in the case of associative graphs that express the lexical [174] or stochastic [34] correlations between terms. Due to the inherent lack of direction in these networks, an initial energy is assigned to the terms that appear in a specific document and is then iteratively disseminated through the network until an equilibrium is reached. We have investigated two alternative approaches (a layered and a continuous approach) coupled with directed spreading activation models that establish non-linear document evaluation functions. The functions take into account the dependencies that the profile represents to assign a single relevance score to each document. In order to evaluate the performance of these approaches, we have conducted single- and multi-topic experiments, which produced promising results.

However, in section 2.11, we have argued that ranking documents according to a quantitative relevance score is not sufficient for multi-topic IF. It should be complemented with additional evidence of a document's aboutness. So in addition, we suggest how such evidence can be provided and how it may be used for presenting the filtering results to the user. Finally, we discuss additional personalisation services that can be supported by the proposed hierarchical profile.

4.1 Layered Approach

Our first attempt to establish a non-linear evaluation function on the hierarchical profile followed a layered approach. The hierarchy is generated by ordering terms according to decreasing RDF (section 3.5). The terms are then assigned to three layers according to RDF thresholds. Forsyth and Rada adopt a similar grouping of words according to frequency ranges, for constructing a decision tree [53]. For our experiments (section 4.1.1) we used the following thresholds: if $RDFmax$ is the RDF of the most frequent profile term, then profile terms with $RDF \geq 0.8 * RDFmax$ are assigned to the top layer, those with $RDF \geq 0.4 * RDFmax$ to the middle layer and the rest of them to the third, lowest layer (fig. 4-1). Since profile terms are only the most specific terms in the user specified documents (see section 3.2) and due to their distribution, very few terms are assigned to the top layer, some more to the middle and the majority to the lowest layer. Terms in the top layer

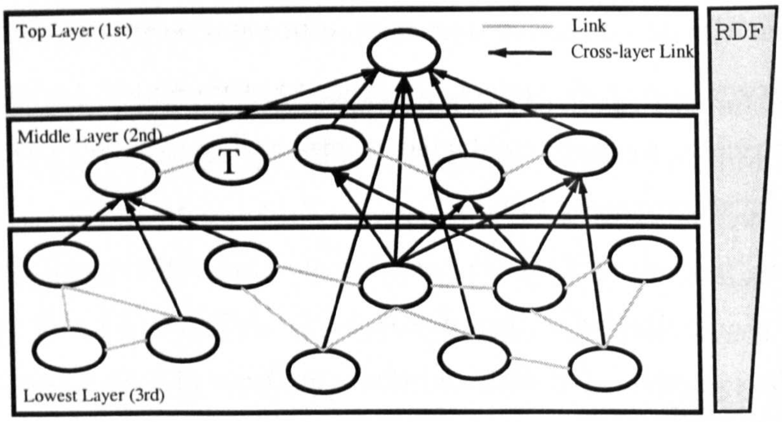

Figure 4-1: Layering of hierarchical profile representation

define the general topic of interest, terms in the middle layer correspond to related subtopics and terms in the lowest layer comprise the subvocabulary used when the topic is discussed. Links that cross layer boundaries connect general terms in the top layers with more specific terms in the layers below (fig. 4-1: links in black). In other words, cross-layer links reflect topic-subtopic relations between terms. This division allows the implementation of the following document evaluation function.

Given a document D, an activation of 1 (binary document indexing) is passed to those profile terms that appear in D. The initial energy of an activated term t_i, is $E_i = 1 \cdot w_i$, where w_i is the weight that has been assigned to the term by the term weighting method (RelDF or IG). If and only if, an activated term t_i is directly linked to another activated term t_j in a higher profile layer, then an amount of energy E_{ij} is disseminated by t_i to t_j through the corresponding link. $E_{ij} = E_i \cdot w_{ij}$, where w_{ij} is the weight of the link between t_i and t_j. A direction from lower to higher profile layers is thus imposed on the cross-layer links (fig. 3-5: arrows). Activated terms in the top two layers update their initial energy by the amounts of energy that they receive from activated terms in the layers below. For example, the updated energy E'_j of term t_j is $E'_j = E_j + \sum_{t_i \in A^l} E_i \cdot w_{ij}$, where A^l is the set of activated terms that are directly linked to t_j and that appear in lower profile layers. Terms in the middle layer update their energies first and terms in the top layer last (feedforward). As a result, activated terms in the middle layer disseminate part of their "updated" energy. If A^h is the set of activated terms in the top two layers that have either received or disseminated energy, then the document's score is calculated by equation 4.1, where NT is the number of words in D. Only terms in A^h are thus allowed to directly contribute to the document's relevance. This constraint is a drawback of this layered approach for two reasons. First, according to this constraint, term T in figure 4-1 does not contribute to the document's relevance, despite representing a subtopic of interest. This can negatively affect the filtering performance, especially for profiles with a small number of terms and consequently of links. In addition, the current document evaluation function favours terms in the top two layers. Activated terms in the lowest layer contribute only implicitly by the energy that they disseminate to activated terms in the layers above.

$$SL_D = \frac{\sum_{t_i \in A^h} E'_i}{log(NT)} \tag{4.1}$$

The above establishes a non-linear document evaluation function that takes into account the term dependencies that the concept hierarchy represents. As energy disseminates from lower to higher hierarchical layers, terms in the higher profile layers that appear in the document together with their associated terms in the layers below, have an increased contribution to the document's score. Therefore, a document about both topics and related subtopics of interest, receives a higher score than a document about the same number of unrelated topics. On the other hand, terms in the lowest profile layer disseminate their energy, only if they are found together with their associated topics and sub-topics of interest, in the layers above. In other words, these terms contribute implicitly to the document's relevance if they are found in the topical context that they were extracted from. Topical correlations between terms are thus taken into account.

Lexical correlations are also taken into account due to the way links are weighted. The amount of energy disseminated between two terms of different profile layers, is larger if they are part of a lexical compound and therefore appear frequently close to each other. On the other hand, if the compound's terms are found in the same layer, then their contribution to the document's score can be enhanced because of the large number of links that they probably have in common. Terms that are found in adjacent positions in text share a lot of links to other terms.

In addition to taking into account both topical and lexical correlations in the way documents are evaluated, the proposed approach is also computationally cheap. In contrast to traditional spreading activation approaches, where numerous computational cycles are required for the network to reach an equilibrium, document evaluation takes place in two forward steps; from the lowest to the top two layers and from the middle to the top layer.

4.1.1 Experimental Evaluation

To evaluate this layered approach to document evaluation we have performed single-topic experiments using a methodology similar to the one described in section 3.3.1. The only difference is that for the current experiments, only the first 30 training documents per topic were allowed for the corresponding profile's initialisation. This allowed the direct comparison with the results of that previous experiment where term independence was assumed. Despite the differences of our experimental methodology to the TREC routing guide-

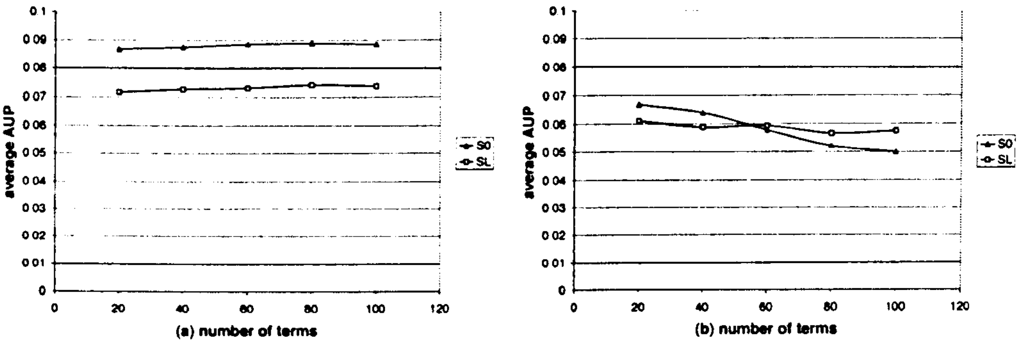

Figure 4-2: Experimental Results for (a)IG and (b)RelDF using the layered approach. The x-axis presents the number of extracted terms and the y-axis the average AUP score over all 10 topics.

lines, comparisons to the TREC results for the first 10 topics have also been made.

More specifically, for each one of the first 10 RCV1 topics a profile was constructed using the best terms on the basis of IG or RelDF weighting. We have experimented with different numbers of extracted terms. These could be 20, 40, 60, 80 and 100. For each possible combination of topic and number of extracted terms two different kind of profiles were constructed. In the first case, no links between terms were generated (unconnected profile version) and documents were evaluated using the inner product based function S0 (equation 3.2). In the second case, the same extracted words were used to generate the layered hierarchical profile (extracted terms were ordered according to decreasing RDF) and documents were evaluated using the above document evaluation function SL (equation 4.1).

Figures 4-2(a) 4-2(b) present for IG and RelDF respectively, the average AUP score over all 10 topics for the unconnected and hierarchical profile version. The x-axis corresponds to the number of profile terms and the y-axis presents the average AUP score over all 10 topics. Table 4.1 presents for each topic, the average AUP score over the different numbers of profile terms. Table 4.2 presents for each topic, the maximum achieved AUP score. This table is complemented with a column that presents the average maximum AUP score achieved by the eighteen participants of TREC 2001 routing subtask[1]. In both tables highlighted scores indicate for each term weighting approach the best score achieved by either the unconnected (S0) or connected (SL) profile. Table 4.2 includes a "+" sign on the right of scores, achieved by connected profiles, that are greater than the average maximum score of the TREC 2001 participants.

In the case of IG weighting, the results for SL are worse than those for S0 (fig. 4-2(a)). This is due to the drawbacks of the above layered approach to document evaluation. For each of the topics, the hierarchical profile's performance is strongly dependent on which terms are assigned to the top two layers. As indicated by table 4.1 (highlighted scores) for only 3 out of the ten topics is the distribution of terms in the three layers successful, resulting in better overall performance for SL. Furthermore, according to table 4.2, SL with IG achieves the best maximum score for only topic R8 out of the ten topics.

In accordance with the results of the experiments described in section 3.3, the performance of SL with RelDF weighting, is in general worse than for IG (fig. 4-2 (a) and (b)). Its performance however is comparable to the corresponding unconnected profile version S0 (fig. 4-2(b)). The hierarchical profile performs better for large numbers of profile terms. In addition, table 4.1 (highlighted scores) shows that for 6 out of the 10 topics SL has a better overall performance over S0. These relatively positive results however, are not in fact sufficiently satisfactory. In section 3.3.2, we have argued, that given the large number of training documents, the change in their content

[1] Many thanks to Ellen Voorhees for providing us with the TREC routing results.

Table 4.1: Per Topic Average AUP for Layered Approach

	IG weighting		RelDF weighting	
Topic	S0	SL	S0	SL
R1	**0.00155**	0.00133	0.00148	**0.0028**
R2	0.06769	**0.06834**	0.0667	**0.06723**
R3	0.00263	**0.00507**	**0.00193**	0.0017
R4	**0.05306**	0.03863	0.01738	**0.03272**
R5	**0.01867**	0.00408	**0.01482**	0.006
R6	**0.25452**	0.19855	0.07184	**0.0905**
R7	**0.03889**	0.02877	**0.032**	0.03176
R8	0.06874	**0.06936**	0.07115	**0.072**
R9	**0.2001**	0.14	**0.12967**	0.1064
R10	**0.17461**	0.17398	0.17488	**0.1757**

Table 4.2: Per Topic Maximum AUP for Layered Approach

	IG weighting		RelDF weighting		TREC
Topic	S0	SL	S0	SL	av. max.
R1	**0.00191**	0.001385	0.00207	**0.00289**	0.0143
R2	**0.07069**	0.069+	0.0699	**0.07006+**	0.0484
R3	**0.035**	0.00594	**0.00335**	0.00186	0.0105
R4	**0.05367**	0.03898	0.02623	**0.03474**	0.0516
R5	**0.019797**	0.00528	**0.01888**	0.00707	0.0237
R6	**0.25746**	0.19958+	**0.11234**	0.09776	0.1719
R7	**0.04024**	0.02927	**0.03431**	0.03244	0.0338
R8	0.06972	**0.07126+**	**0.07503**	0.07492+	0.0633
R9	**0.20172**	0.14707	**0.15996**	0.11983	0.1667
R10	**0.17516**	0.17498+	0.17521	**0.17584+**	0.1519

over time and the limited vocabulary provided by the small number of initialisation documents, it is advantageous to favour only the most general, exhaustive terms. Layering in the case of RelDF has the positive effect of favouring the few general terms and smoothing the effect of the possibly overestimated majority of less general terms. Therefore, in the case of RelDF, the characteristics of the training set may favour the layering of terms. Nevertheless, table 4.2 indicates that SL achieves the best maximum score for only 4 out of the 10 topics (table).

Finally, the comparison with the average TREC results shows that SL with IG performs better than the average TREC participant for 4 out of 10 topics and SL with RelDF for 3 out of 10 topics (table 4.2: scores indicated with "+" sign). These findings are acceptable if one considers that our experimental methodology does not favour our approach. No optimisations of parameters, like layering thresholds and link weights, have been performed. More importantly, we have only used 30 training documents per topic, in contrast to an average of 324.3 documents per topic that are available, according to TREC routing guidelines, for training profiles for the first 10 RCV1 topics[2].

In summary, in this section we have described a layered approach towards establishing a non-linear document evaluation function on the introduced hierarchical profile representation. Although, theoretically, this approach takes into account the dependencies that the profile represents, we have identified a couple of drawbacks that mainly derive from the layering itself. The discretisation that the layering causes, leads to unevenly distributed term importance. The few general terms that are assigned to the top two layers have a

[2] Appendix A includes a table which summarises the thematic categories and topic codes of all topics involved in our experiments

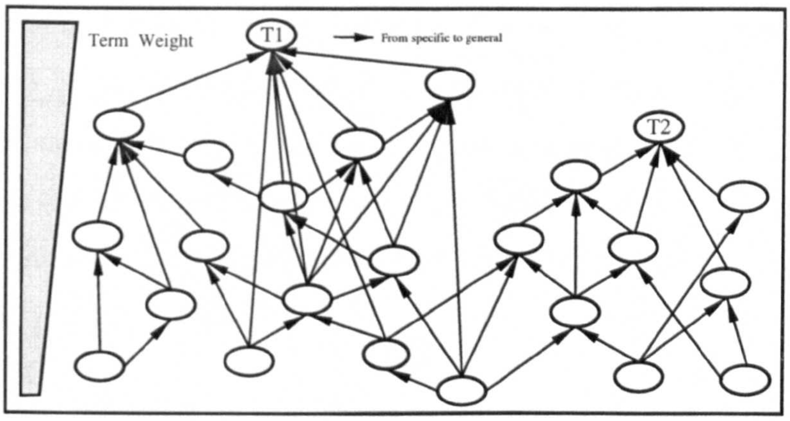

Figure 4-3: Hierarchical Profile Representing Two Topics of Interest. (Dominant Terms T1 and T2.)

more significant contribution to a document's relevance score. Although, the large number of test documents per topic and the small number of training documents render this behaviour advantageous, especially for large numbers of profile terms, this solution did not satisfy us. A possible avenue might have been to fine tune the layering thresholds. However the results did not justify pursuing this course, not least because we are aiming for multi-topic profile and optimum thresholds may be topic dependent. Instead we turned to the exploration of a continuous approach described in the next section.

4.2 Continuous Approach

The disadvantages of the above layered approach motivated us to pursue a continuous approach. In contrast to the layered approach, terms are ordered according to decreasing IG or RelDF weight. So the ordering takes into account both the generality of terms and their specificity in the complete collection. Figure 4-3 depicts a generalised hierarchical profile constructed from a set of documents about two overlapping topics. The two topics are

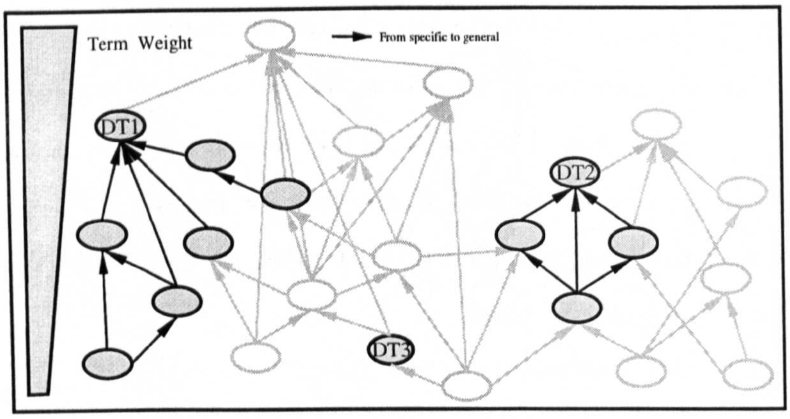

Figure 4-4: Activated Profile. (Dominant Terms T1, T2 and T3.)

reflected by two hierarchical subnetworks that share a small number of common terms. Each hierarchy is rooted to a dominant term (terms T1 and T2) which is only linked to terms with lower weights. The profile's breadth is therefore equal to 2. Each hierarchy's size on the other hand can be measured by counting the number of terms that are explicitly or implicitly linked to the corresponding dominant term. T1's hierarchy has a size of 14 and T2's a size equal to 10. To use such a continuous hierarchical profile for document evaluation we introduce three non-linear document evaluation functions, based on a slightly different spreading activation model.

Given a document D, an initial energy of 1 (binary document indexing), is deposited with those profile terms that appear in D. In figure 4-4, activated terms are depicted by shaded nodes. Subsequently, energy is disseminated sequentially, starting from the activated term with the smallest weight and moving up the weight order. If, and only if, an activated term t_i is directly linked to another activated term t_j higher in the hierarchy, then an amount of energy E_{ij} is disseminated by t_i to t_j through the corresponding link. E_{ij} is defined by equation 4.2, where E_i^c is t_i's current energy, w_{ij} is the weight of the link between t_i and t_j, and A^h is the set of activated terms higher in the hierarchy that t_i is linked to. The purpose of the normalisation parameter $\sum_{k \in A^h} w_{ik}$ is to ensure that a term does not disseminate more than its current energy. The current energy of term t_i is $E_i^c = 1 + \sum_{m \in A^l} E_{mi}$, where A^l is the set of activated terms lower in the hierarchy that t_i is linked to. After the end of the dissemination process the final energy of a term t_i is $E_i^f = E_i^c - \sum_{k \in A^h} E_{ik}$.

$$E_{ij} = \begin{cases} E_i^c \cdot w_{ij} & \text{if } \sum_{k \in A^h} w_{ik} \leq 1 \\ E_i^c \cdot (\frac{w_{ij}}{\sum_{k \in A^h} w_{ik}}) & \text{if } \sum_{k \in A^h} w_{ik} > 1 \end{cases} \tag{4.2}$$

We have experimented with three different ways for assessing a document's relevance score S_D, based on the final energy of activated terms. The simplest variation is defined by equation 4.3, where A is the set of activated profile terms, w_i is the weight of an activated term t_i and NT the number of terms in the document.

$$S1_D = \frac{\sum_{i \in A} w_i \cdot E_i^f}{log(NT)} \tag{4.3}$$

The above process establishes a non-linear document evaluation function that takes into account the term dependencies reflected in the concept hierarchy. Its effect can be demonstrated with the following example. Consider the simple case of a document that has activated two profile terms t_1 and t_2, with $w_2 > w_1 > 0$. If the terms are not connected, then no dissemination takes place, and so the final energy of the terms equals their initial energy. The document's relevance would then be $S_D^u = (1 \cdot w_1 + 1 \cdot w_2)/log(NT)$. This implies that equation 4.3 specialises to the inner product (equation 3.2), if links between terms are ignored. On the other hand, if the terms were connected, then their final energy would be $E_1^f = 1 - (1 \cdot w_{12})$ and $E_2^f =$

$1 + (1 \cdot w_{12})$ respectively. Since $E_2^f > 1 > E_1^f$ and $w_2 > w_1$ it is obvious that $S_D^c = (E_1^f \cdot w_1 + E_2^f \cdot w_2)/log(NT)$ is greater than S_D^u. So if two terms are linked by a topic-subtopic relation they contribute more to the document's relevance than two isolated terms with the same weights. The difference in the contribution is analogous to the weight of the link between the terms, which measures the statistical dependence caused by both topical and lexical correlations.

The overall effect is visible in figure 4-4. Activated profile terms define subhierarchies for each topic of interest discussed in the document. The dominant terms $DT1$, $DT2$ and $DT3$ can be defined as those activated terms that did not disseminate any energy. The number of dominant terms measures the document's breadth b, i.e. the number of interesting topics discussed in the document. For each dominant term, the size of the corresponding subhierarchy is equal to the number of activated terms from which energy was received. The document's size d can thereafter be approximated as the number of activated terms that disseminated energy. Obviously, $b + d = a$, where a is the total number of activated terms. The total amount of energy that a subhierarchy contributes to a document's relevance, amounts to its size, and the weight of the terms and links involved. A document's relevance increases if it activates profile terms that formulate connected subhierarchies with large sizes, and not isolated profile terms. In this latter case, the document's breadth increases without a corresponding increase in size. $DT3$ is an example of an isolated term.

On these premises, we also experimented with two normalised versions of the initial function, that explicitly take into account the above measures. The first is defined by equation 4.4. Here, the document breadth is used to normalise the document's score. The idea is to penalise documents that activate a large portion of unconnected terms. In the second case, the document's score is multiplied by the factor $log(1 + (b + d)/b)$ which favours documents with large sizes and small breadths (eq. 4.5). Logarithmic smoothing is applied to avoid very large document scores.

$$S2_D = S1_D \cdot \frac{1}{b} \tag{4.4}$$

$$S3_D = S1_D \cdot log(1 + \frac{b+d}{b}) = S1_D \cdot log(1 + \frac{a}{b}) \tag{4.5}$$

These last two methods demonstrate how evidence derived by mining the topology of the activated subnetwork can be exploited for document evaluation. Other approaches, that in addition take into account evidence like the breadth and size of the complete profile hierarchies, may also be explored. In general, the above document evaluation functions do not represent the only possible solutions. The proposed hierarchical profile representation may support a whole new domain of functions that require further research. For example, instead of treating each evaluated document as a "bag of words", which is contradictory to the way the profile has been generated, the same sliding window could be employed, or a separate score could be calculated for individual sentences, or preferably, paragraphs. These individual scores may be combined into a single document score, or be used to provide evidence of the distribution of relevance throughout the document. Although this appears to be a promising alternative we have not explored it further as part of our PhD work, mainly for efficiency reasons. Nevertheless, it is an interesting direction for our future research.

4.2.1 Single-Topic Experiments

To evaluate this continuous approach we have initially performed experiments using the methodology described in section 4.1.1. This allowed the direct comparison of the continuous with the layered approach. More specifically, experiments were performed for topics R1-R10[3]. For each one of the 10 topics a separate profile was constructed using the best terms on the basis of IG or RelDF weighting. We have experimented with different numbers of extracted terms. These could be 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 40, 60, 80 and 100. For each possible combination of topic and number of extracted terms two different kind of profiles were constructed. In the first case, no links between terms were generated and documents were evaluated using the inner product based evaluation function S0 (equation 3.2). In the second case the same extracted words were used to generate the continuous hierarchical profile and documents were evaluated using S1 (equation 4.3). S2 and S3 have not been evaluated in this experiment. Since S1 specialises to S0 if links between terms are ignored, any difference in performance is due to the representation of term dependencies by the hierarchical profile. Once more, we also compare our results to the results of the TREC 2001 routing subtask, despite the differences in the experimental methodology.

Figure 4-5 (a) and (b) present for IG and RelDF, the average AUP score over all 10 topics for each number of profile terms. Table 4.3 presents for each topic, the average AUP score over the different numbers of profile terms. Table 4.4 presents for each topic, the maximum achieved AUP score and is complemented with a column that presents the average maximum AUP score achieved by the eighteen participants of TREC 2001 routing subtask. In

[3] Appendix A includes a table which summarises the thematic categories and topic codes of all topics involved in our experiments

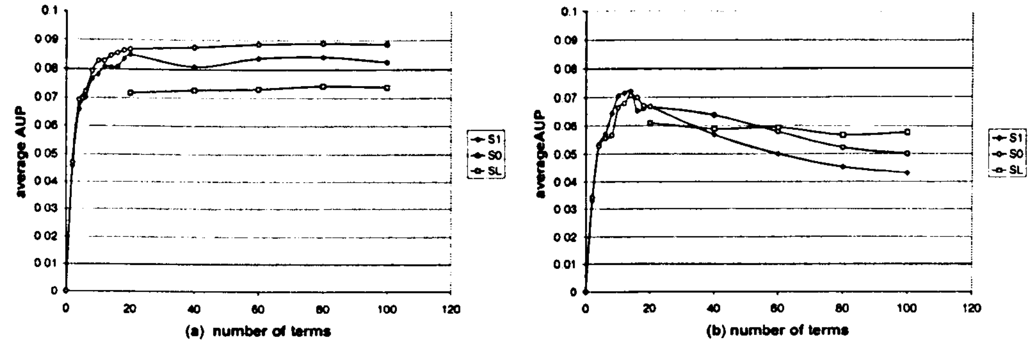

Figure 4-5: Experimental Results for (a)IG and (b)RelDF using the continuous approach. (The x-axis presents the number of extracted terms and the y-axis the average AUP score over all 10 topics.)

these tables highlighted scores indicate for each term weighting approach the best score achieved by either the unconnected (S0) or connected (SL) profile. Table 4.4 includes a "+" sign on the right of scores, achieved by continuous hierarchical profiles, that are greater than the average maximum score of the TREC 2001 participants.

For IG, the results for S1 are better than the corresponding layered profile (SL: fig. 4-5(a)). Nevertheless, they are still worse than the unconnected profile using S0, especially for large number of extracted terms. Though, as indicated by table 4.3, for most of the topics S1 and S0 have comparative performance. S1 has a better overall performance for 4 out of the 10 topics (table 4.3), but it achieves the best maximum score for only 2 out of the 10 topics (table 4.4). Its average performance is significantly worse only for topic R5 (table 4.3). What distinguishes R5 is that although it is relatively specific (Reuters-code C1511 [152]), it has a large number of documents in the test set. As already mentioned, in this case it may be preferable that the most general, exhaustive terms are favoured. Indeed, for topic R5, S0 achieves the maximum performance for only 4 profile terms. This is a representative example of the effect of the large number of test documents per topic on the experimentation results. Consequently the additional information that the hierarchical network encodes, especially in the case of a large number of profile terms, is not necessary and may have a negative effect on the filtering performance.

As expected the results for RelDF are once more worse than those for IG (fig. 4-5(b)). Nevertheless, in this case, S1 achieves the maximum overall performance and is in general better than S0 and the layered approach SL for small number of profile terms. According to table 4.3, S1 has a better overall score for 4 out of the 10 topics, while table 4.4 indicates that S1 achieves the best maximum score for the same 4 topics. For large number of profile terms the performance drops significantly and becomes worse than both S0 and SL. This confirms our previous insight that the experimental setting does not reflect a situation where a lot of profile terms are necessary for representing the user interests. In addition to the large number of test documents, most of the documents are relatively short and with a very focused subject. A few informative terms are sufficient for representing the general topic of interest, which given the large number of test documents may result in increased performance [146]. A hierarchical profile with a large number of terms that represents both topics and subtopics of interest is not necessary and on the contrary, given the small number of training documents, it may result in a densely interconnected network which includes terms and term relations that are not informative. Once more, the hierarchical profile's poor performance for topic R5 contributes to this argument.

This time, the comparison to the TREC results is more promising. Despite the fact that no optimisation took place and, instead of hundreds, only 30 training documents per topic were used, table 4.4 indicates (scores marked with a "+" sign) that for both IG and RelDF weighting continuous hierar-

Table 4.3: Per Topic Average AUP for Continuous Approach

	IG weighting		RelDF weighting	
Topic	S0	S1	S0	S1
R1	0.00121	**0.00137**	0.001334	**0.00146**
R2	0.0516	**0.0518**	**0.054650**	0.05436
R3	**0.0017**	0.0013	**0.001577**	0.00102
R4	**0.0425**	0.035	**0.023031**	0.02162
R5	**0.02028**	0.00496	**0.010835**	0.00547
R6	**0.2236**	0.2213	0.091811	**0.10758**
R7	**0.02805**	0.024	**0.025812**	0.02113
R8	0.06577	**0.0658**	0.064635	**0.06504**
R9	**0.1568**	0.15213	**0.119017**	0.10394
R10	0.16197	**0.1621**	0.162362	**0.16259**

Table 4.4: Per Topic Maximum AUP for Continuous Approach

	IG weighting		RelDF weighting		TREC
Topic	S0	S1	S0	S1	av. max.
R1	0.00191	**0.00206**	0.00207	**0.00275**	0.0143
R2	**0.07069**	0.06902+	**0.06994**	0.06927+	0.0484
R3	**0.0035**	0.00289	**0.00335**	0.00178	0.0105
R4	**0.05367**	0.04591	**0.04067**	0.03703	0.0516
R5	**0.03704**	0.00975	**0.02165**	0.012	0.0237
R6	**0.25746**	0.25595+	0.14891	**0.17695+**	0.1719
R7	**0.04024**	0.03460+	**0.03441**	0.03426+	0.0338
R8	**0.07566**	0.0755+	0.07787	**0.07826+**	0.0633
R9	**0.20173**	0.19879+	**0.17822**	0.17608+	0.1667
R10	0.17529	**0.17587+**	0.17521	**0.17573+**	0.1519

chical profiles perform better than the average TREC participant for 6 out of the 10 topics. These include the last 4 topics (R7 to R10) which are relatively specific and correspond, on average, to a smaller number of available training documents (see Appendix A).

Overall, these single topic experiments indicated that the continuous approach represents an improved solution, over the layered approach, to the problem of establishing a document evaluation function on the hierarchical profile. Nevertheless, despite some positive results in the case of RelDF weighting, the hierarchical profile using the basic evaluation function (S1) does not outperform the unconnected profile (S0). We have attributed these results to the characteristics of the test set in combination with the small number of initialisation documents. For most of the topics just a few exhaustive terms may result in good filtering performance, which renders a densely interconnected hierarchical profile containing a large number of terms, inferior. Despite the small number of training documents, the comparison to the TREC results has been promising and although it has not been our primary goal, it prompts further investigation.

4.2.2 Two-Topic Experiments

So far we have performed single-topic experiments where a different profile is built for each of the first 10 RCV1 topics. However, one of the main issues that we try to address is document evaluation by a single multi-topic profile. But, as already mentioned, the shortage of single, multi-topic profile representations has unfortunately been coupled with a lack of appropriate evaluation methodologies. We have attempted to establish such a methodology using yet another variation of the TREC-2001 routing subtask, which, as already discussed in section 2.14.2 adopts the RCV1 corpus, a collection

Table 4.5: Two-topic combinations: codes and collection statistics

Comb.	Topics	Code	Training	Test
	R1	C11	597	23651
I	R2	C12	351	11563
	R7	C171	403	17876
II	R8	C172	251	11202
	R29	E12	630	26402
III	R68	GJOB	419	16770
	R10	C174	212	5625
IV	R32	E131	140	5492
	R6	C16	42	1871
V	R21	C32	39	2041
	R41	E311	35	1658
VI	R79	GWELF	42	1818

of 806,791 English language news stories. RCV1 is split into 23,864 training and 782,927 test stories and is categorised into 84 topic categories. According to the TREC guidelines a separate profile is built for each of the topics using the complete set of available training documents.

Instead, to simulate the statistical and semantic characteristics of multi-topic interests we experimented with profiles trained for combinations of two and three topics. In this section, we concentrate on the two-topic experiments. Of course a very large number of such combinations can be synthesised out of the 84 TREC topics. We synthesised six combinations with topics of different topical proximity and level, based on their assigned topic codes and collection statistics (table 4.5). Simply put, RCV1 topics are indicated by a code comprising a letter identifier and two digits (e.g. C11), while subtopics are indicated by a code comprising a letter identifier and three digits (e.g. C171) [152]. Related topics and subtopics share a common initial code substring (e.g. C11 is related to C12)[4]. So, combination I com-

[4] Appendix A includes a table which summarises the thematic categories and topic codes of all topics involved in our experiments.

prises two related topics, combination II two related subtopics, combination III two unrelated topics and combination IV two unrelated subtopics. For combinations IV, and especially V and VI, we have deliberately chosen respectively topics and subtopics with a small number of test documents, for reasons explained shortly.

A single profile was built for each of the above combinations. The training set contained only the first 30 training documents for each of the topics involved (a total of 60 documents)[5]. Another difference to what has been done so far is that instead of extracting an absolute number of the most informative terms, term selection was based on a weight threshold. Terms with weight over the threshold were selected to populate the profile. Therefore the number of extracted terms depends on the characteristics of the training set.

We have experimented with different threshold values. For IG these were 0, 0.003, 0.006, 0.0075, 0.009 and 0.011. For RelDF, the thresholds were 0, 0.1, 0.15, 0.2, 0.25, 0.3. These values were selected, based on empirical observations during the previous experiments, so that they correspond as far as possible to similar number of extracted terms for both IG and RelDF. However, as we will shortly see, this goal was not met successfully and as a result the comparison between IG and RelDF was not straightforward.

For each topic combination and threshold value, the selected terms were used to construct an unconnected profile that evaluates documents using S0 and connected hierarchical profiles that use the proposed methods (S1, S2 and S3). It is important to stress at this point, that since all four profiles are constructed using the same set of weighted terms, any difference in their performance can only be attributed to the way links (term dependencies) are taken into account during document evaluation. Hence, we use a linear

[5] The user does not have to categorise the documents.

multi-topic profile as a benchmark for evaluating the effect of taking into account term dependencies in a multi-topic filtering problem. Another possible benchmark for future evaluations could be separate, linear single-topic profiles, one for each topic of interest. Although we have argued that using multiple single-topic profiles has certain disadvantages (section 2.10.1), it could allow a direct comparison to current practices. Each single topic profile could be used to generate a separate ordered list of documents. After normalisation of the document scores the lists would then be merged into a single ordered list which could be directly compared to the ordered list that our single multi-topic profile produces.

Profiles were tested against the test set and evaluated on the basis of the best 3000 documents. A separate AUP score was computed for each topic in the combination. The combination's overall score can then be calculated as the average AUP of its constituent topics. A topic's absolute AUP value depends on the number of relevant documents in the test set. In order to facilitate the comparison between the scores of a combination's topics, we have synthesised in most cases, topics with similar number of test documents. We have increased the number of evaluated documents from 1000 (according to TREC) to 3000 for two reasons. Firstly, to act as a remedy to the large number of test documents per topic [146]. Secondly, despite the equal number of training documents per topic, the best 1000 documents can be easily dominated by the topic with the largest number of test documents or with the strongest profile representation. Using 3000 documents, we avoided zero scores for the least dominant topics. The results were hence comparable, but remain unevenly distributed between topics. This is one of the reasons for concentrating on topics with a small number of documents for the last three combinations. The drawback of this remedy however, is that it does not

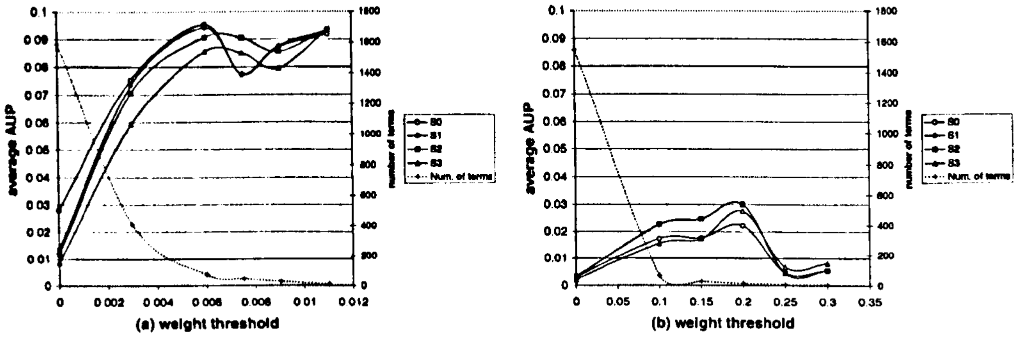

Figure 4-6: (a)IG and (b)RelDF for topics R1/R2(I)

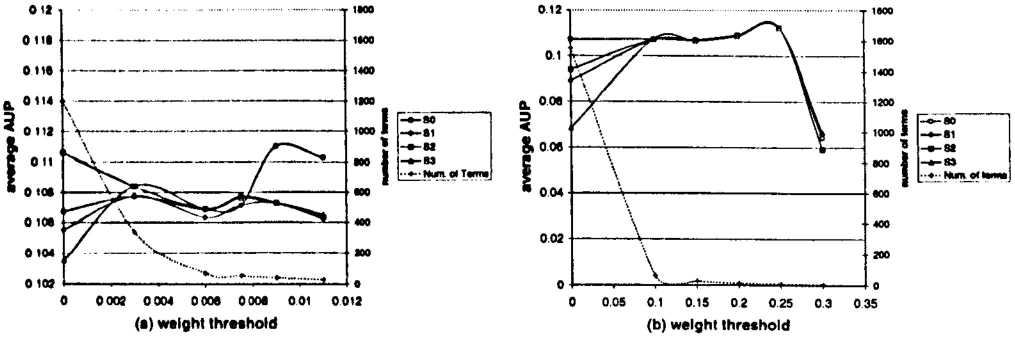

Figure 4-7: (a)IG and (b)RelDF for topics R7/R8(II)

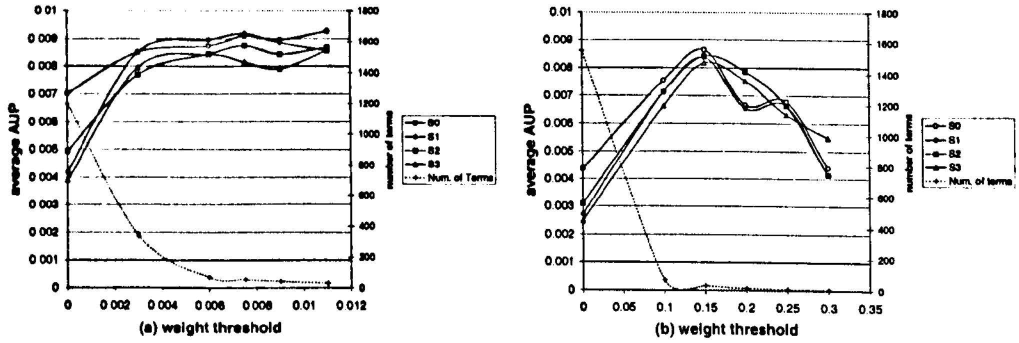

Figure 4-8: (a)IG and (b)RelDF and topics R29/R68(III)

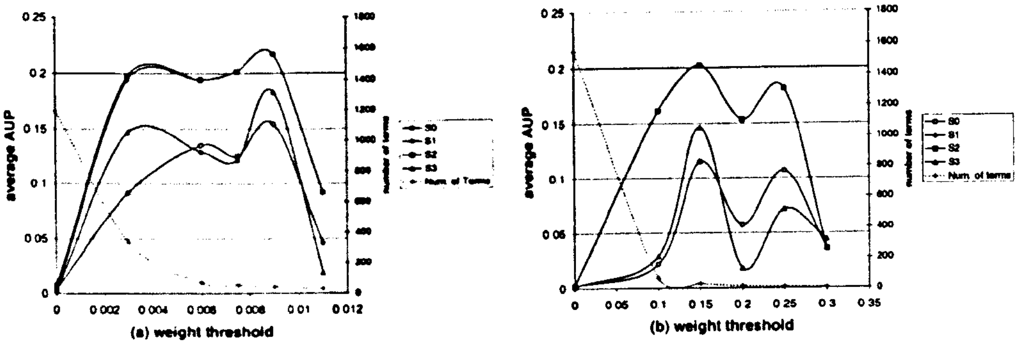

Figure 4-9: (a)IG and (b)RelDF for topics R10/R32(IV)

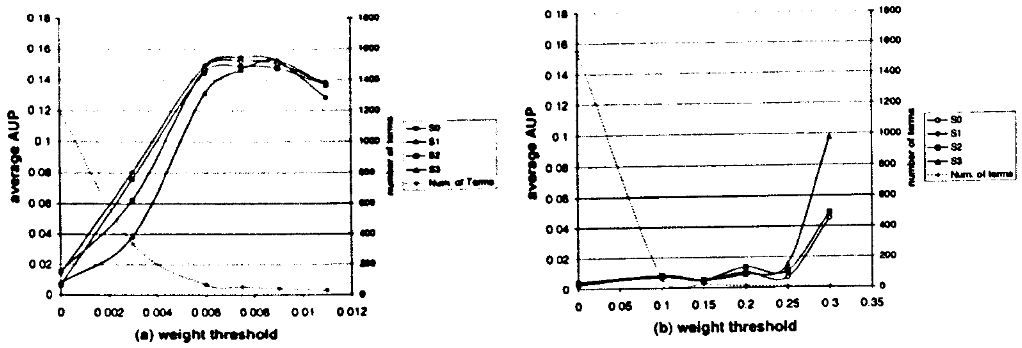

Figure 4-10: (a)IG and (b)RelDF and topics R6/R21(V)

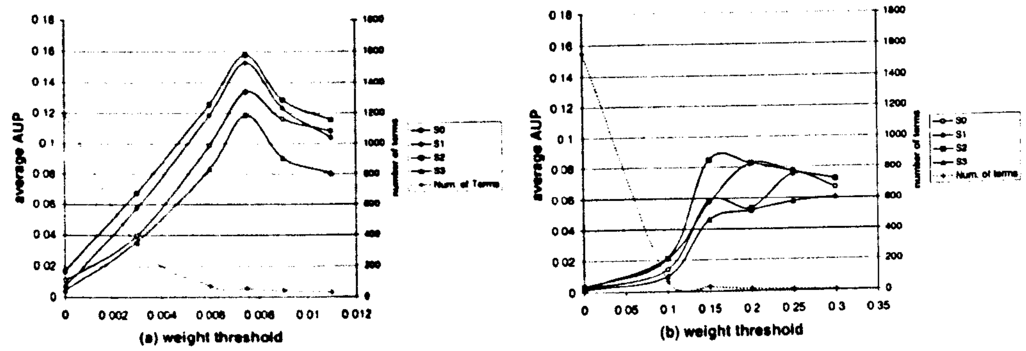

Figure 4-11: (a)IG and (b)RelDF for topics R41/R79(VI)

allow a direct comparison to the TREC results. The absolute value of AUP depends on the number of documents in the list, and hence, given the large number of test documents per topic, a list of 3000 documents can produce AUP values that are three times larger than those produced by the TREC participants. Instead, we compared the results with AUP scores for random generated lists of documents.

Figures 4-6 to 4-11 present for each topic combination and weight threshold, the average AUP score of the constituent topics. Each graph has been complemented with a dashed line that shows the number of profile terms for the different weight thresholds.

We first observe that for most combinations a profile's performance drops significantly for threshold 0, or, in other words, if all terms in the training documents are selected to populate the profile. This is particularly true for the last three combinations (figures 4-9 to 4-11). We may conclude, that for effective filtering some threshold larger than 0 is required for selecting the most informative terms in the training documents. For the same threshold, the results for the first three combinations (figure 4-6 to 4-8) indicate that S0 outperforms the rest of the methods in most cases. The hierarchical profiles (methods S1, S2 and especially S3) appear to suffer for very large number of profile terms and consequently an even larger number of links. For our further analysis we will exclude this extreme and disadvantageous threshold value and concentrate on thresholds larger than 0.

For such thresholds the profiles perform better, but, despite our intentions, there is a significant difference in the number of extracted terms between IG and RelDF. Figure 4-12, which presents for thresholds larger than 0 the average number of profile terms and links for IG and RelDF, summarises this difference. Clearly the threshold values that we have chosen result in

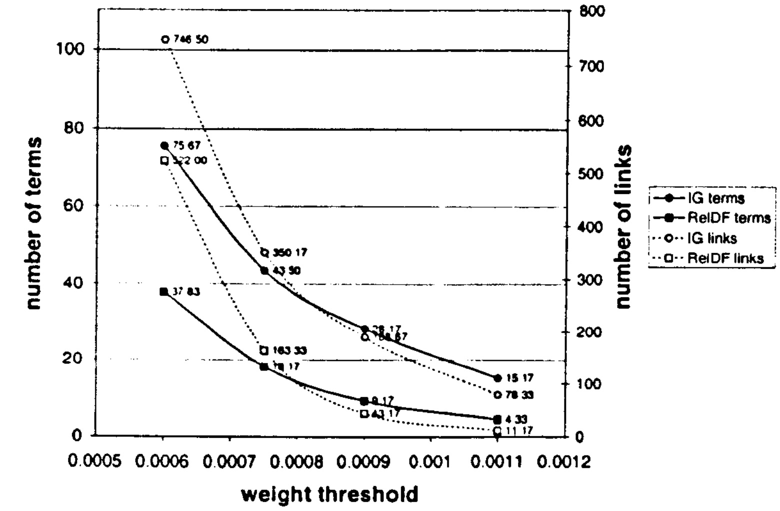

Figure 4-12: Average number of profile terms and links for different weight thresholds. (Notice that only the IG thresholds appear on the x-axis.)

different number of profile terms. Therefore, the current experimental setting does not allow direct comparison between IG and RelDF. The results are in general better for IG, but, given that more than one topic has to be represented, this may be due to the larger number of terms that IG thresholds extract. Nevertheless, this difference in the number of profile terms enables comparison between S0, S1, S2 and S3 on a larger spectrum of cases.

Figures 4-6 to 4-11 do not reveal a clear difference between hierarchical and unconnected profiles. Only S3 is in general inferior than the rest of the document evaluation approaches. Nevertheless, for combinations IV (fig. 4-9) and VI (fig. 4-11), the hierarchical profiles that use S1 and S2 outperform the unconnected profiles that use S0, for both IG and RelDF. The same observation can also be made for combination I, in the case of RelDF (fig. 4-6).

To further facilitate the comparison between hierarchical and unconnected profiles, tables 4.6 and 4.7 present for IG and RelDF respectively, the average AUP score over thresholds larger than 0, for individual topics and their combination. For each topic or combination the highest average AUP score is highlighted. The tables were complemented with an additional column that presents AUP scores for randomly generated lists of documents. More specifically, for each topic (or topic combination) we calculated the probability that a random test document will be relevant to that topic. For example, in the case of topic R1 this probability is equal to P(R1)=no. of relevant doc./no. of doc. = 23651/782927=0.03 (see Appendix A). We then used a random number generator to decide, based on this probability, if a document in a random list of 3000 will be relevant to a topic (or a combination) or not and we calculated the AUP of this list. For each topic (or topic combination) the random AUP score presented in the tables is the average AUP of 1000 such randomly generated lists. We used a "+" sign on the right of a random AUP score to indicate those topics or combinations for which most of the profiles perform better than random. We should note however, that we are not comparing the maximum achieved AUP score in each case, but rather the average over weight thresholds larger than 0.

It is confirmed that, given the specified thresholds, IG results in better filtering performance. For example, the overall score (average over all topics) for unconnected profiles (S0) and IG is 0.090387297 while it is only 0.042761276 for RelDF. Similar differences are observed for hierarchical profiles (S1, S2 and S3). As already mentioned, it appears that the RelDF thresholds do not extract a sufficient number of profile terms.

Another general observation concerns differences in the average AUP score between a combination's constituent topics. These differences are not fully justified by differences in the number of relevant test documents. Notice for example, that although topics R41 and R79 have a similar amount of test

Table 4.6: Two-Topic Average AUP for thresholds >0 and IG

	Document Evaluation Function				
Topic	S0	S1	S2	S3	random
R1	0.00005158	0.00004944	**0.0000563**	0.00005278	0.00629
R2	0.17058079	0.170104449	**0.172202195**	0.160972115	0.00342+
R1/R2(I)	0.085316185	0.085076945	**0.086129248**	0.080512448	0.00916+
R7	**0.000002372**	0.000001686	8.674E-08	0.000000367	0.00494
R8	**0.217200092**	0.213916881	0.214681689	0.214700334	0.0033+
R7/R8(II)	**0.108601232**	0.106959283	0.107340888	0.107350351	0.0077+
R29	0.0063713	**0.006785422**	0.00675889	0.005639931	0.007
R68	**0.011459611**	0.010829648	0.010024484	0.010746821	0.0045+
R29/R68(III)	**0.008915455**	0.008807535	0.008391687	0.008193376	0.011
R10	0.208589111	0.356701692	**0.358147506**	0.234131198	0.002+
R32	**0.011848945**	0.00259072	0.002670854	0.005437104	0.002+
R10/R32(IV)	0.110219028	0.179646206	**0.18040918**	0.119784151	0.0033+
R6	0.249275648	**0.259868847**	0.257292509	0.235145642	0.00095+
R21	**0.010658389**	0.006021198	0.00413597	0.007089893	0.001+
R6/R21(V)	0.129967018	**0.132945022**	0.130714239	0.121117768	0.0015+
R41	0.196054835	0.220589557	**0.236768546**	0.16196689	0.0009+
R79	**0.002554897**	0.001760789	0.001349763	0.001737926	0.001+
R41/R79(VI)	0.099304866	0.111175173	**0.119059154**	0.081852408	0.0014+
OVERALL	0.090387297	0.104101694	**0.105340733**	0.086468417	

Table 4.7: Two-Topic Average AUP for thresholds >0 and RelDF

	Document Evaluation Function				
Topic	S0	S1	S2	S3	random
R1	**0.001536066**	0.001342302	0.001343198	0.001200038	0.00629
R2	0.025514523	0.033677821	**0.033762491**	0.028993593	0.00342+
R1/R2(I)	0.013525294	0.017510061	**0.017552845**	0.015096816	0.00916+
R7	0.000374435	**0.000540158**	0.00053996	0.00033491	0.00494
R8	0.199763687	0.197293643	0.197377236	**0.200348375**	0.0033+
R7/R8(II)	0.100069061	0.0989169	0.098958598	**0.100341643**	0.0077+
R29	**0.004416934**	0.004127599	0.0041926	0.004389507	0.007
R68	0.009211312	0.009014338	**0.009493758**	0.009274633	0.0045+
R29/R68(III)	0.006814123	0.006570969	**0.006843179**	0.00683207	0.011
R10	0.107864732	0.278322949	**0.278440207**	0.099084399	0.002+
R32	**0.026698707**	0.015382827	0.015361084	0.024420375	0.002+
R10/R32(IV)	0.06728172	0.146852888	**0.146900646**	0.061752387	0.0033+
R6	0.025187493	0.030326426	0.030326489	**0.04729663**	0.00095+
R21	0.005017785	0.00477042	0.004775878	**0.006556899**	0.001+
R6/R21(V)	0.015102639	0.017548423	0.017551183	**0.026926765**	0.0015+
R41	0.107283345	0.124298406	**0.135498344**	0.090447313	0.0009+
R79	**0.0002663**	0.00012106	0.00008446	0.000188	0.001
R41/R79(VI)	0.053774822	0.062209733	**0.067791402**	0.045317656	0.0014+
OVERALL	0.042761276	0.058268162	**0.059266309**	0.042711223	

documents (table 4.5), there is a significant difference in their average AUP scores, both for IG (table 4.6) and RelDF (table 4.6). Given that the same number of training documents (30) is used for both topics, this difference is probably due to differences in the statistical characteristics of the training documents. This results in one of the topics being represented better by the profile and so the relevant documents dominate the best 3000 documents. In other words, the fact that evaluation is based on one fixed list of documents, combined with the large number of test documents per topic, exaggerates any differences in the way constituent topics are represented.

This effect is further illustrated by comparing the results to the random AUP scores. We note that for IG (table 4.6) and combinations I, II and III, the profiles perform better than random for only one of the constituent topics in each combination. Although the results for topics R68, and especially, R2 and R8 are much better than random, the results for topics R1, R7 and R29 are worse. The first three combinations comprise topics with a large number of test documents (table 4.5) and hence, the result list is easily dominated by documents about the better represented topic. For the last three combinations (IV, V, and VI) however, that comprise topics with a smaller number of test documents, the results are better than random for both constituent topics in each combination. With the exception of combination III, the results for topic combinations are better than random. The bad results for combination III are probably due to the large difference between the total of 1049 (630 + 419) training documents that are available for topics R29 and R68 and the 60 (30+30) training documents that we used. The above findings are confirmed by the results for RelDF (table 4.7), except for topic R79 that, in contrast to IG, produces worse than random results.

Turning to the comparison between unconnected and hierarchical profiles, we count, in the case of IG (table 4.6), 6 topics and 2 combinations for which the unconnected profile (S0) exhibits the highest average AUP score. The hierarchical profile using S1 is the best approach for 2 topics and 1 combination. With S2, the hierarchical profile outperforms the rest of the approaches for 4 topics and 3 combinations, including combinations IV and VI, for which the largest differences to the unconnected profile are observed. S3 is the worst performing approach. S2 has the largest overall score, with S1, S0 and S3 following in order.

In the case of RelDF (table 4.7), the unconnected profile is on average the best for 4 topics. The hierarchical profile exhibits the best average AUP score, for 1 topic with S1, for 4 topics with S2, and for 3 topics with S3. The last two approaches are also the best for 4 (including IV and VI) and 2 combinations respectively. In terms of overall score, S2 is the best approach, followed in order by S1, S0 and S3.

These results are encouraging. Although, in the case of IG (table 4.6), the unconnected profile (S0) is the best for most topics, the hierarchical profile with S2 is the best for 3 out of the six combinations. For combination IV and VI in particular S1 is also on average better than S0. In the case of RelDF (table 4.7), more significant relative differences are indicated. S2 is on average the best approach for 4 combinations, and for most of them S1 is also better than S0. Despite having the worst overall score, S3 is the best for 2 combinations. Furthermore, there is now a more significant relative difference between the overall scores for S2 and S1 and that for S0. We believe, that although with RelDF an insufficient number of terms is extracted, a positive side-effect is the smaller number of links in relation to IG (fig. 4-12). We have already found that the hierarchical profiles suffer for very large number of links. So in the case of IG, we may suggest that

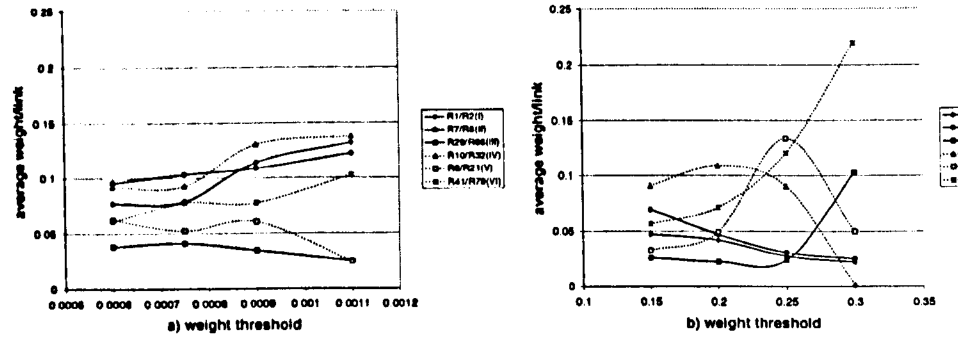

Figure 4-13: Average link weight for different topic combinations and weight thresholds: (a)IG and (b)RelDF.

the performance of hierarchical profiles is affected more by this drawback. The results for the three-topic experiments support this argument (see next section).

We also noted, that the hierarchical profiles perform particularly well for combinations IV and VI. What distinguishes these combinations from the rest? Since for each combination and term weighting method, both the unconnected and the hierarchical profiles are built with the same set of weighted terms, it is natural to turn again to links for the answer. Figures 4-13(a) and 4-13(b) present for IG and RelDF respectively, the average link weight for the different topic combinations and for weight thresholds larger than 0. It appears, that especially in the case of RelDF there is a correlation between the average link weight and how well the hierarchical profile performs in relation to the unconnected. The hierarchical profile performs better if it includes links with large weights, or in other words links that we are confident about. In addition to combinations IV and VI, this is also true, in the case of RelDF, for combination V (fig. 4-13b). We should also remember that these last three combinations comprise unrelated topics (table 4.5). (R6(C16) and R21(C32) are only related at a higher level). For such combinations the terms and their corresponding links are extracted from different contexts.

Table 4.8: Three-topic combinations: codes and collection statistics

Comb.	Topics	Code	Training	Test
I'	R1	C11	597	23651
	R2	C12	351	11563
	R3	C13	821	36463
II'	R7	C171	403	17876
	R8	C172	251	11202
	R9	C173	68	2560
III'	R1	C11	597	23651
	R29	E12	630	26402
	R68	GJOB	419	16770
IV'	R10	C174	212	5625
	R32	E131	140	5492
	R50	E71	149	5104
V'	R6	C16	42	1871
	R20	C313	38	1074
	R21	C32	39	2041
VI'	R41	E311	35	1658
	R58	G157	41	1991
	R79	GWELF	42	1818

4.2.3 Three-Topic Experiments

In addition to the above two-topic experiments, we have experimented with combinations of three topics. For these three-topic experiments the two-topic combinations of table 4.5 have been complemented with an additional topic per combination (table 4.8). We used the same methodology, which produced the following results.

Figures 4-14 to 4-19 present for each three-topic combination the average AUP score for the different weight thresholds. The graphs were again complemented with a dashed link showing the number of profile terms per threshold value.

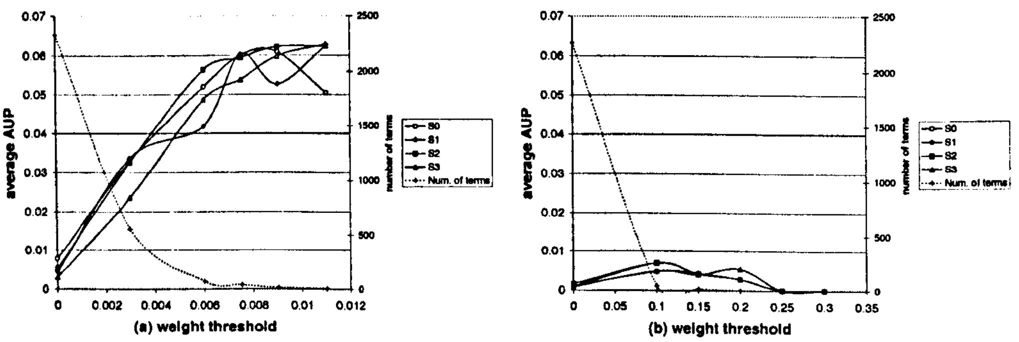

Figure 4-14: (a)IG and (b)RelDF for topics R1/R2/R3(I')

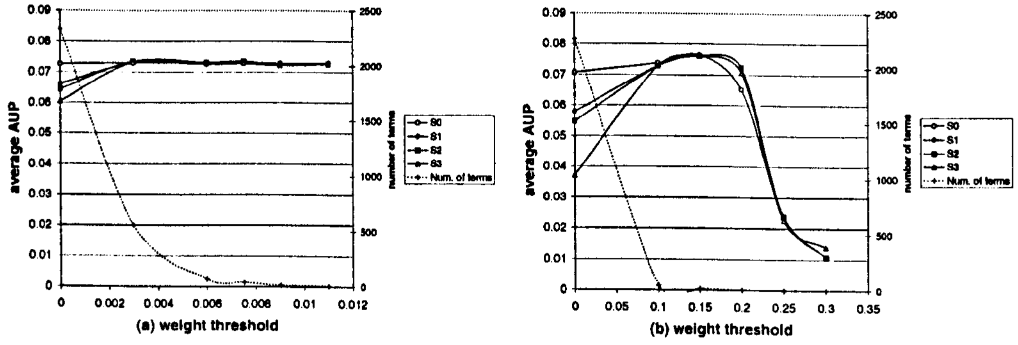

Figure 4-15: (a)IG and (b)RelDF for topics R7/R8/R9(II')

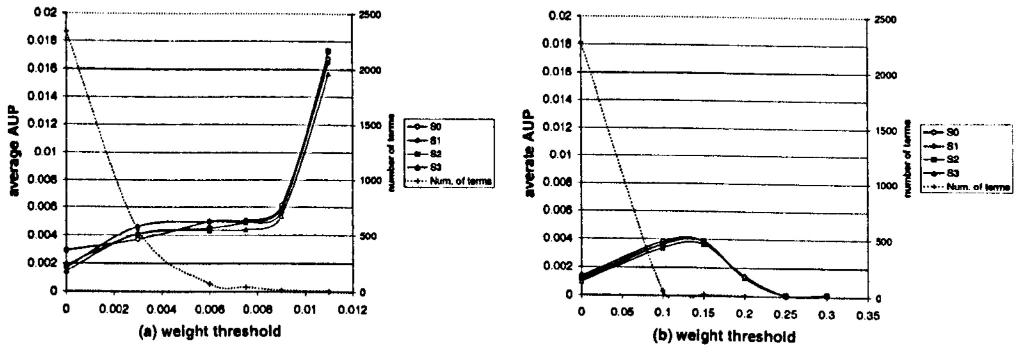

Figure 4-16: (a)IG and (b)RelDF and topics R1/R29/R68(III')

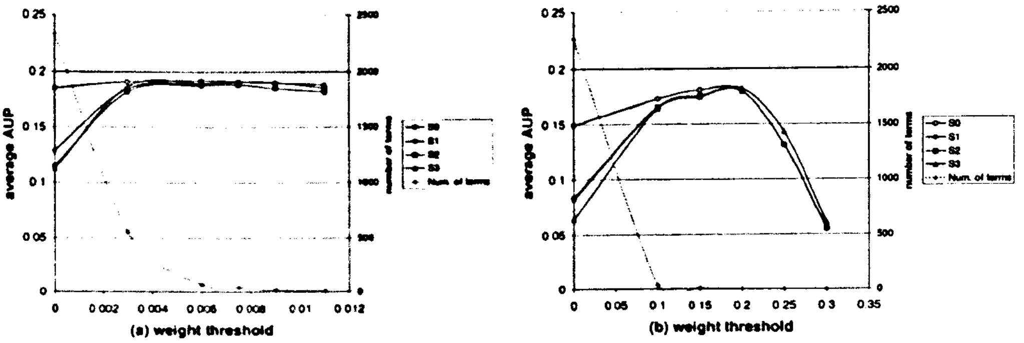

Figure 4-17: (a)IG and (b)RelDF for topics R10/R32/R50(IV')

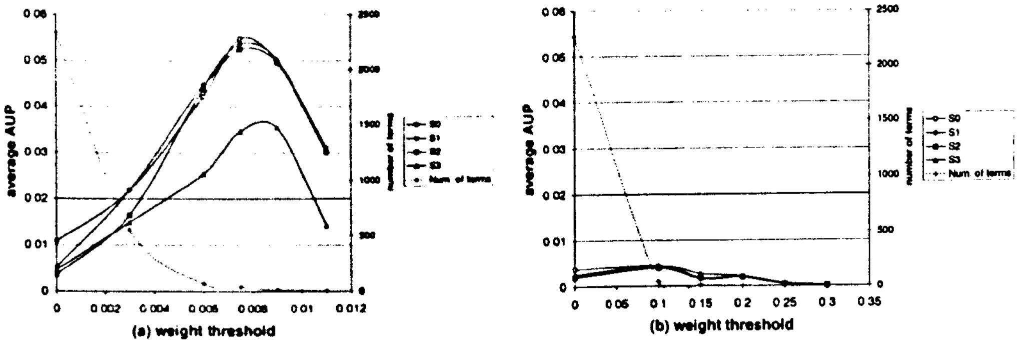

Figure 4-18: (a)IG and (b)RelDF and topics R6/R20/R21(V')

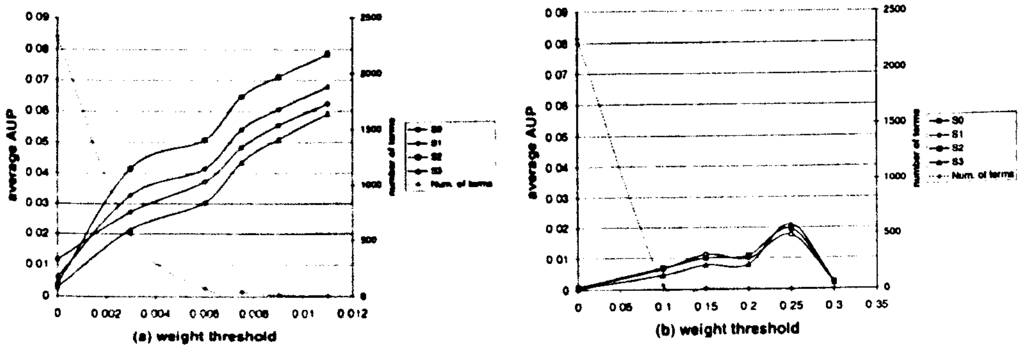

Figure 4-19: (a)IG and (b)RelDF for topics R41/R58/R79(VI')

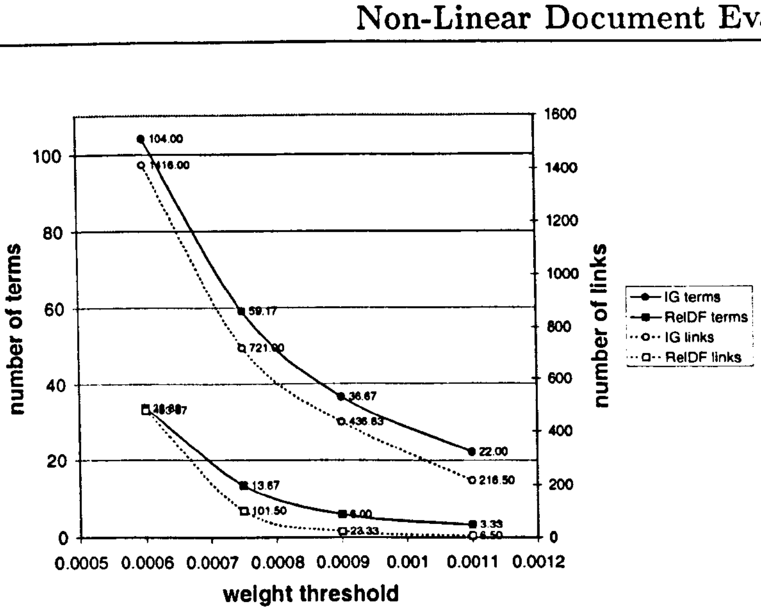

Figure 4-20: Average number of terms and links for different weight thresholds. Notice that only the IG thresholds appear at the x-axis.

Once more these figures do not reveal any significant difference between unconnected and hierarchical profiles. With the exception of combination VI' (fig. 4-19), for which S1 and S2 are clearly better than S0, all four methods appear to have comparable performance. Nevertheless, the results confirm that a profile's performance is inferior if all terms in the training documents are extracted (threshold = 0). Hierarchical profiles, and especially S3, are more sensitive to this extreme case, apparently due to the very large number of links. We again exclude thresholds equal to 0 from the rest of the analysis.

In relation to the two-topic experiments, the results for IG are now even better than those for RelDF. Their difference can be again attributed to differences in the number of extracted for IG and RelDF thresholds. Figure 4-20 presents for thresholds larger than 0 the average number of terms and links in the case of IG and RelDF. A comparison with the corresponding figure in the two-topic experiments (fig. 4-12) indicates that the number of terms extracted for IG has now increased. Justifiably, more terms are extracted from the 90 training documents per combination to represent the constituent three topics. The same however is not true for RelDF. The average number of terms drops in comparison to the two-topic experiments. The absolute weight values that RelDF assigns depend on the number of training documents. The number of terms that are extracted based on the specified, fixed thresholds varies accordingly. No direct comparison between IG and RelDF can be made, but this not the main objective of these experiments.

To allow the comparison between unconnected and hierarchical profiles we have produced tables 4.9 and 4.10, which present for IG and RelDF respectively, the average AUP score over thresholds larger than 0, of the different topics and their combinations. The tables were once more complemented with an additional column that presents AUP scores for randomly generated lists of documents. We used a "+" sign on the right of a random AUP score to indicate those topics or combinations for which most of the profiles perform better than random.

As expected, the performance for IG is on average almost double that for RelDF. There are also differences in the average AUP of each combination's constituent topics. As in the case of the two-topic experiments, they are not fully justified by the differences in numbers of test documents and so indicate differences in the way the constituent topics are represented. They are also further exaggerated by the fact that although the evaluation is again based on the best 3000 documents, three topics, and hence a larger number of relevant documents in the test set, correspond to each profile. For the same reason, the overall performance for the three-topic combinations is clearly worse than the performance for the corresponding two topic combinations (tables 4.6 and 4.7).

The comparison of the results to the random AUP values confirm the com-

Table 4.9: Three-Topic Average AUP for thresholds >0 and IG

	Document Evaluation Function				
Topic	S0	S1	S2	S3	random
R1	0.00007558	0.0000783	**0.0000866**	0.00007734	0.00629
R2	0.152260836	0.149156731	**0.162477153**	0.147948848	0.00342+
R3	**0.001478699**	0.00138353	0.001291171	0.001318886	0.0095
R1/R2/R3(I')	0.051271705	0.050206187	**0.054618308**	0.049781691	0.018+
R7	0.000001942	0.000001976	**0.000004476**	0.000002058	0.00494
R8	0.218808172	0.217852622	**0.218879488**	0.218507595	0.0033+
R9	2.62E-08	0	**0.0000256**	0	0.0011
R7/R8/R9(II')	0.072936713	0.072618199	**0.072969855**	0.072836551	0.0083+
R1	6.7812E-06	0.000005563	**1.00216E-05**	5.4968E-06	0.00629
R29	0.007712369	**0.008334294**	0.006489833	0.007381439	0.007+
R68	0.014275402	0.013927854	**0.015443835**	0.013034531	0.0045+
R1/R29/R68(III')	0.007331517	**0.00742257**	0.007314563	0.006807156	0.0167
R10	0	0	0	0	0.002
R32	0.000040958	0.00019064	**0.00021352**	0.00001805	0.002
R50	**0.570440521**	0.556657454	0.55451417	0.567303262	0.0019+
R10/R32/R50(IV')	**0.190160493**	0.185616031	0.18490923	0.189107104	0.0045+
R6	0.099930048	**0.101783813**	0.099586776	0.051469845	0.00095+
R20	0.000614135	0.000580456	0.000581029	**0.00078658**	0.00078
R21	0.019362335	0.018102938	0.015213222	**0.022502006**	0.001+
R6/R21/R20(V')	0.039968839	**0.040155735**	0.038460342	0.024919477	0.0018+
R41	4.15644E-05	5.51152E-05	**0.0001666**	0.000136302	0.0009
R79	0.137421257	0.153280099	**0.182968599**	0.122096473	0.001+
R58	0.000755906	0.000678552	0.000786183	**0.000990052**	0.00095
R41/R79/R58(VI')	0.046072909	0.051337922	**0.061307127**	0.041074276	0.0018+
OVERALL	0.06795703	0.067892774	**0.069929904**	0.064087709	

Table 4.10: Three-Topic Average AUP for thresholds >0 and Reldf

	Document Evaluation Function				
Topic	S0	S1	S2	S3	random
R1	**0.001884627**	0.00176264	0.001764924	0.001450246	0.00629
R2	0.004777779	0.006130061	0.006120732	**0.006634937**	0.00342+
R3	0.000566602	0.00057808	0.000577275	**0.000688885**	0.0095
R1/R2/R3(I')	0.002409669	0.002823594	0.002820977	**0.002924689**	0.0018+
R7	0.005601685	**0.005640097**	**0.005640097**	0.003839054	0.00494+
R8	0.139351514	0.142806808	**0.142903177**	0.142070424	0.0033+
R9	0.005181316	0.005029459	0.005029459	**0.007766454**	0.0011+
R7/R8/R9(II')	0.050044839	0.051158788	0.051190911	**0.051225311**	0.0083+
R1	0.00022401	**0.00022404**	**0.00022404**	0.00019274	0.00629
R29	0.003472768	**0.003478907**	0.003477058	0.003399739	0.007
R68	**0.001818033**	0.001700437	0.001700448	0.001621021	0.0045
R1/R29/R68(III')	**0.00183827**	0.001801128	0.001800515	0.001737834	0.0167
R10	**6.27E-08**	3.794E-08	3.794E-08	1.42E-08	0.002
R32	0.0151914	**0.0152446**	**0.0152446**	0.008848581	0.002+
R50	0.416873914	0.409518829	0.409518829	**0.426119363**	0.0019+
R10/R32/R50(IV')	0.144021792	0.141587822	0.141587822	**0.144989319**	0.0045+
R6	0.000070722	0.000060414	0.000072434	**0.000101564**	0.00095
R20	0.002364256	**0.00236516**	0.00232631	0.002287842	0.00078+
R21	0.002784053	0.002365698	0.002501553	**0.003641415**	0.001+
R6/R21/R20(V')	0.001739677	0.00159709	0.001633432	**0.002010274**	0.0018
R41	0.003638569	0.004571317	**0.004574857**	0.002430641	0.0009+
R79	0.024407378	0.024706248	**0.024714705**	0.023148172	0.001+
R58	**0.000378058**	0.000311068	0.000311068	0.00033447	0.00095
R41/R79/R58(VI')	0.009474668	0.009862878	**0.009866877**	0.008637761	0.0018+
OVERALL	0.034921486	0.034805217	0.034816756	**0.035254198**	

bined effect of a fixed list of evaluation documents with the large number of test documents that correspond to each topic. In the case of IG (table 4.9), and most topic combinations, the profiles perform better than random for only one of the constituent topics. Only for combinations III' and IV', do two of the constituent topics produce better than random results. For the same possible reasons as before (see previous section), combination III' is the only one with worse than random results. The results for RelDF (table 4.10) however, are more evenly distributed between constituent topics. For the last 3 combinations that comprise topics with a small number of test documents, two of the constituent topics in each case produce better than random results. For combination II', the profiles perform better than random for all three constituent topics. For combination I', the results for only one of the topics are better than random and for combination III' for none. The latter, and combination V' marginally, are the only combinations for which profiles perform worse than random. The difference in the way performance is distributed between topics in the case of IG and RelDF is possibly justified by the fact that we apply the binary version of IG to a problem that in a sense m-ary. The m-ary version of IG however requires preclassification of documents that we try to avoid.

Regarding the comparison between unconnected and hierarchical profiles, the results for IG (table 4.9) indicate that S0 is only the best for 2 topics and 1 combination. The hierarchical profiles on the other hand, exhibit the best average AUP score for, 2 topics and 1 combination with S1, 10 topics and 3 combination with S2 and 3 topics for S3. S2 has best overall score, with S0, S1 and S3 following in order.

In the case of RelDF (table 4.10), the results show that S0 is the best approach for 4 topics and 1 combination, S1 for 4 topics, S2 for 6 topics and

1 combination and S3 for 6 topics and 4 combinations. S3 has the highest overall score, with S0, S2 and S1 following in order.

So on average, the results for the three-topic experiments are also encouraging. The hierarchical profiles outperform the unconnected from most topics and combinations. However, the four approaches produce only marginally different results. The only exception is for IG and combination VI', which includes 3 unrelated subtopics with a small number of relevant documents per topic (table 4.8). This drop in relative performance can be justified in two ways. Firstly, it is now more likely that the best 3000 documents are dominated by documents relevant to the better represented topic in each profile. More relevant documents correspond to each combination, but the number of evaluation documents has remained the same. Secondly, in the case of IG (table 4.9), the increased number of extracted terms results in an even larger number of links (fig. 4-20), which affect the performance of hierarchical profiles. For RelDF on the other hand, the number of extracted terms is too small to generate significant differences. The corresponding small number of links favours S3. Note that S3 is the approach that takes into account both the breadth and size of activated profiles (section 4.2), and also the approach, which is affected more by large numbers of extracted terms and hence of links (threshold = 0).

4.2.4 Discussion

To evaluate the proposed continuous approach to document evaluation we have performed a series of experiments. Single-topic experiments involved the construction of a separate profile for each of the first 10 RCV1 topics (section 4.2.1). The results for these experiments have shown that the continuous approach is superior to the layered approach and that it compares positively to the results of the average TREC 2001 routing participant. However, the performance of the hierarchical profile, in comparison to the unconnected profile version was not satisfactory. We have attributed this outcome to the large number of test documents that are relevant to each topic. This acknowledged drawback of RCV1, leads in most cases to increased performance for even a very small number of general, exhaustive profile terms. In this context, the ability of the hierarchical network to represent the dependencies between terms and therefore more than one topic of interest, does not render it competitive. On the contrary, the high connectivity of a profile with a large number of terms may have a negative effect.

Nevertheless, our goal has been to effectively evaluate documents with a single, multi-topic profile. The lack of multi-topic profile representations has been coupled with a lack of appropriate evaluation methodologies and therefore, to evaluate our approach we have attempted to establish a new methodology. It involved training profiles for combinations of two and three topics. Various combinations of topics with different characteristics have been synthesised for this purpose. Further alterations to the strict TREC routing guidelines have also been made to enable such experimentation. These included the appropriate calculation of a single AUP score for each combination and the increase of the number of evaluation documents from 1000 to 3000. The experiments involved both unconnected profiles using the document evaluation function S0 and hierarchical profiles which employed the three proposed functions S1, S2 and S3. For each topic combination the four kinds of profile were constructed using the same set of terms, which have been selected on the basis of either IG or RelDF weights using empirically selected thresholds.

Unfortunately, the experimental setting did not allow further comparison between IG and RelDF. Due to the fixed RelDF thresholds, the number of extracted terms is smaller in the case of RelDF, resulting in inferior performance for most topic combinations. The comparison could be facilitated if we had used relative threshold values, like for example fixed percentages of the maximum assigned weight for each combination.

Nevertheless, the comparison between unconnected and hierarchical profiles has been promising. Hierarchical profiles, especially using S2, perform on average better for most topics and their combinations. For the two-topic experiments in particular, it was observed that a hierarchical profile performs particularly well for combinations that produce profiles with large average link weight. These are combinations that comprise semantically unrelated topics. The difference in average performance between hierarchical and unconnected profiles was smaller for the three topic experiments. This relative drop in difference is justified to some extent by the fixed number (3000) of evaluation documents and in the case of IG by the increase in the number of links in relation to the two-topic experiments.

In general, the experiments indicate that the performance of our approach (especially using S3) is affected negatively if too many links are generated. This is exaggerated for threshold 0 which extracts all terms in the training documents and therefore results in a very large number of links. It is our priority for the future to control, as suggested in section 3.6, the number of generated links using appropriate thresholds on link weights. According to the results for the two-topic combinations IV and V, we expect that further improvements in performance can be achieved if the most significant links are maintained.

Another conclusion that is made evident by the experiments is that multi-topic information filtering cannot reside only on a single relevance score.

Although exaggerated due to the large number of test documents per topic, it is obvious that differences in the way topics are represented within the profile may result in ordered lists that are dominated by documents about the best represented topic. This implies that ways of presenting the filtered documents in such a way that their topic(s) is indicated are required. We suggest some possible solutions in the next section.

In summary, the experimental results have been promising and motivating, despite the fact that the hierarchical profile has not been optimised. Parameters that may be optimised include the window size, a threshold for maintaining the most important associations between terms and a threshold for extracting the most important terms from the user specified documents. The number of parameters that require optimisation is nevertheless smaller than for systems that use a separate profile for each topic of interest [195, 125, 108]. Furthermore, we should not forget that the experiments conducted so far test the performance of the initial profile which has been constructed using a small number of initialisation documents. Profile adaptation based on additional feedback documents may result in further performance improvements, especially by increasing the quality of generated links.

4.3 Additional Evidence

We have already argued that the quantitative relevance score, which a document evaluation function assigns to each document, is not sufficient for multi-topic information filtering. The user should be able to distinguish between documents of different topics, which implies that additional evidence of a document's aboutness should be provided.

In section 3.6 we described how the general topics that the profile represents can be identified by dominant terms that are only linked to terms further down in the hierarchy. The same terms concentrate a large number of links comparative to the rest of the terms in the hierarchy. Moving down the hierarchy, the same evidence can be used to identify terms that define subtopics of interest which are related to the major topics. The level of interest in each of the represented topics or subtopics is reflected by the size of the corresponding hierarchical network. The user can therefore be provided with an overview of her dynamically represented interests, which may include aspects she was not previously aware of.

During filtering, the user profile may be used as a concept hierarchy to structure the filtering results according to the main topics and subtopics of interest. The application of concept hierarchies for interactive access to information has already been suggested [5, 129, 163]. Typically, the user is provided with, either dynamically generated windows, or menus that can be used to focus from topics to related subtopics. Although, such interfaces can be supported by the hierarchical profile, their disadvantage is that they do not order documents according to their relevance to the user interests. One can envisage an integration of dynamically generated windows, or menus, with ordered lists of documents. More specifically, for each major topic of interest an ordered list of all related documents can be presented to the user. The user may then focus on a related subtopic, successively moving to more specific ordered lists. Such an interface overcomes the problems associated with a single ordered list of documents due to the uneven distribution of relevance scores.

To assign each filtered document to the topics and subtopics represented by the profile, evidence derived from mining the topology of the activated profile's subnetwork can be exploited. In section 4.2, we described how in a way similar to the complete profile, one may identify the dominant activated terms and the size of the corresponding subhierarchies. A document's dominant term(s) can then be used to assign the document to the ordered list(s) under the corresponding topic or subtopic of interest in the above dynamically generated menus. Within each list, the document's rank will depend on its relative score, which as already mentioned, is analogous to the size of the corresponding activated subhierarchy.

The above additional evidence can also be used to provide a hierarchical summary of the document's content. Hierarchical multi-document summaries based on concept hierarchies have been suggested by [94]. In a similar way a document's score can be complemented with the hierarchy of topics and subtopics that are discussed in a document.

Finally, for domains with long documents or books, we have already mentioned that all of the above processes can in principle be applied at the paragraph or sentence level. In this case a document's relevance can be described by an histogram that depicts its distribution through the document. Summaries of individual paragraphs can also be provided. The user is therefore pointed towards specific parts of the document that are more likely to be of interest.

In summary, the computational advantage of the proposed hierarchical profile in estimating a document's relevance score, may be coupled with the ability to mine its topology to derive evidence about a document's aboutness. Such additional evidence supports document summarisation and interactive access to the filtering results. We intend to tackle these interface issues in our future research.

4.4 Personalisation Services

So far in this chapter we have concentrated on how the user profile can be applied for non-linear document evaluation, but we did not specify how the documents to be filtered are obtained. We avoided thresholding for reasons explained in section 2.11 and to isolate its additional influence on the filtering performance during evaluation. In other words we have concentrated on the content-based filtering of a batch of documents, but we did not exclude dynamic information sources. Instead, we approach IF within a broader Personalised Information Delivery context.

We envisage a scenario where each user in a community (organisational, academic, etc.) has a separate hierarchical profile. Each user has access to external sources of information, either static or dynamic, and a central repository or index of shared documents. Given this setting, a number of personalisation services can be supported.

Each time a user wishes to make a document public, the user submits it to the central repository or index. The submitted document may also be annotated as judged relevant by that user. It is then evaluated by the profiles of the rest of the users in the community and those interested enough are notified. This of course implies the ability to make the binary decision of notifying the user or not. However, as we have argued in section 2.11, such binary decisions cannot be made based only on the document's relevance score. Documents relevant to different topics may be assigned scores of different scale according to the relative importance of these topics within a user's profile. The decision making process should also take into account the additional evidence that we have described in the previous section. Another source of information can be the relevance score of past documents that received positive user feedback. The same kind of thresholding can also be applied for the filtering of documents from external dynamic information sources. Although we intend to explore these new directions, we still believe that in many real situations the user is only periodically engaged in information seeking episodes. In the mean time, the received documents can be collected into a batch and therefore we can revert to the presentation techniques described in the previous section, without the need for thresholding.

In addition to the passive reception and filtering of information, retrieval of documents from static information sources can also be facilitated or automated based on a user's profile. A user may actively search for information by either submitting a conventional query or by specifying a document of interest. In the first case, the query can be expanded with profile terms that relate to the query terms. The expansion may exploit the topic-subtopic relations between terms. The query can be generalised by moving up the hierarchy, or specialised by moving downwards. In the second case, the user wishes to find documents relevant to the specified document. For such document-based retrieval, a query can be automatically formulated with those profile terms that appear in the document and their related terms. It can also be autonomously initiated by the system based on the document that the user is currently reading or editing. The expanded or automatically formulated query can then be submitted to the central repository or to external search engines as in the case of [127]. The retrieval results are then filtered by the complete profile and are presented to the user as we described. For the filtering process, the query, or the document, may be used to temporarily activate profile terms and therefore move the profile towards the query's or document's topic, as in [72].

Another source of interesting information can be other users in the community. A user might initialise an expert finding process by submitting a query or a complete document. The profile expanded query or the complete document are then evaluated against the profiles of the rest of the users and the most relevant users are returned. For this purpose an expertise profile may also be generated from documents that the user has produced. In a business environment these may be progress reports, whereas in an academic environment they may be a researcher's papers. Collaboration between the users in the community may also be boosted by identifying those that share similar interests. In this case each user's complete profile is evaluated by the profile's of the rest of the users. The result is a *shared interest matrix* which measures for each pair of users the similarity in their interests. This matrix can be used to complement content-based filtering, performed by the profile itself, with collaborative filtering. For documents that at least one user in the community has annotated, the shared interest matrix can be used to calculate a recommendation score which may be used complementary to the content-based relevance score. Such a hybrid approach overcomes the sparsness problem in collaborative filtering [143].

Arguably, the proposed hierarchical profile is not constrained to the content-based filtering of documents. Additional personalisation services may be provided which extend its scope. These services can be important in the domain of decentralised Knowledge Management (KM) [23]. In fact, similar profile-based services have already been employed in a real situation by the Knowledge Sharing Environment (KSE) system [42]. Another example of an approach that employs IF for KM is the Knowledge Pump system [58]. We believe, that our innovative approach to profile representation and document evaluation can enable further steps forward along this direction.

Finally, it is important to note at this point that neither the profile representation nor the document evaluation process rest on syntactical evidence.

Therefore. in principle, the approach is also applicable to other media, like audio and image, for which descriptive features can be statistically extracted. We are particularly interested in personalised music delivery.

4.5 Summary and Conclusions

In the previous chapter, we presented a methodology for generating a hierarchical user profile out of a set of user specified documents. The hierarchical profile represents the statistical dependencies caused by both topical and lexical correlations and distinguishes topic-subtopic relations between terms. The net effect is that the profile can represent more than one topic of interest. Our goal in this chapter was mainly to establish a non-linear document evaluation function that allows a single, multi-topic profile to be used computationally for document evaluation.

In particular, we have experimented with two alternative approaches. Initially, we have partitioned profile terms into three hierarchical layers. A directed spreading activation model was then introduced which supports an evaluation function that takes into account the represented term dependencies. The layering however causes a discretisation which leads to uneven distribution of term importance. Consequently, experiments on the first 10 RCV1 topics, using a variation of the TREC routing methodology, did not produce satisfactory results.

These results prompted the exploration of an alternative continuous approach with a different spreading activation model, which we argued may support a whole new domain of document evaluation functions. We have introduced three such functions, which exploit the dependencies and topic-subtopic relations that the hierarchical profile represents. The last two of the evaluation functions in particular, take into account additional evidence that can be extracted by mining the topology of the hierarchical profile subnetwork that a document activates.

The proposed continuous approach was tested against an unconnected profile version, which evaluated documents using the inner product, in a series of experiments. Initial, single-topic experiments on the first 10 topics revealed that the continuous approach is superior to the layered. The comparison however between the hierarchical profile using the first of the introduced functions and the unconnected profile produced only partially positive results. We have argued that the large number of relevant documents per topic in the test set influences the experimental results by favouring generality. For most topics a very small number of general, exhaustive terms may produce a large score.

To create a more challenging experimental setting and to support our argument, that in contrast to the state-of-the-art the proposed hierarchical profile can represent multiple topics of interest, we have attempted to establish a methodology for evaluating single, multi-topic profiles. The methodology was based on yet another variation of the TREC routing guidelines and was used to conduct both two-topic and three-topic experiments. We experimented with both unconnected profiles that use the linear inner product and hierarchical profiles that use the three non-linear functions introduced. Despite the fact that the hierarchical profile has not yet been optimised, and the small number of initialisation documents, the results indicated that the hierarchical profile performs on average better than the unconnected. However, significant differences in performance have only been observed in situations where strong links are generated. In general, our approach appears to suffer from large number of links. Both findings point towards the use of thresholds for controlling the quality and quantity of generated links. The encouraging results are further supported by experiments conducted in the next chapter.

The results of the multi-topic experiments have also supported our argument that for multi-topic filtering the quantitative relevance score should be complemented with additional evidence. We described how such evidence can be derived by mining the topology of the complete or the activated profile. This evidence may support improved interactive presentation of the filtering results and document summarisation. Therefore, complementary to the relevance score, additional evidence of a document's aboutness can be provided. In the case of long documents, the same processes may also be applied at the paragraph or sentence level.

In addition to the above document evaluation functionalities, we suggested ways of using the profile to provide other personalisation services including, automated retrieval, expert finding and collaborative filtering. The scope of IF is therefore broadened and may include application domains like KM. We have also argued that in principle our approach can also be applied to any media for which features can be statistically extracted.

In conclusion, in the last two chapters we have proposed a generalised solution to the focusing processes that comprise PID. It was made evident that the hierarchical profile may support automated document retrieval, filtering of obtained documents, enhanced presentation schemas and other personalisation services. Starting with the accessible information space we have therefore reached the presented information space. We are now left with the fascinating challenge of adapting the user profile based on documents that either explicitly or implicitly received user feedback.

Chapter 5

Profile Adaptation through Self-Organisation

So far, we have tackled the first of the two issues discussed in section 2.10: multi-topic information filtering with a single user profile. Here we turn to the second issue; profile adaptation to changes in user interests. We concentrate in other words, on *Adaptive Document Filtering*. As we have already argued in section 2.13, user interests are by nature dynamic. A combination of parameters causes a variety of changes. Frequent changes in the user's short-term needs contribute to progressive changes in the user's long term interests and vice versa. The user's interests may shift frequently between different topics or related subtopics. Occasionally, new topics and subtopics of interest emerge and the interest in a certain topic might be lost. A subtopic may attract increased interest to become a general topic of interest. For example, a general interest in *Knowledge Management* can trigger an interest in *Intelligent Information Agents*, which may evolve to include related aspects like *Information Retrieval* and *Information Filtering*. The latter may themselves develop, causing a decay in the initial interest in *Knowledge Man-*

agement and the emergence of other topics like *Term Weighting*, *Complex Adaptive Systems* and so on.

Despite the complex, dynamic nature of user interests, there is an evident tendency in the literature to couple single-topic profile representations with linear learning algorithms like Rocchio's, which assume a steady change pace, reflected by a constant learning coefficient [90, 165, 26, 66]. Alternatively, reinforcement learning algorithms that adjust the learning coefficient over time have been employed [168, 109, 24, 7]. Nevertheless, the learning coefficient in this case is adjusted in a way external to the change itself and the goal is usually to optimise a profile to a specific topic. The same goal is shared by machine learning algorithms inherited from TC research, and hence, their appropriateness for adapting the profile to the above drifts in user interests has been contested [191]. To account for the difference in the pace of changes in long-term interests and short-term needs, dual profiles have been suggested [195, 33, 19, 50]. They employ a separate profile representation level for each of these two kinds of changes, with each being adapted at a different, usually constant, pace. Different weights are also assigned to each single-topic profile and are adjusted separately to reflect changes in the relative importance of different topics [195, 90, 125].

The above approaches suffer fundamentally from the fact that a combination of linear representations and linear learning algorithms is used. They attempt to tackle dynamic changes in the multiple user interests by breaking up the task into single-profile representations and separate adaptation levels. Then they compose these elements into a global solution. As a result, they can't account for the continuous variety of changes. In practice, they are confronted with a large number of parameters, like learning coefficients and relative importance weights, that require optimisation, which may have to be performed separately for each individual user.

A biologically inspired approach to profile adaptation, derives from the application of Genetic Algorithms (GAs) or Memetic Algorithms (MAs). IF systems that adopt this approach, like [127, 171, 8, 111] and others, maintain a population of linear profiles that collectively represent the user interests. The difference between GAs and MAs is that the latter combine evolution with linear learning at the individual profile level (see section 2.13.2). MAs, in particular, are therefore able to tackle the trade-off between frequent changes in a user's short-term needs and progressive, radical changes in the long-term user interests. Nevertheless, this ability comes with a large computational cost. A large population of profiles is required for effective adaptation through evolution. In addition, the relative importance of topics represented by individual profiles is reflected by their fitness and not by the representation itself. Typically, each profile includes the same, fixed number of terms.

Finally, profile adaptation in the few connectionist approaches to IF has been achieved either using linear learning algorithms for updating the weights of terms and links [174] or Hebbian learning of link weights [108]. Although, these systems have influenced our work, they do not tackle multiple topics of interest. Single-topic profiles are described in both cases.

Similar to GAs, we draw analogies from biology, to achieve profile adaptation through a process of self-organisation, a common characteristic of living systems. The concept of self-organisation is not new in the domain of text processing. The Self-Organising Map (SOM) is a type of neural network that can map an originally high-dimensional document space onto a usually two-dimensional map grid that expresses content similarity between documents in an intuitive graphical fashion. Related documents appear in nearby grid locations. The SOM has been applied for the self-organisation of a large document collection [85] and supports the visualisation of the document space for interactive access and document retrieval [89]. We use self-organisation for a different purpose. Instead of producing a static graphical display of a document collection, the goal is to adapt our multi-topic profile both to short-term variations in the user's needs and to progressive, but potentially radical changes in long-term interests. In the next section we set the theoretical foundations of the self-organisation process, which is described in detail in section 5.2. It is then evaluated using virtual users in section 5.3. The results indicate the profile's ability to respond with structural and modifications to a variety of changes in a stream of feedback documents. The profile appears to be able to adapt to a variety of simulated changes in a virtual user's interests.

5.1 Networks, Self-Organisation and Autopoiesis

Work within philosophy, biology, cognitive science and social theory suggests that we should be looking at organisations, societies, the economy and language as 'living systems'. According to Fritjof Capra [29]:

> *"Living systems are integrated wholes whose properties cannot be reduced to those of smaller parts. Their essential, or 'systemic', properties are properties of the whole, which none of the parts have. They arise from the 'organizing relations' of the parts, i.e. from configuration of ordered relationships that is characteristic of that particular class of organisms, or systems."*

In computer science, this new perspective has given rise to a variety of biologically inspired algorithms and computational models, usually within the domain of *Artificial Life* (A-Life) . These include the aforementioned GAs and MAs, Neural Networks, Artificial Immune Networks, and others, each with significant scientific implications and practical applications [17]. Immune networks for example, have been recently applied for document classification [82] and for web site recommendations [124].

Central to this focus on living systems has been the concepts of *network*, i.e. a map of the 'organizing relations' of a living system's parts, and of *self-organisation*, i.e. the spontaneous emergence of order. As Capra puts it: "The pattern of life, we might say, is a network pattern capable of self-organisation" [29]. Recently the work of Watts and Strogatz [188] and of Barabási [13] have demonstrated that many biological, technological and social systems can be mapped as networks that exhibit common important characteristics. Language networks of terms (nodes) and links between related terms share these common characteristics [67, 126]. In general, the importance of the biologically inspired study of language has been recognised [81].

During the second half of the 20th century, various theoretical models were developed to account for self-organisation [29]. However, they all share three common characteristics:

1. Their interconnectedness renders self-organising systems non-linear.
2. Self-organising systems are open systems—energy and matter flow through the system—that operate far from equilibrium.
3. New structures and new modes of behavior are created in the self-organisation process.

Humberto Maturana and Francisco J. Varelas' *autopoietic theory* describes such a model of self-organisation [106]. Simply put, autopoietic theory tells us that a living system's organisation is embodied in its 'structure' (its components (nodes) and their interactions (links)) and the processes that this structure performs, which continuously reorganise the structure that generates them. Adaptive behaviour then is the phenomenological result of the *structural coupling* of the system with its environment (another system). Structurally coupled systems are mutually affective through feedback loops that perturb the plastic, self-organising structure of these systems and therefore the processes they perform [141]. Through structural coupling the embodied system appears to respond, adapt, to changes in the environment. According to Maturana and Varela, embodiment and structural coupling are the basis for the emergence of language and cognition in general. The importance of the autopoietic approach to cognition for information systems and Artificial Intelligence has been stressed by Mingers [115].

We have already complied with the first of the above characteristics of self-organising systems, non-linearity. We have established, in the previous chapter, non-linear document evaluation functions, based on the hierarchical ordering of terms. In fact, hierarchical organisations are common in nature. Evidence for the hierarchical organisation of biological and other complex networks have been recently provided [142]. According to Richard Dawkins: "If animals such as crickets, who work with general memory of past fights, are kept together in a closed group for a time, a kind of dominance hierarchy is likely to develop. An observer can rank the individuals in order. Individuals lower in the order tend to give in to individuals higher in the order" [43]. An analogy with our hierarchical approach to multi-topic information filtering is possible. Terms in the profile are ordered according to their weight, a measure of their dominance of appearance in relevant documents. Memory is manifested in terms of links between terms of different order. Terms lower in the hierarchy "tend to give in", disseminate energy, to terms higher in the hierarchy through the corresponding links. Such collaborations are stronger within each hierarchy representing a separate topic of interest. Collectively, hierarchies compete with each other for the common shared resource, the user's positive feedback, and therefore for representational importance. The resulting non-linearity allows for document evaluation according to multiple topics of interest. We will argue that the process described in the next section assigns to the profile the remaining two characteristics of self-organisation.

5.2 Self-Organisation Process

We introduce a process comprising five deterministic, but interrelated, steps, that allows the profile to self-organise in response to changes in the user interests. The latter are reflected by the documents that the user chooses to read. Of course, what the user reads affects the further evolution of user interests. As we have already argued, this choice does not have to be constrained to the documents that the IF system presents to the user. Other sources of information may still be available. Profile adaptation requires that the relevance of read (or just reviewed) documents is specified through relevance feedback. Although typically it is the user that has to declare read documents as relevant or not, techniques which alleviate this extra burden by trying to implicitly induce the relevance of documents from the user actions [123, 177] and potentially body language [136] are being developed. For the rest of this chapter, we will assume that relevance feedback on at least some read documents is indeed available and that it reflects, to some extent, changes in the user interests.

The self-organisation process causes, in response to feedback documents, two kinds of structural modifications. *Tuning* adjusts the weights of existing profile components (terms and links). This subprocess is similar to learning (section 2.13.1), but, as we will see, a fixed learning coefficient is not employed. The pace of adaptation is not constant, but depends on the relevance feedback and the profile's current structure. Nevertheless, as learning, tuning alone cannot adapt the profile to radical changes in the user interests, like the emergence of a new topic of interest. We use *alteration* to refer to the subprocess responsible for adding new components to the profile and for removing existing, no longer competent ones. Alteration, like the evolutionary process in GAs (section 2.13.2), allows a profile to acquire the additional terms and generate the links necessary to learn a new topic of interest. Similarly, existing terms and links representing an *unexcited* topic, i.e. a topic that does not receive positive feedback, or a *non-relevant* topic, i.e. a topic that explicitly receives negative feedback, are eventually purged, causing the topic to be forgotten. But, unlike GAs, alteration applies to a non-linear, multi-topic profile, and not a population of linear, single-topic profiles.

We tackle the trade-off between progressive, radical changes in the long-term user interests and faster modest changes in the short-term needs by combining the effects of tuning and alteration into a single self-organisation process. We thus avoid the separate adaptation levels of dual profiles (section 2.13.1) and the computationally expensive combination of evolution and learning in MAs (section 2.13.2). Tuning and alteration are not distinct processes, but the net effect of the following deterministic steps.

5.2.1 Step 1: Extract Informative Terms

The number of documents that received relevance feedback may vary, depending on the characteristics of the user, the time constraints, the success

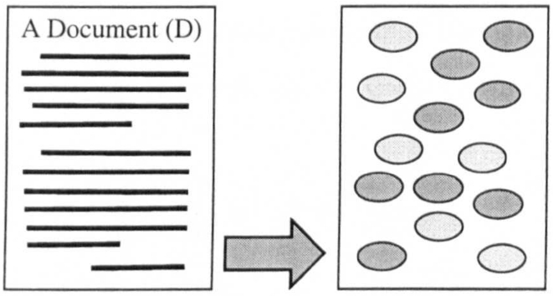

Figure 5-1: Step 1: Term weighting and extraction. (Terms in light grey already appear in the profile and terms in dark grey do not.)

of the filtering process and other parameters. It may range from just one document to many. The process should be able to cope with this variability. Our solution to this problem is based on a variation of RelDF (equation 3.1). More specifically, if R is the number of documents that received either positive or negative feedback, then the weight of a term t that appears in these documents is calculated using equation 5.1. In this way we have attempted to account for the statistical importance of the sample of feedback documents. If the sample is statistically important ($R > 20$), then the weight of a term is calculated using the original RelDF equation. In the opposite case ($R \leq 20$), we reflect the lack of confidence in the sample by dividing with a constant $R = 20$.

$$RelDF_t = \begin{cases} \frac{r}{R} - \frac{n}{N} & \text{if } R > 20 \\ \frac{r}{20} - \frac{n}{N} & \text{if } R \leq 20 \end{cases} \tag{5.1}$$

In the special case of just one relevant, or nonrelevant, document the above equation takes the form of equation 5.2, which represents the online version of RelDF. Its simplicity has been an additional reason for concentrating on RelDF in this chapter. It allows profile adaptation based on just one feedback document. To demonstrate this ability our description of the self-organisation process will concentrate on just one feedback document, but the same process is applicable for more documents.

$$RelDF_t^o = w_t^D = \frac{1}{20} - \frac{n}{N} \tag{5.2}$$

More specifically the first of the five steps involves the weighting and extraction of informative terms. Given a document D, after stop word removal and stemming, equation 5.2 is applied to weight terms in the document. To extract the most informative terms an appropriate threshold is required. As we will see this is the main parameter of the process that requires optimisation. Nevertheless, for the forthcoming experiments we have chosen a threshold equal to 0.03. In contrast to the two- and three-topic experiments of the previous chapter (sections 4.2.2 and 4.2.3 respectively), this value was chosen small enough to substantially increase the number of extracted terms. The term extraction process results in a set of weighted terms (fig. 5-1), some of which may already appear in the profile (fig. 5-1: terms in light grey) and some may not (fig. 5-1: terms in dark grey). As we will explain shortly, the overlap between the profile and the set of extracted terms has a significant effect on the adaptation pace.

5.2.2 Step 2: Update Profile Term Weights

The second step of the self-organisation process concentrates on those extracted terms that already appear in the profile. For each such profile term t, an updated weight w_t' is calculated using equation 5.3. In the case of a relevant document D, w_t' is calculated by adding to the profile term's initial weight w_t, its weight w_t^D in the document (or documents). In the case of a nonrelevant document, w_t^D is subtracted from the initial weight w_t. The weight of profile terms that don't appear in the extracted set remains unchanged ($w'_t = w_t$).

$$w'_t = \begin{cases} w_t + w_t^D & \text{if } D \text{ relevant} \\ w_t - w_t^D & \text{if } D \text{ nonrelevant} \\ w_t & \text{if } t \ni D \end{cases} \tag{5.3}$$

Furthermore, in the case of a relevant document, we sum up the additional weights that have been assigned to the profile terms and then subtract this sum evenly from all profile terms. This process is expressed by equation 5.4, where NP is the number of profile terms. The opposite takes place in the case of a nonrelevant document. Therefore, given a profile with a specific set of terms, this last process assures that the overall weight of profile terms remains stable.

$$w''_t = \begin{cases} w'_t - \frac{\sum_{t \in D} w_t^D}{NP} & \text{if } D \text{ relevant} \\ w'_t + \frac{\sum_{t \in D} w_t^D}{NP} & \text{if } D \text{ nonrelevant} \end{cases} \tag{5.4}$$

The net effect of the above process is the tuning of existing term weights. Profile terms that appear in a relevant document increase their weights at the expense of those that do not. The opposite takes place in the case of a nonrelevant document. In contrast to traditional linear learning algorithms (e.g. Rocchio's) however, the relative increase, or decrease, in the weight of profile terms, is not defined only by the weight of extracted terms and a constant learning coefficient. It depends on a series of additional parameters: the number of profile terms, the number of extracted terms, and their overlap. As the number of profile terms grows in relation to the average number of extracted terms the relative changes in weight that a document causes reduce.

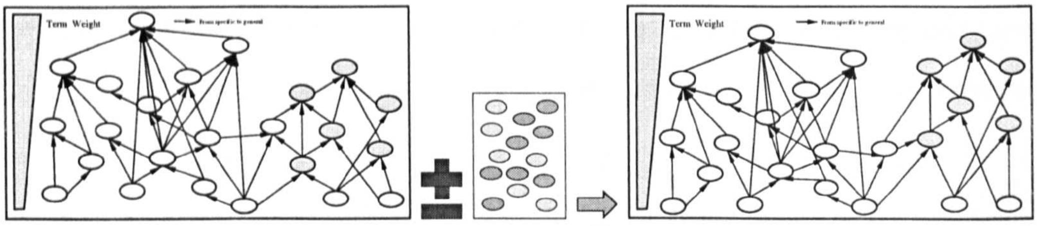

Figure 5-2: Step 2: Redistribution of term weights. (In light grey, profile terms that have been extracted from the relevant document and that get reinforced.)

Essentially, instead of adjusting a learning coefficient over time, as in the case of reinforcement learning, the rate of tuning depends on the current profile structure. The more a profile learns the more resistant to tuning it becomes. Finally, tuning does not only adjust the weight of profile terms but also causes changes in the hierarchy's ordering, thus affecting further what the profile represents. Figure 5-2 illustrates this effect in the case of an example profile representing two topics of interest and a document relevant to one of these topics. Profile terms that have been extracted from the relevant document (fig. 5-2:terms in light grey) have their weight reinforced, while the weight of the rest of the profile terms decreases. The reinforced terms climb higher in the hierarchy in relation to the rest of the profile terms. So during document evaluation, it is more likely that reinforced terms will receive, rather than disseminate energy.

5.2.3 Step 3: Remove Incompetent Profile Terms

Another side-effect of the decrease in the weight of profile terms, which is caused either implicitly in the case of a relevant document, or explicitly in the case of a nonrelevant one, is that some profile terms "run out of weight". In our case this means that the weight of some terms becomes less than zero. Following the example of figure 5-2, in the left part of figure 5-3 such terms

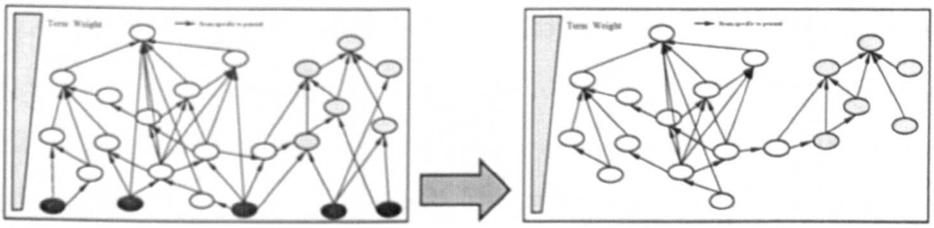

Figure 5-3: Step 3: Removal of incompetent terms. (Terms that "ran out of weight" appear in black and reinforced terms in light grey.)

are depicted as black nodes. In this third step of the self-organisation process, terms that run out of weight are purged from the profile together with all of their links to other terms (fig. 5-3). With this mechanism, we aim to remove terms that were mistakenly added to the profile, or that have become incompetent due to changes in the user interests. As already mentioned, this kind of alteration gives to the profile the ability to forget an unexcited or a non-relevant topic. At the same time we sum up the initial weight, i.e. the weight with which a term had entered the profile (see next section), of the purged profile terms (equation 5.5). The reason for this will be explained in the next section.

$$W_{purged} = \sum_{t \text{ purged}} w_t^{init} \tag{5.5}$$

5.2.4 Step 4: Add New Terms

Having updated the weight of profile terms and removed incompetent terms, at step 4 of the self-organisation process, those terms that have been extracted from a relevant document and do not already appear in the profile are added to the profile. For our example, this process is depicted by figure 5-4. The initial weight of each added term is equal to the term's weight in the document ($w_t^{init} = w_t^D$). The addition of new terms does not influence

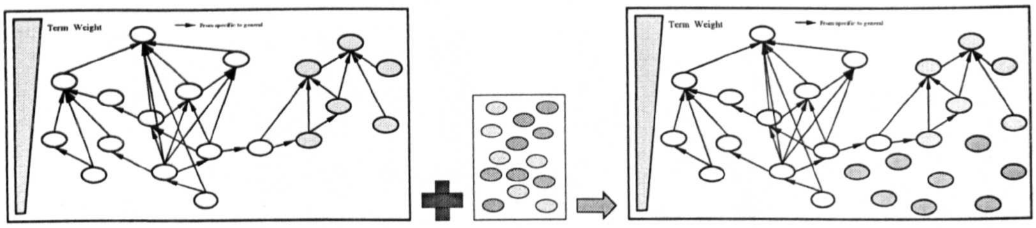

Figure 5-4: Step 4: Adding new profile terms. (In dark grey, extracted terms that were not already included in the profile.)

the weights of existing terms. Therefore, with the addition of every new term the overall weight of profile terms increases. The number of terms that are added depends on the semantic novelty of the relevant document in relation to what is already being represented. For example, if the profile represents the topics *Knowledge Management* and *Intelligent Agents*, a relevant document about, let's say, *Geography*, is very likely to contribute a large number of new informative terms to the profile. Whereas, a document about "the application of intelligent agents for knowledge management" will contribute very few new terms, if any, since it is likely that the corresponding informative terms are already included. We should also stress, that the added terms do not replace terms that have been purged in the previous step. There is no relationship between the number of purged terms and added terms. The number of profile terms is not fixed, but rather changes dynamically according to user feedback.

Finally, after the new profile terms are added, we subtract evenly from all profile terms the sum of the initial weights of those terms that have been purged in the previous step. This is expressed by equation 5.6, where NP' is the number of profile terms after the addition of new terms. Practically speaking, this is done to avoid the escalation of the overall weight of profile terms due to the addition of new weight with every new term. Its importance in terms of self-organisation however, is that it renders the profile open to the environment: weight (energy) flows through the profile. The amount of weight (energy) that every new term adds to the profile is removed from the profile when and if the term is purged.

This weight decaying process may result in some profile terms with negative weight. These terms do not affect the profile's functionality and are maintained in the profile to be purged as part of the next adaptation cycle. Alternatively, we could backtrack to step 3, but this could cause an unnecessary iterative process, involving steps 3 and 4, until no further terms run out of weight. Instead, we have chosen to perform this action after new terms are added to the profile, so that W_{purged} is evenly subtracted from more terms and therefore fewer terms may end up with negative weights. However, a side effect of this process is that new terms may loose part of their initial weight.

$$w_t''' = w_t'' - \frac{W_{purged}}{NP} \tag{5.6}$$

5.2.5 Step 5: Re-establish Links

So far we discussed how tuning and alteration takes place based on terms extracted from a feedback document. The weight of existing terms has been updated, incompetent profile terms have been removed and new informative terms have been added. It is now time to turn to links. For this purpose we refer back to the link generation process described in section 3.4. There, we had described how correlations between terms are identified and how the corresponding links are weighted. Using a sliding window approach, two terms t_i and t_j were linked if they appeared at least once within the window. The weighting of the link between the two terms involved the following parameters (equation 3.4): the number of times fr_{ij} that t_i and t_j appeared within

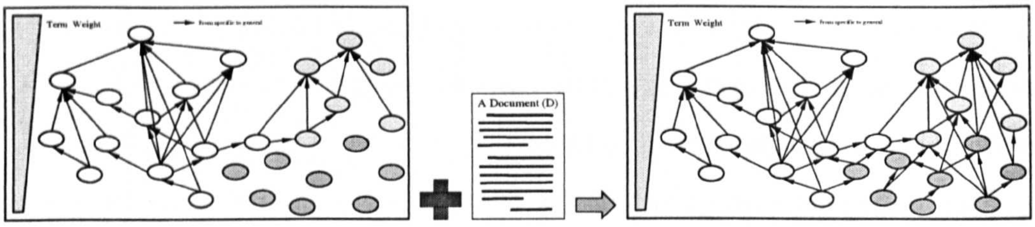

Figure 5-5: Step 5: Link generation and weighting

the window, their respective frequencies fr_i and fr_j in the initialisation documents and the average distance d_{ij} between them. Although in section 3.4 the link generation process was applied to a set of user-specified initialisation documents, all of the above parameters can be updated online.

More specifically, for each profile term t_i we may maintain in memory its overall frequency fr_i in the relevant documents processed so far. For each new relevant document D that the term appears in, the term's frequency is simply updated using equation 5.7, where fr_i^D is its frequency in the document. Exactly the same process can be used for updating the frequency fr_{ij} with which t_i and t_j appear in the sliding window (equation 5.8). However, for updating online the average distance between the two terms, one has to maintain the aggregate distance $dist_{ij}$ between the two terms in the processed documents. $dist_{ij}$ can be updated online using equation 5.9, where $dist_{ij}^D$ is the aggregate distance between the two terms in D. We can then calculate the new average distance d'_{ij} using equation 5.10.

$$
\begin{aligned}
fr'_i &= fr_i + fr_i^D && (5.7)\\
fr'_{ij} &= fr_{ij} + fr_{ij}^D && (5.8)\\
dist'_{ij} &= dist_{ij} + dist_{ij}^D && (5.9)\\
d'_{ij} &= \frac{dist'_{ij}}{fr'_{ij}} && (5.10)
\end{aligned}
$$

This fifth and final step of the process takes advantage of the ability to update online the parameters that are involved in link weighting. After adding the new terms, the relevant document is processed using a sliding window of size 10 to identify links between profile terms and update the above parameters using equations 5.7 to 5.10. Once links have been established and the parameters updated, the weight of new links and the updated weight of existing links is calculated using the original equation 3.4 of section 3.4. For the running example, the result of this fifth step is depicted by figure 5-5. Since the feedback document is about one of the represented topics, it is more likely that the extracted terms will be linked to each other and to existing terms corresponding to that topic.

Of course, this is only a simple solution for the profile's tuning and alteration in terms of links, and has certain disadvantages. It is inefficient for example, to maintain in memory and keep updating the involved parameters for all of the documents processed so far. This would imply that frequency and aggregate distance values could keep increasing indefinitely, especially in the case of persistent terms. Consequently, the weight of links between persistent and new terms could be underestimated, even if it is important given the current user interests. To overcome these drawbacks one may employ two remedies. Either a maximum value for each of the parameters is defined and the values are periodically normalised so that none of them exceeds this maximum, or only the frequencies and aggregate distance for the last, let's say, 30 documents, is maintained for each term and link. Nevertheless, for the experiments conducted further in this chapter, we did maintain the complete frequencies and aggregate distances for all documents, because only a small number of training documents is used. For our further reseach, we also intend to investigate less ad hoc solutions like Hebbian learning.

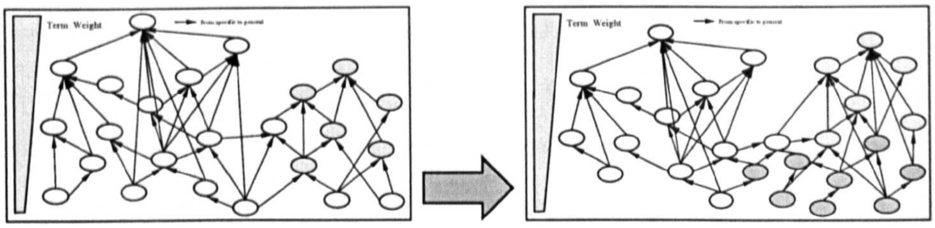

Figure 5-6: The effect of adaptation on the profile

5.2.6 Overview

These deterministic, interwoven, steps involve the weighting and extraction of informative terms from a feedback document, the updating of the weight of profile terms, the removal of incompetent terms, the addition of new terms and finally the identification and weighting of links. During this process the profile is open to the environment. Energy in the form of term weight flows through it. The effect is constant structural change that maintains the profile far from equilibrium. We can therefore argue that the process exhibits the second characteristic of self-organisation.

We illustrated such a structural change loosely, with a hypothetical profile representing two topics and a document about one of these topics. The overall change (fig. 5-6) shows that tuning and alteration combined, cause an increase in the size of the hierarchy corresponding to the topic discussed in the feedback document and a decline in the size of the hierarchy corresponding to the topic that did not receive positive feedback. Through self-organisation, the profile responds to feedback with structural modifications, which of course affect document evaluation. We evaluate the effectiveness of this process for profile adaptation in the next section.

The presentation has concentrated on one feedback document, but clearly exactly the same 5 steps may be applied to a batch of documents. Equation 5.1 would then be used to extract a set of informative terms and the rest of the steps would be performed unaltered. Nevertheless, to avoid the need to define appropriate document batches, the following experiments have been conducted by adapting the profile on a per document basis.

It is also important to note at this point, that if we exclude the steps related to the removal, addition and updating of link weights, the self-organising process is applicable in the case of an unconnected profile representation. This allows the experimental comparison between the hierarchical profile and an unconnected profile with the same terms.

5.3 Experimental Evaluation

The recent increased interest in adaptive information filtering has been reflected by the incorporation of the adaptive filtering track as part of TREC-7 and subsequent TRECs. However, as we have described in section 2.14.2, according to TREC's guidelines, profiles are tested for their ability to adapt to changes over time in the content of documents that relate to each of the 84 RCV1 topics that TREC adopts. Such changes are not only loosely controlled, but furthermore, they don't reflect possible radical changes in the user interests, like loss of interest in a topic or the emergence of a new topic of interest.

Alternatively, virtual or synthetic users have been used to simulate such radical changes. As described in section 2.14.3, given a preclassified collection of documents, a virtual user's current interests are defined by a subset of the classification topics. Training documents that relate to the topics in the subset comprise the positive feedback. To simulate the loss of interest in a topic, it is removed from the subset. The corresponding documents are typically used as negative feedback. The emergence of a new topic of interest is simulated by adding the topic to the subset. Virtual users have already been employed for the evaluation of adaptive profiles [196, 90, 171]. However, all of these suffer from the small number of test documents and the relatively ad-hoc nature of the evaluation methodology used.

5.3.1 Experimental Methodology

To evaluate our approach to AIF, we have instead synthesised virtual users and simulated changes in their interest using the same two- and three-topic combinations that we experimented with in the previous chapter (tables 4.5 and 4.8 respectively). We have defined a methodology based on a further variation of TREC's routing guidelines.

Similarly to [196], we tested a profile's ability to adapt on two learning and two forgetting tasks. A task is a scenario that describes a radical change in a virtual user's interests. It consists of a series of two topic combinations separated by "→", symbolising the interest change. For example, a virtual user may be initially interested in topics R1/R2 (combination I) and then an additional interest in topic R3 emerges. The learning task for this scenario is formulated as $R1/R2(I) \rightarrow R1/R2/R3(I')$. In this fashion we defined the following general tasks, where C represents a two-topic combination, C' the corresponding three-topic combination and T_i a specific topic:

(α) $T_1/T_2(C)$ This learning task tests the ability of an empty profile to learn from scratch two topics of interest (T_1 and T_2) in parallel. This task involves only one two-topic combination and therefore it doesn't simulate a radical change of interest. Its difference to the two-topic experiments of section 4.2.2 is that profile training takes place online.

(β) $T_1/T_2(C) \rightarrow T_1/T_2/T_3(C')$ Here we test an existing profile's ability to learn an additional topic of interest. The virtual user is initially interested in topics T_1 and T_2 alone and after some time an interest in the third topic T_3 emerges in addition to the existing interests.

(γ) $T_1/T_2/T_3(C') \rightarrow T_1/T_2(C)$ The first forgetting task tests, in a way symmetrical to task (β), an existing profile's ability to forget one of the initial three topics of interest. Here the user is initially interested in topics T_1, T_2 and T_3 and then the interest in the first two topics is maintained while the interest in topic T_3 is lost. T_3 becomes unexcited.

(δ) $T_1/T_2/T_3(C') \rightarrow T_1/T_2/\neg T_3$ The second forgetting task differs from task (γ) in that after the initial interest in three topics, the virtual user maintains the interest in the first two and explicitly specifies with negative feedback that the third topic ($\neg T_3$) is non-relevant.

For each general task we experimented with specific task formulations which are summarised in table 5.1. These reuse the two and three topic combinations from chapter 4. Each topic combination in a task corresponds to a training phase, a period of time during which the virtual user's interests remain stable. During a training phase, a profile is trained online using a set of documents comprising the first 30 training documents per topic in the combination (60 for two topics of interest and 90 for three). Only in the case of a negated topic $\neg T$ are the corresponding training documents used as negative feedback.

We only used the first 30 training documents per topic to enable a common experimental setting for all combinations including those with a small number of training documents. However, this implies that training documents used as part of the first training phase are reused during the second training phase. Although this practice is not realistic, nevertheless, it is not

Table 5.1: Two learning and two forgetting tasks

α tasks	
α.1	$R1/R2(I)$
α.2	$R7/R8(II)$
α.3	$R29/R68(III)$
α.4	$R10/R32(IV)$
α.5	$R6/R21(V)$
α.6	$R41/R79(VI)$
β tasks	
β.1	$R1/R2(I) \rightarrow R1/R2/R3(I')$
β.2	$R7/R8(II) \rightarrow R7/R8/R9(II')$
β.3	$R29/R68(III) \rightarrow R29/R68/R1(III')$
β.4	$R10/R32(IV) \rightarrow R10/R32/R50(IV')$
β.5	$R6/R21(V) \rightarrow R6/R21/R20(V')$
β.6	$R41/R79(V) \rightarrow R41/R79/R58(VI')$
γ tasks	
γ.1	$R1/R2/R3(I') \rightarrow R1/R2(I)$
γ.2	$R7/R8/R9(II') \rightarrow R7/R8(II)$
γ.3	$R29/R68/R1(III') \rightarrow R29/R68(III)$
γ.4	$R10/R32/R50(IV') \rightarrow R10/R32(IV)$
γ.5	$R6/R21/R20(V') \rightarrow R6/R21(V)$
γ.6	$R41/R79/R58(VI') \rightarrow R41/R79(V)$
δ tasks	
δ.1	$R1/R2/R3(I') \rightarrow R1/R2/\neg R3$
δ.2	$R7/R8/R9(II') \rightarrow R7/R8/\neg R9$
δ.3	$R29/R8/R1(III') \rightarrow R29/R68/\neg R1$
δ.4	$R10/R32/R50(IV') \rightarrow R10/R32/\neg R50$
δ.5	$R6/R21/R20(V') \rightarrow R6/R21/\neg R20$
δ.6	$R41/R79/R58(VI') \rightarrow R41/R79/\neg R58$

statistically incorrect.

Documents in a training set have been ordered according to their date of publishing. Therefore, the distribution of documents per topic during an online training phase has not been homogeneous, but rather reflected the temporal variations in the publication date of documents about each topic. For our experiments, we make the assumption that these variations reflect changes in a virtual user's short-term needs. In a real situation however, other parameters, like the availability of documents about a certain topic in the accessible information space, might cause similar variations.

To evaluate a profile, it is tested periodically during the last training phase in each task (task α has only one training phase). In other words, after a radical change of interest has occurred (task α does not simulate a radical change). After every five training documents the profile is used to filter the complete test set. A separate AUP score was then calculated for each topic, on the basis of the best 3000 scoring documents. We thus use a variation of TREC's routing guidelines to measure the profile's ability to adapt in response to variations in a stream of feedback documents.

We have experimented with both unconnected profiles using the inner product based evaluation function S0 (equation 3.2) and hierarchical profiles using either of the proposed evaluation functions, S1 (equation 4.3) and S3 (equation 4.5)[1]. We conducted two parallel experiments. The unconnected profile was adapted using the proposed process, without the link related steps. The complete adaptive mechanism was used in the case of the hierarchical profile. Both the unconnected and hierarchical profiles involved the same terms and the same processes for updating their weights, and for removing

[1] Initial analysis of the results for the two- and three-topic experiments had not revealed the difference between S1 and S2. Consequently, here we have chosen to concentrate on the simplest (S1) and the most elaborate (S3) of the three introduced functions.

and adding new terms. Thus, we were able to evaluate the effect of links and in general of the network's topology on the profile's performance during adaptation.

5.3.2 Task α: parallel interest in two topics

Figures 5-7 to 5-12 show the experimental results for the α tasks, which test the ability of profiles to learn two topics of interest in parallel. For each task two graphs have been generated. The first graph represents for each topic of interest and evaluation function the fluctuation of AUP score over the training phase. For reasons discussed in the previous chapter (section 4.2), for some tasks there is a significant difference in the scores achieved for the topics of interest. For these combinations a secondary y-axis was used to facilitate the visualisation of the results. The second graph shows the distribution of documents per topic in the training set. The values on the y-axis count the number of documents per topic within each 5 document interval, between subsequent profile evaluations. We have assumed that variations in the distribution of feedback documents reflect frequent changes in a virtual user's short-term needs. Since this task does not simulate a radical change of interest, it allows us to concentrate on how the profile responds to such short-term variations in the feedback stream.

For most tasks, the results do not show a progressive increase in the score of the two topics being learned. Such a behaviour is only clear for tasks α.3 and α.5, which comprise relatively unrelated topics. For tasks α.1 and α.2, which comprise related topics with a large number of documents in the test, it appears that a few informative terms extracted from the initial training documents are sufficient for increased performance. A similar observation has been made in the single-topic experiments of section 4.2.1. In the case of task $\alpha.4$, the two topics are not learned in parallel. Topic R32 is learned first, followed by topic R10. Finally the results for task $\alpha.6$ show that for the most part of the training period only one of the two topics (R79) is learned. The score for topic R41 increases only towards the end of the training period.

We observe, that in most cases, fluctuations in the score of the two topics in each task are roughly symmetrical. When the score for one topic drops the score for the other increases and vice versa. A comparison between a topic's score and the corresponding distribution of training documents, reveals a correspondence. When more feedback documents about a certain topic are processed, its score increases, while the score of the less excited topic drops. For example, in the extreme case of task $\alpha.4$, the training set is initially dominated by documents about topic R32 causing an increase in its score. For the same period topic R10 is not learned. Subsequently, and for a period of more than 20 documents all training documents are about topic R10 and its score increases substantially, while the score for topic R32 drops. Finally, the last training documents are again about topic R32 only, and the score for topic R10 drops, but this time, no significant increase in the score of topic R32 is noted. Similar observations can be made for tasks $\alpha.1$, $\alpha.2$ and $\alpha.6$. The distribution for tasks $\alpha.3$ and $\alpha.5$ are more homogeneous and these are the cases where the profile appears to learn both topics in parallel. We should note, that the fluctuations in score are exaggerated because a fixed number (3000) of evaluation documents is used. Nevertheless, they suggest that the profile responds to variations in the distribution of feedback documents. It adapts, according to our assumption, to frequent changes in a virtual user's short-term needs.

The problem is that the profile appears to be too responsive. As a result, a topic may be quickly forgotten in absence of feedback documents (see for

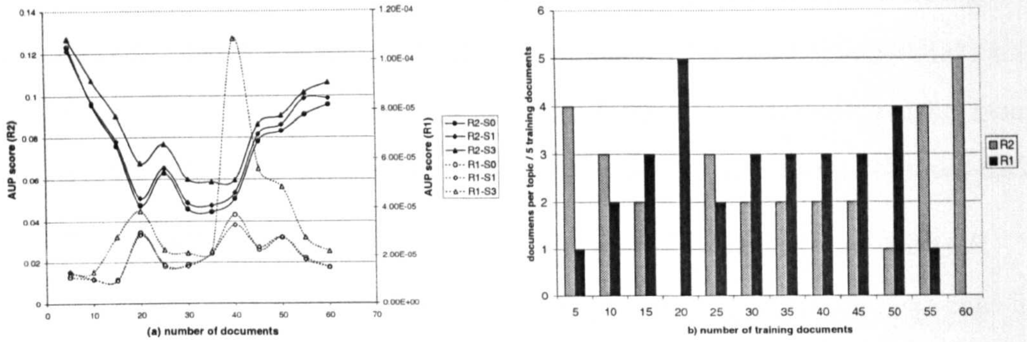

Figure 5-7: Results for for task α.1: (a) AUP score fluctuations per topic, (b) training document distribution per topic

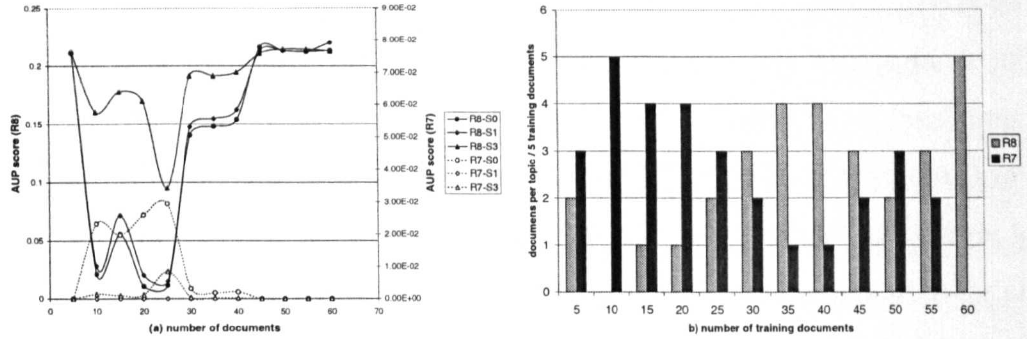

Figure 5-8: Results for for task α.2: (a) AUP score fluctuations per topic, (b) training document distribution per topic

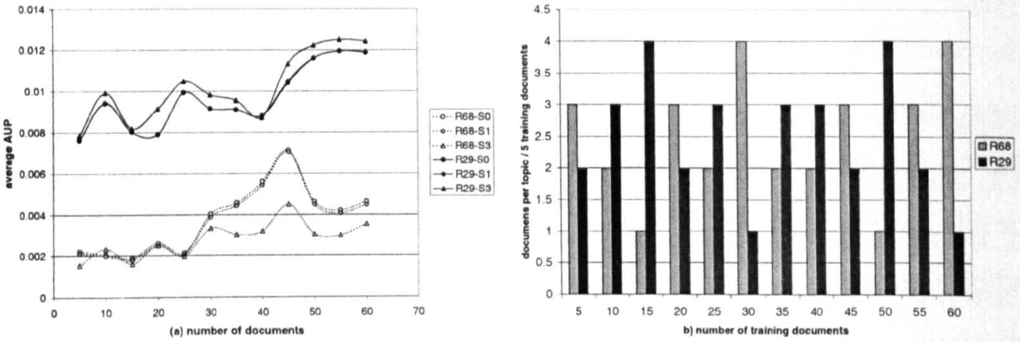

Figure 5-9: Results for for task α.3: (a) AUP score fluctuations per topic, (b) training document distribution per topic

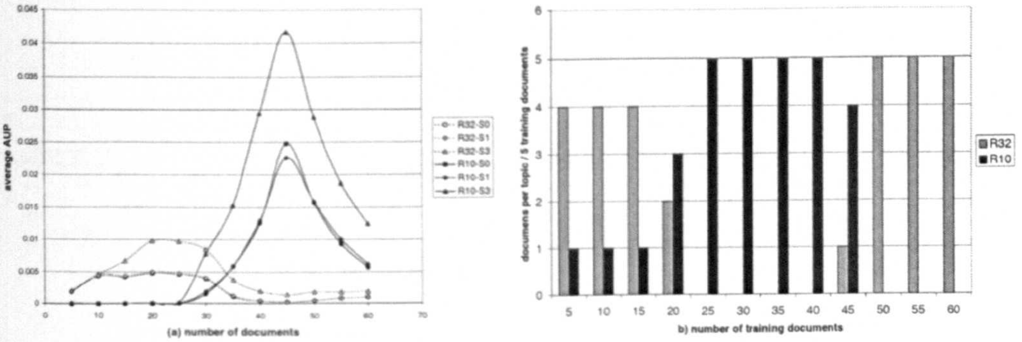

Figure 5-10: Results for for task $\alpha.4$: (a) AUP score fluctuations per topic, (b) training document distribution per topic

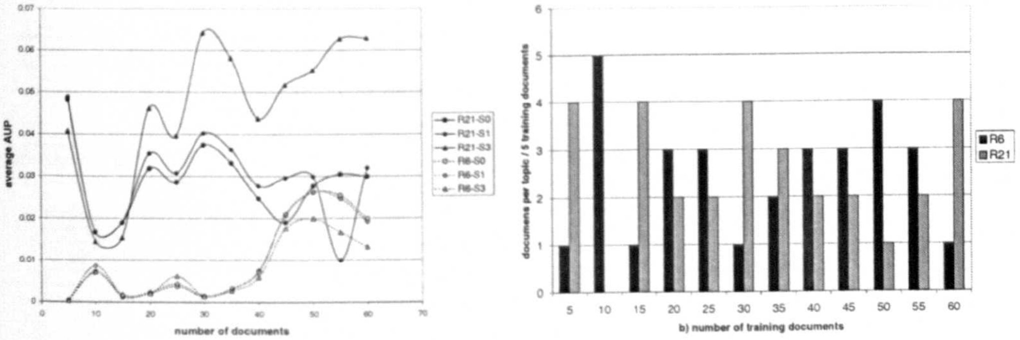

Figure 5-11: Results for for task $\alpha.5$: (a) AUP score fluctuations per topic, (b) training document distribution per topic

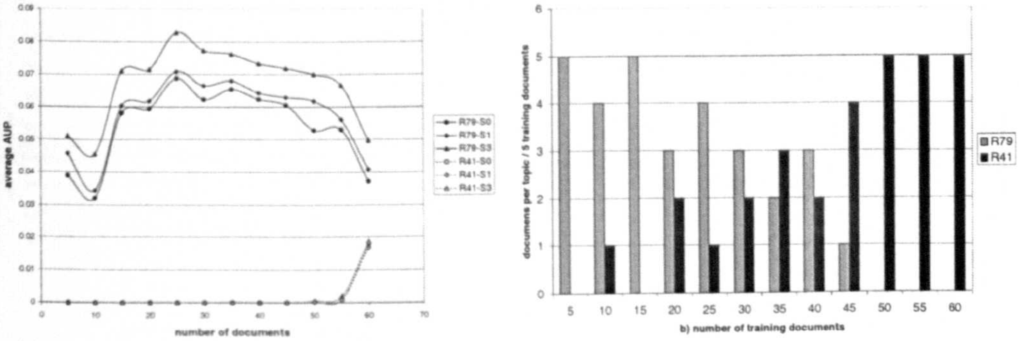

Figure 5-12: Results for for task $\alpha.6$: (a) AUP score fluctuations per topic, (b) training document distribution per topic

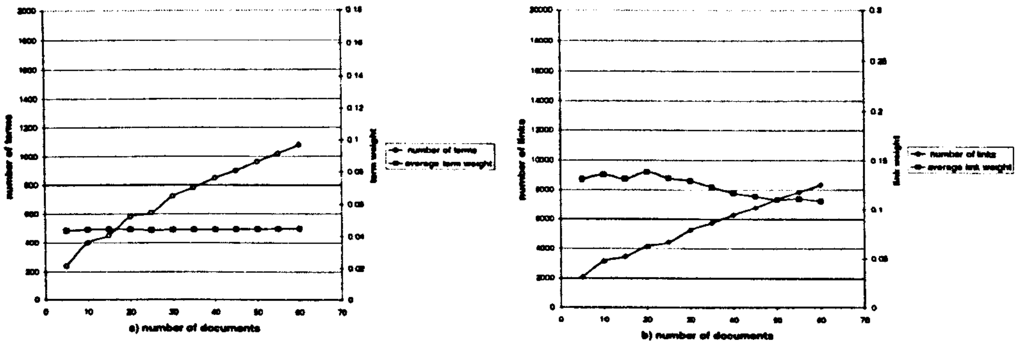

Figure 5-13: Profile statistics for task α.1: a) number of terms and average term weight b) number of links and average link weight

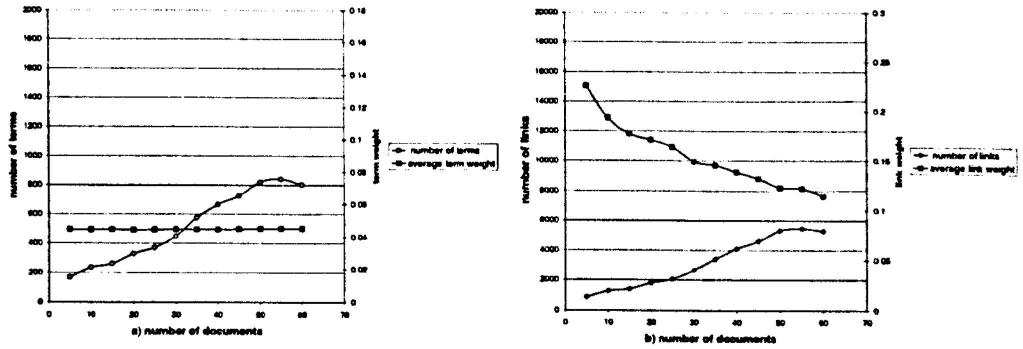

Figure 5-14: Profile statistics for task α.2: a) number of terms and average term weight b) number of links and average link weight

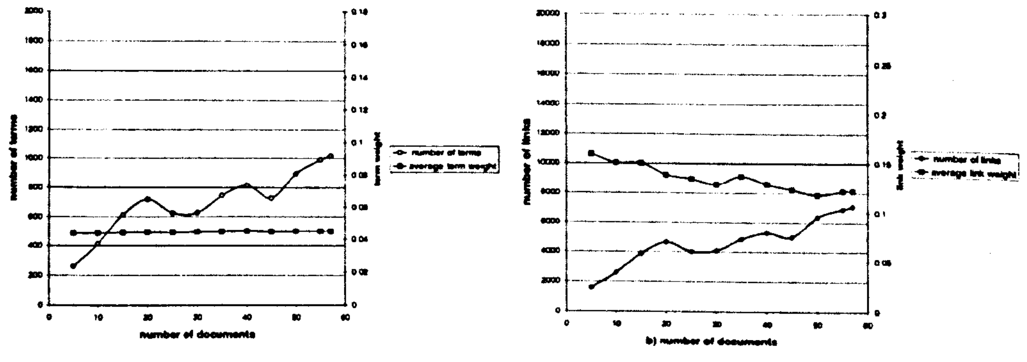

Figure 5-15: Profile statistics for task α.3: a) number of terms and average term weight b) number of links and average link weight

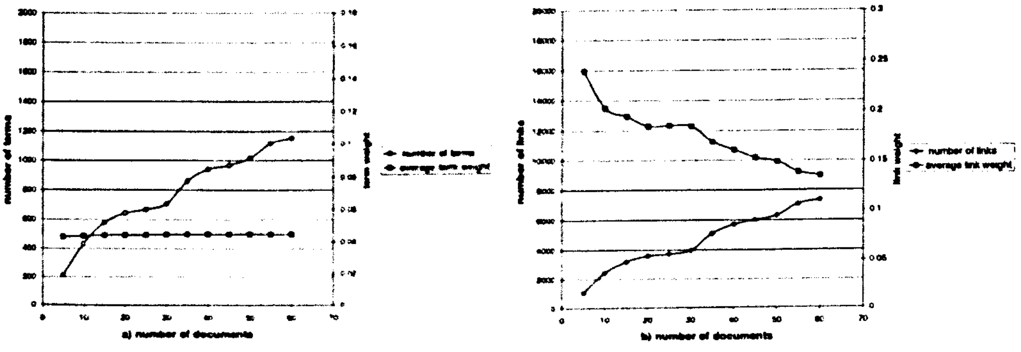

Figure 5-16: Profile statistics for task α.4: a) number of terms and average term weight b) number of links and average link weight

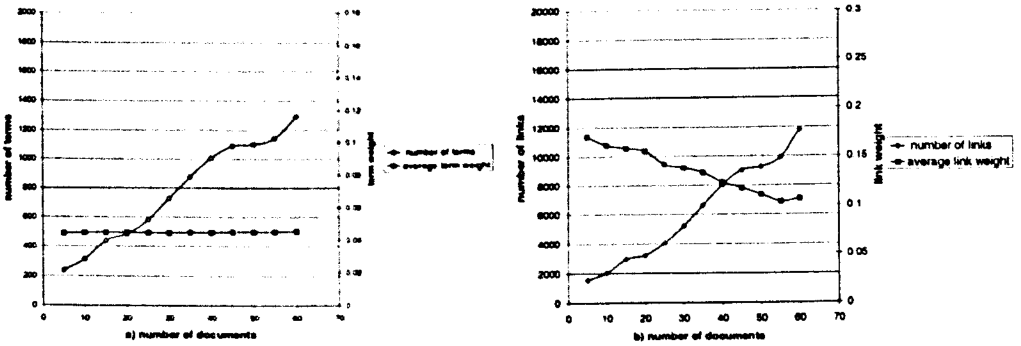

Figure 5-17: Profile statistics for task α.5: a) number of terms and average term weight b) number of links and average link weight

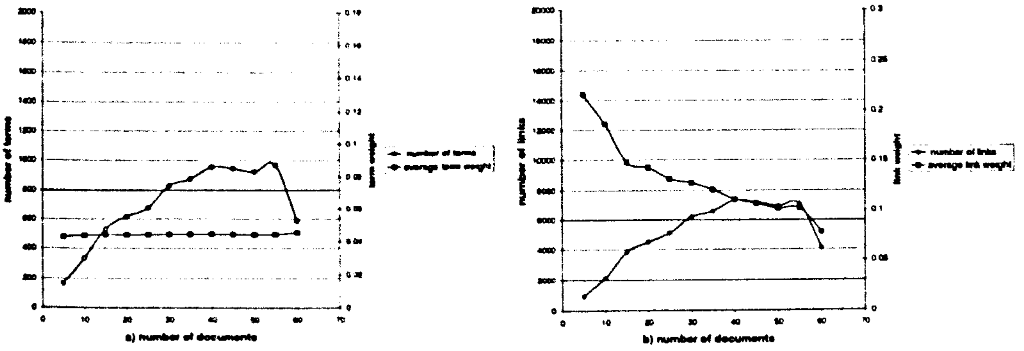

Figure 5-18: Profile statistics for task α.6: a) number of terms and average term weight b) number of links and average link weight

example task α.4). One possible explanation is that the generated hierarchies are relatively shallow. The training documents do not provide enough informative terms, possibly due to their small number and their journalistic style (short length and not too technical vocabulary). As we will shortly see, this hypothesis is supported by the fact that the average weight of profile terms is relatively small. Note also, that the persistence of terms in the profile depends on their weight. It takes more feedback cycles for a term with a large weight to run out of weight. As hierarchies develop, they become more persistent. On the other hand, shallow hierarchies can be quickly purged from a profile.

Another interesting finding is that the hierarchical profile using S3 exhibits clearly the best performance for most topics. It is not the best approach approach only for topic R7 in task α.2, topic R29 in task α.3 and topic R6 in task α.5, the least scoring topics in each combination. The hierarchical profile using S1 is also at least as good as the unconnected profile using S0 and in some cases slightly better. The unconnected profile using S0 is clearly the best approach only for the least scoring topic (R7) in task α.2.

The above qualitative results measure the profile's filtering performance, when the profile is trained online with documents about two different topics. They do not show the structural changes that cause the observed fluctuations in performance. For that purpose we should be able to visually monitor the profile, but this is another challenging issue left for future research. Here, we simply present, with figures 5-13 to 5-18, some macroscopic measures that only provide indications of the profile's structural self-organisation. Each figure corresponds to a task and includes two graphs. The first shows how the number of profile terms and their average weight change over the training period. A second graph shows, for the same training period, changes in the number of links and their average weight.

For most tasks the number of profile terms increases throughout the training period. The profile grows with the addition of more terms than those removed. Fluctuations in the number of terms are indicated in task α.3 and a drop towards the end of the training period in task α.6. It seems that, due to the small number of training documents, the profile does not acquire enough terms to reach some balance in their number. Furthermore, it is important to note that the increase in the number of terms does not reflect the above fluctuations in score. This makes us believe that the latter are mainly due to the tuning of existing profile components, the redistribution of term weights and the update of link weights in response to feedback.

In all cases, the average weight of profile terms appears to remain constant, with values around 0.044. In fact, a narrower y-axis scale would reveal that during the training period there is an overall, non-monotonic increase in the average term weight, from values around 0.043 to values around 0.045. Here we keep the scale used in the rest of the tasks and this behaviour is not apparent.

Finally, there is, as expected, a clear correlation between the number of terms and the number of links. The latter is substantially larger than the former and increases in parallel. The average link weight drops as a result of the large increase in their number.

5.3.3 Task β: a new topic of interest emerges

In the β task our focus shifts from variations in a virtual user's short-term needs to a radical change, the emergence of a new topic of interest. We test the ability of profiles to respond to the introduction of documents about a new topic in the feedback stream (section 5.3.1). In other words, we test

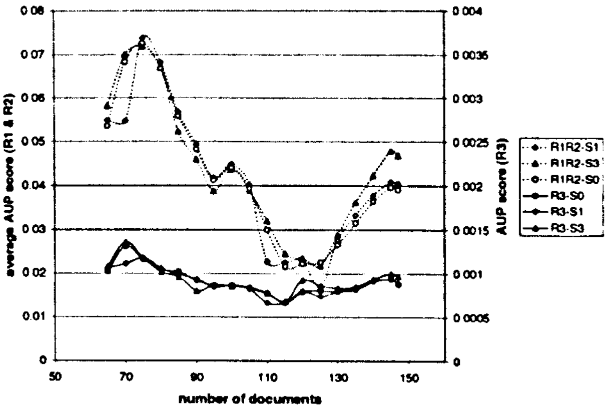

Figure 5-19: Score fluctuation for task β.1

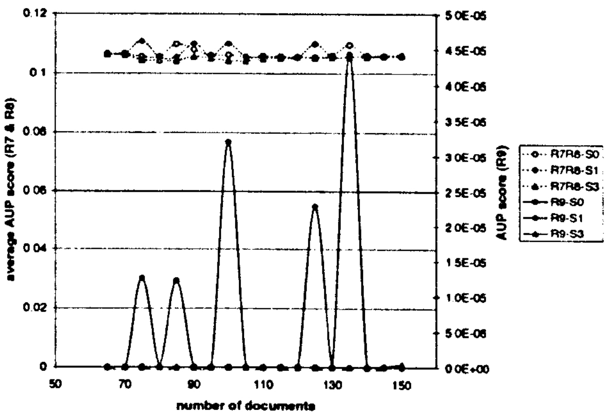

Figure 5-20: Score fluctuation for task β.2

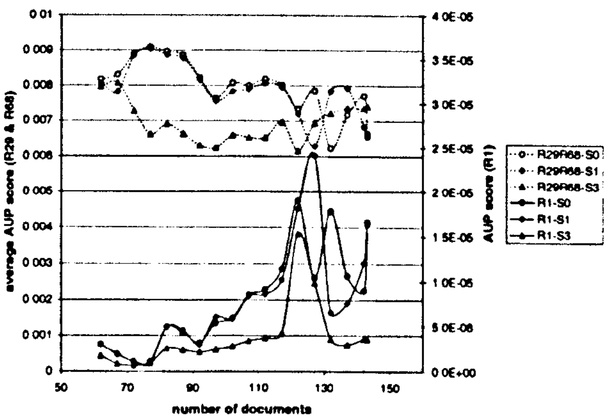

Figure 5-21: Score fluctuation for task β.3

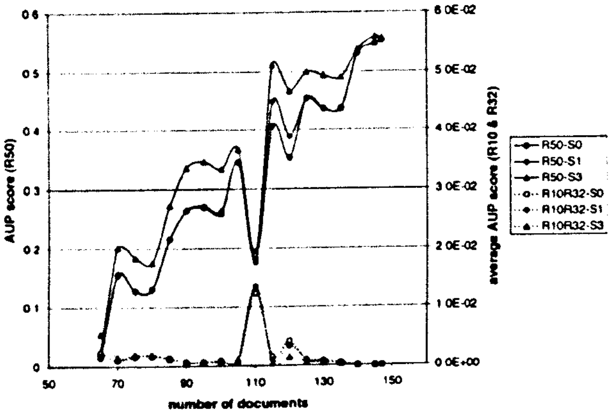

Figure 5-22: Score fluctuation for task β.4

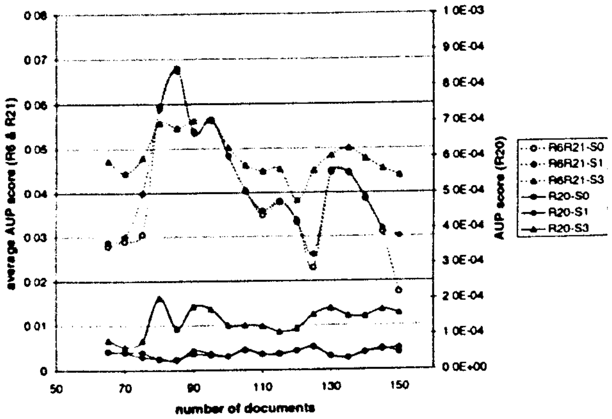

Figure 5-23: Score fluctuation for task β.5

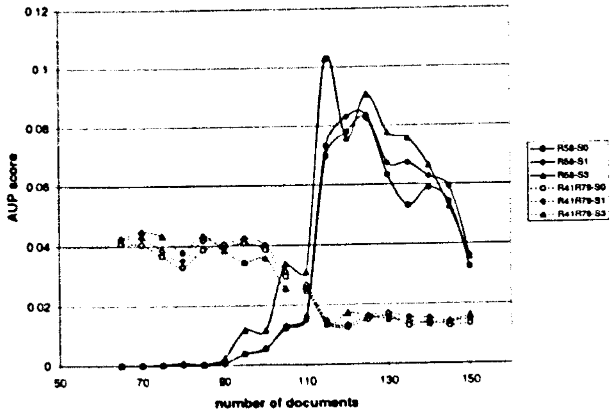

Figure 5-24: Score fluctuation for task β.6

the ability of profiles to learn a new topic of interest. Figures 5-19 to 5-24 present for each β task, the average AUP score for the initial two topics with dashed lines and the AUP score of the new third topic with solid lines. Separate lines are drawn for each document evaluation approach. Whenever necessary a secondary y-axis was used to account for the difference between the average for the initial two topics and the third topic's score. We have chosen to present the average score of the first two topics for visualisation reasons and also to be able to concentrate on the new topic that has to be learned. But as a result, short-term fluctuations in their individual scores are hidden. Hence, in this and the subsequent tasks, we don't include graphs showing the distribution of training documents per topic during the training period.

With the exception of task β.2, the rest of the β tasks produced encouraging results. For tasks β.1 and β.5 the results do not indicate a clear progressive increase in score. It appears, that in both cases the profile already contains terms related to the new topic, due to the semantic proximity between the latter and the initial two topics (table 4.8). They already represent aspects of the topic to be learned. As a consequence, the results for these two tasks show that it is difficult for related topics to distinguish themselves from other topics in the profile. This finding is exaggerated by the large number of test documents per topic and the fixed number of evaluation documents. Thus, in the extreme case of task β.2, the score for the new topic (R9) is very small. Note that, in contrast to the rest of the tasks, R9 corresponds to a relatively small number of test documents in relation to R7 and R8. The best 3000 documents can be more easily dominated by documents about these last two topics.

Nevertheless, for tasks β.3, β.4 and β.6, which include more unrelated topics, the results reveal the profile's ability to learn a new topic of interest. In all three cases, there is a clear increase in the third topic's score combined with periodic fluctuations in score, especially towards the end of the training phase. Usually, the average score for the persistent two topics reflect these fluctuations. The results thus show, to some extent, the combined effect of both tuning and alteration. It appears that a new hierarchy grows to represent the new topic, and, once developed enough, it starts competing with existing hierarchies. During this process, the overall drop in the average score for the persistent two topics is small. What is already represented is not forgotten. We should note again, that such a decrease in score is also exaggerated by the fact that evaluation is based on a constant number (3000) of documents, while the number of relevant documents in the test set increases with the addition of the third topic of interest.

The β tasks support the emerging case for the superiority of S3 which is, for the last three tasks, the best approach in both learning the new topic and not forgetting the existing two. These are the tasks that comprise topics with a small number of test documents. It is however the worst approach in task β.3. S0 and S1 behave in almost identical ways, with only a slight advantage for S1 in many cases.

Once more, we complement the above evaluation with macroscopic profile statistics. For each separate task, figures 5-25 to 5-30 show, in one graph, the number of profile terms and the average term weight, and in a second, the number of links and the average link weight.

The statistics' trends are clearly different from those in the α tasks, but a common pattern can be again identified. For most tasks, the number of terms remains initially almost constant and then it drops suddenly. Hundreds of terms run out of weight and are removed from the profile, before the number

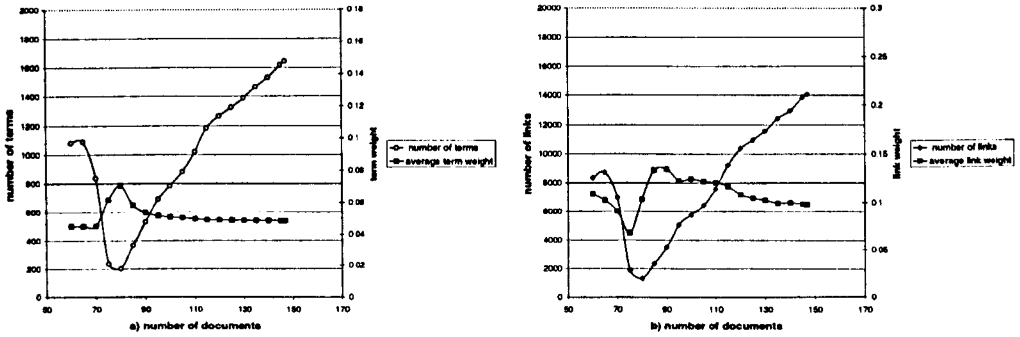

Figure 5-25: Profile statistics for task β.1: a) number of terms and average term weight b) number of links and average link weight

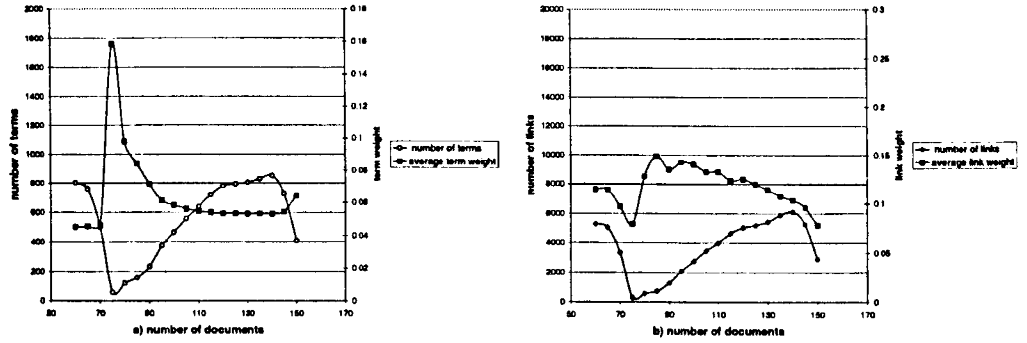

Figure 5-26: Profile statistics for task β.2: a) number of terms and average term weight b) number of links and average link weight

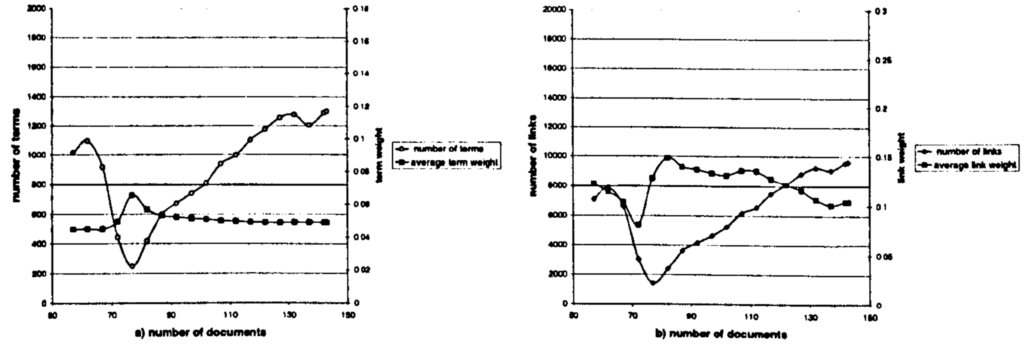

Figure 5-27: Profile statistics for task β.3: a) number of terms and average term weight b) number of links and average link weight

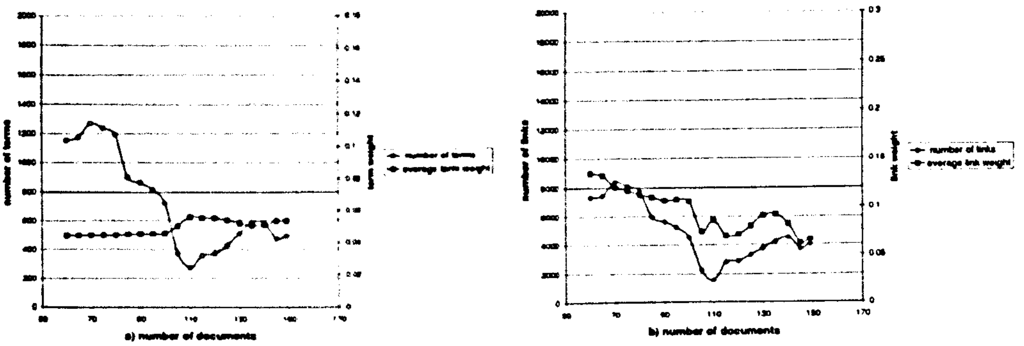

Figure 5-28: Profile statistics for task β.4: a) number of terms and average term weight b) number of links and average link weight

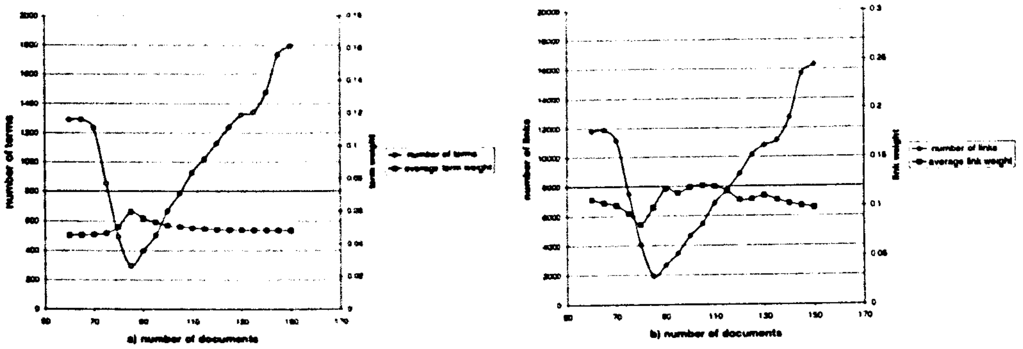

Figure 5-29: Profile statistics for task β.5: a) number of terms and average term weight b) number of links and average link weight

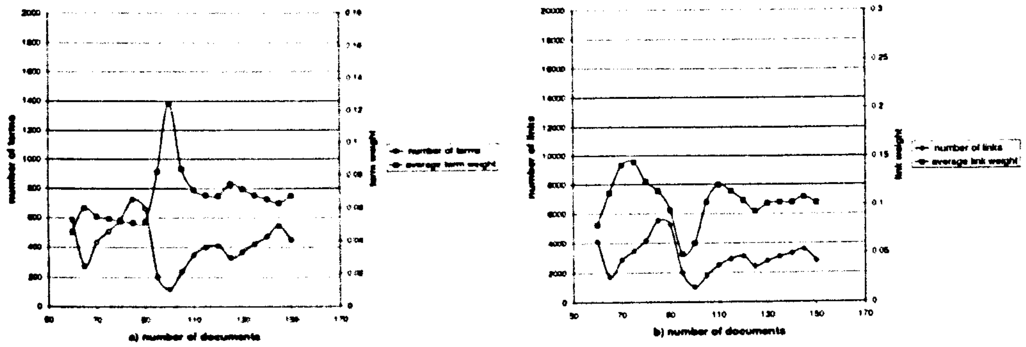

Figure 5-30: Profile statistics for task β.6: a) number of terms and average term weight b) number of links and average link weight

of terms starts increasing again. This pattern is more clear for tasks β.1, β.2, β.3, and β.5, where after the initial decrease the number of profile terms keeps increasing and overcomes the initial value. For task β.2 in particular, there is a further drop in the number of terms towards the end of the training phase. This is its only evident difference to the rest of the tasks in this group, but it does not explain the bad results. In the case of task α.4 and α.6, which produced the most positive results, the number of terms changes in a somewhat different way. In task β.4 the number of terms does not escalate after the initial drop. It only increases slightly and then drops again. More intense fluctuations are shown for task α.6. Here the number of terms fluctuates roughly around 400 terms. Maybe, a sufficient vocabulary of terms has been assembled in both profiles and so the number of terms does not increase further.

For all tasks, changes in the number of terms are coupled with changes in the average term weight. When the number of profile terms drops, their average weight increases suddenly, and then, as the number of profile terms increases again, their average weight drops slowly and stabilises at a value larger than the initial average. This behaviour is smoother for tasks β.1, β.2, β.3, and β.5, but is also apparent for tasks β.4 and β.6. One possible explanation is that, initially, new informative terms are extracted from documents about the new topic of interest. They are subsequently reinforced causing a large number of less informative terms low in the hierarchy to run out of weight and be removed. Eventually the profile acquires a sufficient vocabulary of informative terms about the emerging topic. The corresponding hierarchy grows and less informative terms start entering the profile anew, thus increasing the number of terms. Whatever the exact process, it is important to note that the average weight appears to stabilise at values larger than the initial average. To represent three topics of interest the profiles needs more informative terms, or in other words, it needs to store more information. The overall increase in weight is more evident for the most successful tasks, β.4 and β.6, but is also relatively large for the least successful task β.2.

Similar observations can be made about the number of links and their average weight. The number of links follows the distribution of the number of terms. Their average weight initially drops, because links are not removed according to their weight, but only when the corresponding terms are removed. Subsequently, as with the average term weight, the average link weight increases quickly and then drops progressively.

5.3.4 Task γ: forgetting a topic

In task γ, we test the ability of profiles to forget one of three topics of interest. For each γ task, a profile is initially trained with documents about three topics and subsequently with documents about only two of the topics (section 5.3.1). As before, figures 5-31 to 5-36 present for each γ task and document evaluation function, the average AUP score for the two topics of consistent interest with dashed lines and the AUP score of the third, unexcited topic, with solid lines. A secondary y-axis was again used whenever necessary.

As with task β.2, the results for task γ.2 do not show any significant differences in the profile's performance. The topic to be forgotten was not effectively learned in the first place. Its initial score is very low. However, although not observable in the figure, initial scores in the order of E-07 are followed by some sudden fluctuations for S0 and S1 and then topic R9 is completely forgotten (zero scores). In task γ.1, that, as for task γ.2, involves

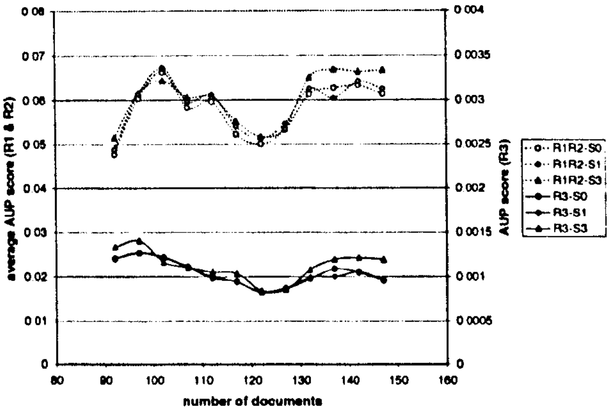

Figure 5-31: Score fluctuation for task γ.1

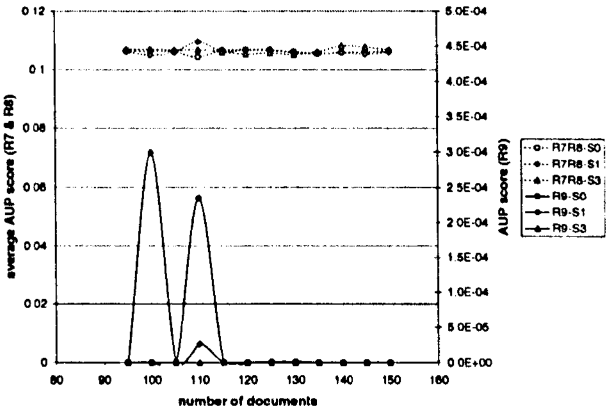

Figure 5-32: Score fluctuation for task γ.2

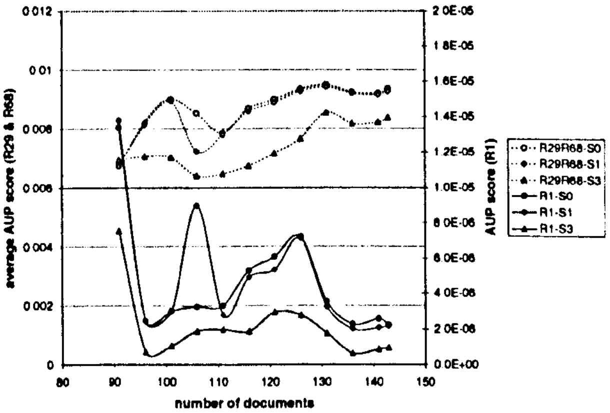

Figure 5-33: Score fluctuation for task γ.3

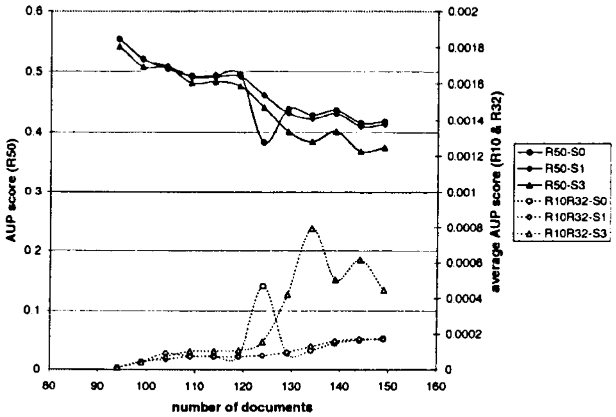

Figure 5-34: Score fluctuation for task γ.4

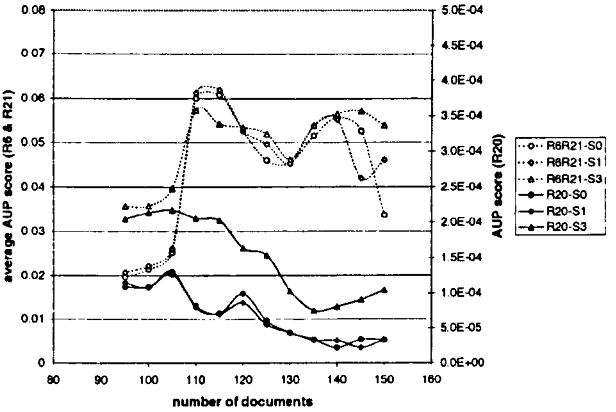

Figure 5-35: Score fluctuation for task γ.5

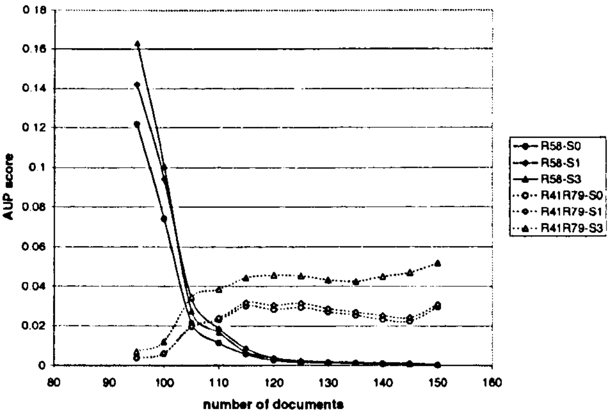

Figure 5-36: Score fluctuation for task γ.6

Table 5.2: Ratio of decrease in score to initial score

	γ tasks			δ tasks		
task	S0	S1	S3	S0	S1	S3
.1	0.203	0.201	0.106	-0.058	-0.079	0.0514
.2	1.0	1.0	1.0	1.00	1.00	1.00
.3	0.835	0.843	0.874	0.762	0.922	0.911
.4	0.245	0.254	0.307	1.0	1.0	0.997
.5	0.694	0.714	0.488	0.868	0.867	0.81
.6	0.997	0.997	0.997	1.000	1.000	1.00

Table 5.3: Overall decrease in score

	γ tasks			δ tasks		
task	S0	S1	S3	S0	S1	S3
.1	2.4e-04	2.4e-04	1.4e-04	-6e-05	-8e-05	6e-05
.2	4.1e-07	1.4e-07	1.5e-07	4.1e-07	1.4e-07	1.5e-07
.3	1.1e-05	1.2e-05	6.6e-06	1.2e-05	1.4e-05	8.1e-06
.4	0.1357	0.1408	0.16654	0.55836	0.55755	0.53615
.5	7.6e-05	8.2e-05	1e-04	9.3e-05	9.3e-05	1.6e-04
.6	0.12166	0.14166	0.16272	0.13401	0.13401	0.14517

related topics, there is no progressive decrease in the score of the topic (R3) to be forgotten. Terms maintained in the profile as part of the representation of the two persistent topics may reflect concepts discussed in documents about the third related topic. This hypothesis is supported by the fact that the score for topic R3 roughly reflects the average score for the two persistent topics (R1 and R2). The drop in the third topic's score is also not progressive in task γ.3. After a significant relative drop, the score fluctuates, especially for S0 and S1, to eventually reach a low value.

However, a progressive drop in the score of the topic to be forgotten is evident in the three last γ tasks, which comprise topics with a small number of test documents. Table 5.2 presents for each γ and δ task (for comparisons made in the next section) the ratio of the decrease in score to the initial score of the topic to be forgottern (i.e. (initial score - final score)/initial score). For the same tasks, table 5.3 summarises the overall decrease in score (i.e. initial score - final score). The score drops in all tasks, but is more significant, in terms of ratio for tasks γ.3, γ.5 and γ.6 (the overall decrease for topic R9 in task γ.2 is insignificant) and in terms of overall decrease for tasks γ.4 and γ.6 (tasks which comprise unrelated topics). However, only in task γ.6, does the score drop to zero values, following roughly a power law distribution. More feedback cycles are possibly required for other tasks. Nevertheless, we can still argue that the results indicate that, following the withdraw of documents about one of the initial three topics from the feedback stream, the profile's performance for that topic drops. Usually, the drop is coupled with an increase in the average score for the persistent two topics, but as already mentioned this increase is exaggerated by the fixed number of evaluation documents.

The results also show (table 5.3), that the overall decrease in score is, for the last three tasks, larger in the case of the hierarchical profile using S3. With the exception of task γ.3, the same approach exhibits the best average score for the persistent two topics in each task. S3 is not only good at forgetting the unexcited topic, but also in representing the still interesting ones. S1 is again at least as good as S0.

As already done in the previous tasks, figures 5-37 to 5-36 show for each task, the number of profile terms, the average term weight, the number of links and the average link weight. Here, two different kinds of behaviour can be identified. In the case of tasks γ.1, γ.3 and γ.5, the number of terms starts at values larger than 1000, it then decreases rapidly by around 1000 terms and finally, it starts increasing progressively, to reach values smaller than the initial number of terms. The change in the number of profile terms is coupled with a sudden increase in the average term weight, which follows an initial

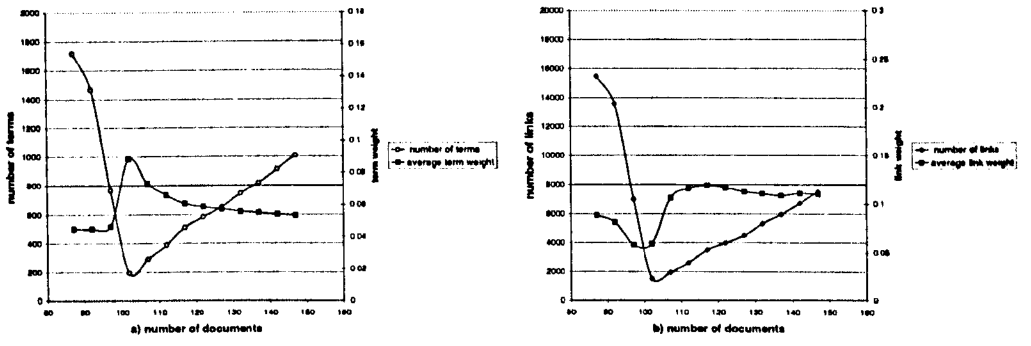

Figure 5-37: Profile statistics for task γ.1: a) number of terms and average term weight b) number of links and average link weight

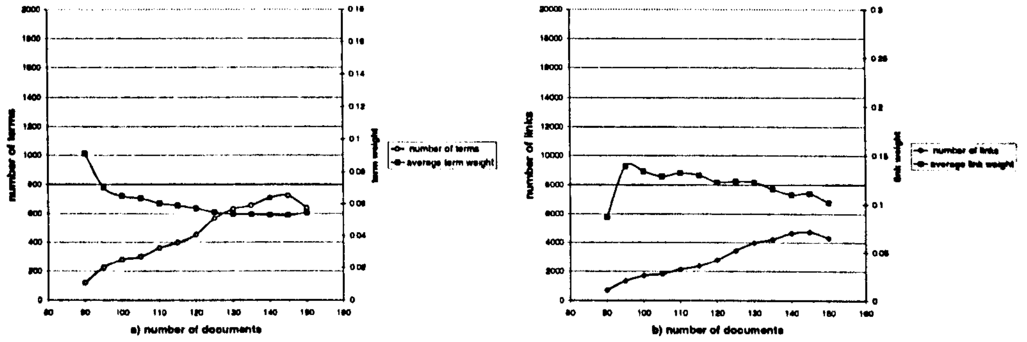

Figure 5-38: Profile statistics for task γ.2: a) number of terms and average term weight b) number of links and average link weight

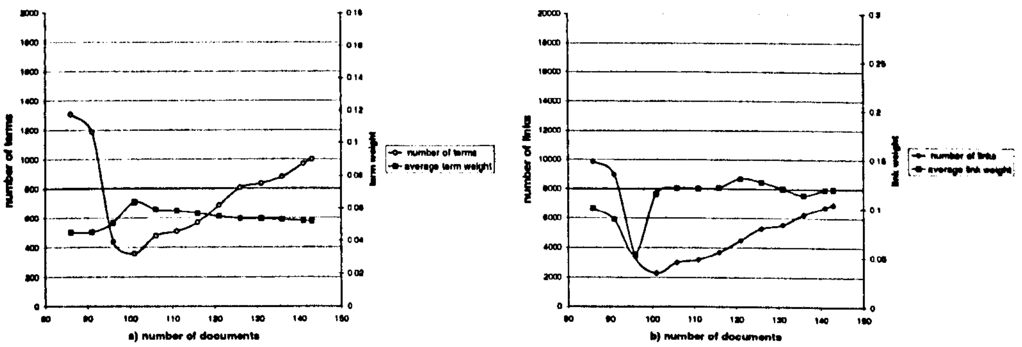

Figure 5-39: Profile statistics for task γ.3: a) number of terms and average term weight b) number of links and average link weight

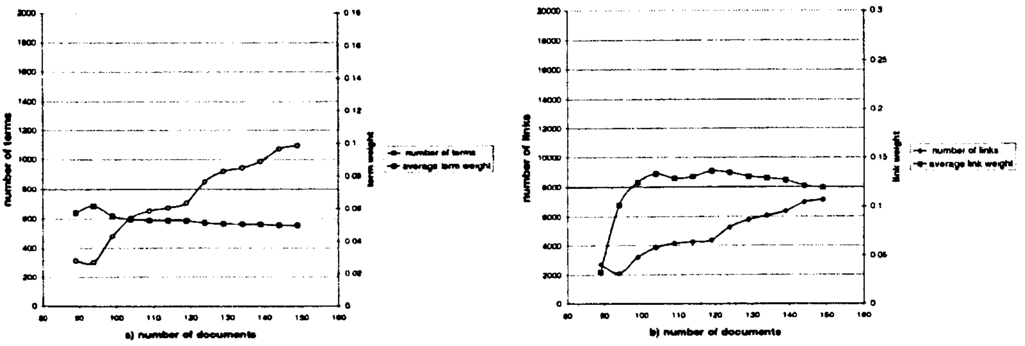

Figure 5-40: Profile statistics for task γ.4: a) number of terms and average term weight b) number of links and average link weight

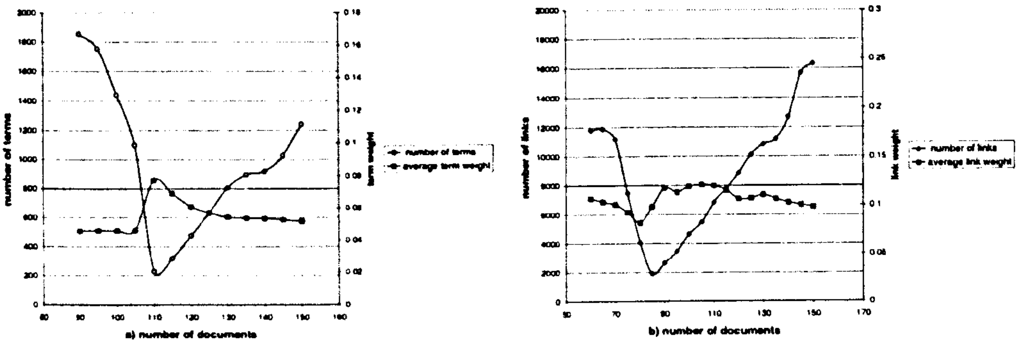

Figure 5-41: Profile statistics for task γ.5: a) number of terms and average term weight b) number of links and average link weight

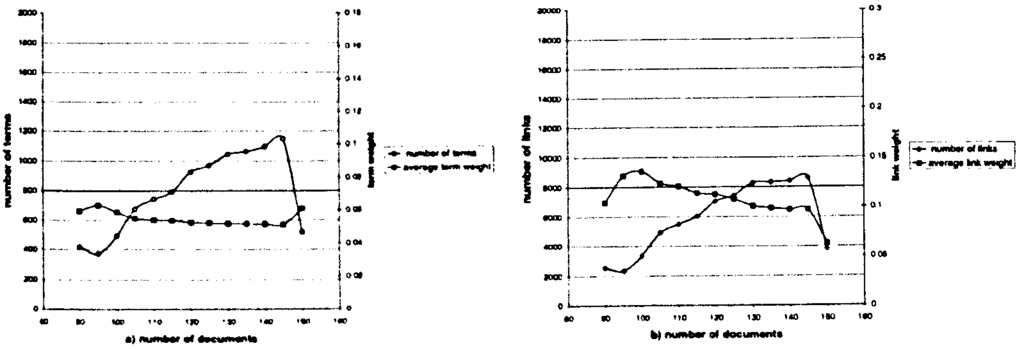

Figure 5-42: Profile statistics for task γ.6: a) number of terms and average term weight b) number of links and average link weight

period of relatively constant weight. Subsequently, the average term weight drops slowly, but remains over the initial value at the end of the training period. Similarly, the number of links follows the changes in the number of terms and their average weight initially drops, then increases suddenly and finally starts decreasing progressively.

In contrast, for tasks $\gamma.2$, $\gamma.4$ and $\gamma.6$, the number of terms is initially much smaller, less than 500. A small initial decrease, in the case of the last two tasks ($\gamma.4$ and $\gamma.6$), is followed by a progressive increase towards values close to 1000, common to all tasks. In $\gamma.6$ in particular, there is also a significant decrease at the end of the training period. At the same time, the average term weight progressively decreases. Similar observations can be made for the links and their average term weight.

The difference in the initial number of profile terms (number of profile terms after the first training period) might justify the two different types of behaviour. In the first case, the profile includes many terms and so after the radical change of interest more terms are removed than are added. The opposite takes place in the latter type of behaviour. Therefore, how the profile responds to changes in the feedback stream depends on its current state.

We expected a decrease in the average term weight because fewer topics have to be represented now, but only for tasks $\gamma.2$ and $\gamma.6$ is an overall decrease evident. In task $\gamma.4$, the general decrease in the average term weight is followed by a sudden increase and so the final value is larger than the initial. For tasks $\gamma.1$, $\gamma.3$ and $\gamma.5$, further feedback cycles maybe required for an overall decrease in average term weight to take place. This possibility is supported by the fact that for this task the profiles are initially trained with 90 documents and subsequently only with 60.

5.3.5 Task δ: forgetting a topic with negative feedback

The difference between γ and δ tasks is that in the latter case the user explicitly indicates that the third topic is non-relevant through negative feedback. The question here is: does such non-relevance information boost the forgetting process? Figures 5-43 to 5-48 present the results for δ tasks. The average AUP score for the two persistent topics of interest in each task are presented with a dashed line and the AUP score for the non-relevant topic with a solid line.

The results for the δ tasks are similar to those for γ tasks. For task δ.1 the score for the third unexcited topic initially decreases and then increases back to values close to the initial. Once more, the results for task δ.2 do not clearly indicate changes in the profile's performance in response to the radical change of interest. In task δ.3, the score for the non-relevant topic initially drops radically and then fluctuates, especially for S0 and S1, before it reaches a final low score.

Once more, a progressive decrease in the score of the non-relevant topic is apparent in the last three tasks, that comprise topics with a small number of test documents. According to table 5.2 the ratio of the decrease in score to the initial score is larger for tasks δ.3 (S1 & S3), δ.4, δ.5 and δ.6 (marginally) than the corresponding γ tasks. Similarly table 5.3 indicates that the overall decrease is more significant for tasks δ.4 and δ.5 than for the corresponding γ tasks. Although less significant, a similar difference exists between task δ.3 and task γ.3. As in task γ.6, in task δ.6 the score for the non-relevant topic drops to zero values, but clearly the decrease is now faster. Therefore, in most cases, the results indicate that negative feedback intensifies the forgetting process.

Furthermore, the results support the superiority of S3 both in forgetting

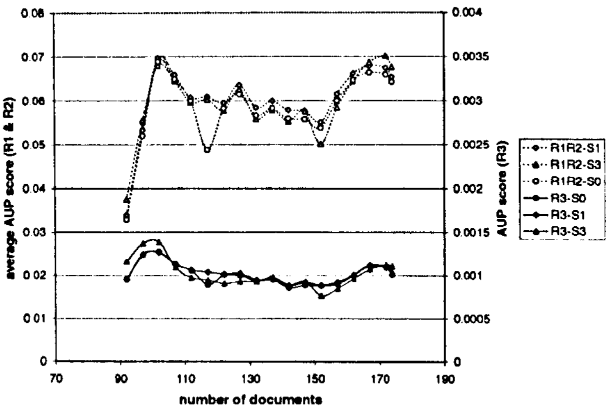

Figure 5-43: Score fluctuation for task δ.1

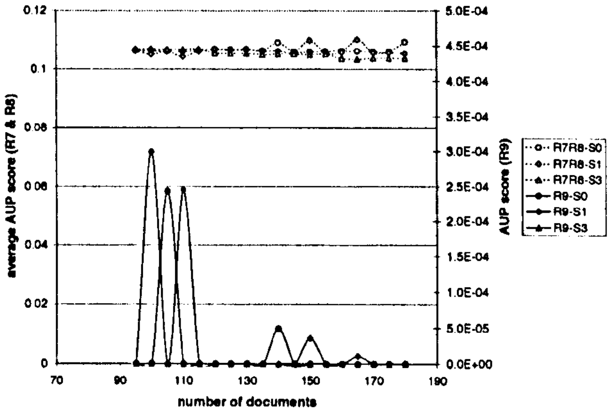

Figure 5-44: Score fluctuation for task δ.2

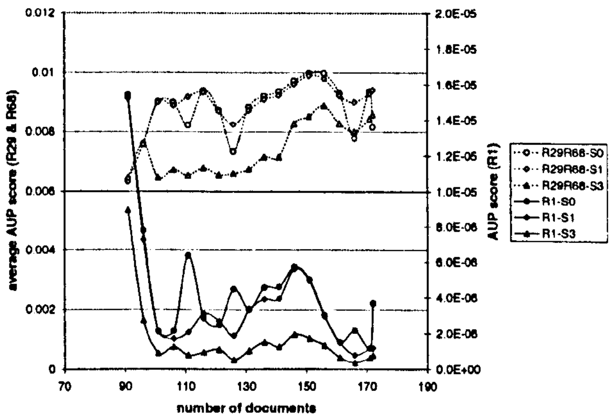

Figure 5-45: Score fluctuation for task δ.3

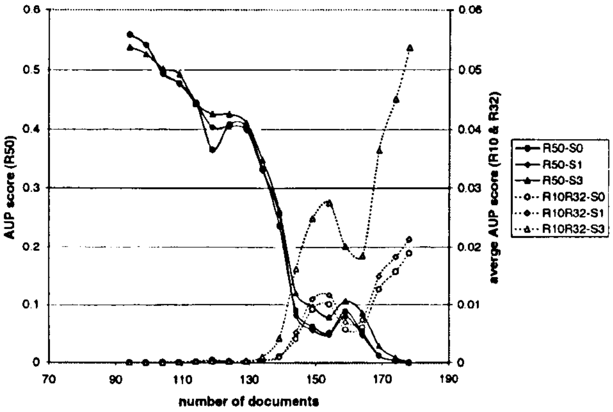

Figure 5-46: Score fluctuation for task δ.4

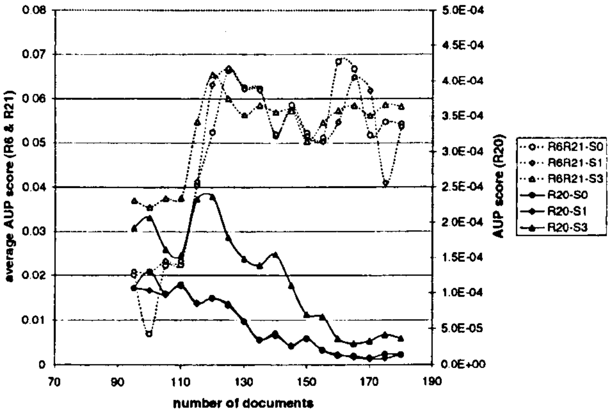

Figure 5-47: Score fluctuation for task δ.5

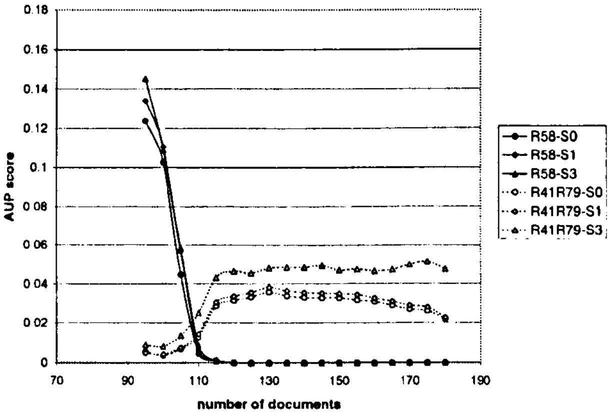

Figure 5-48: Score fluctuation for task δ.6

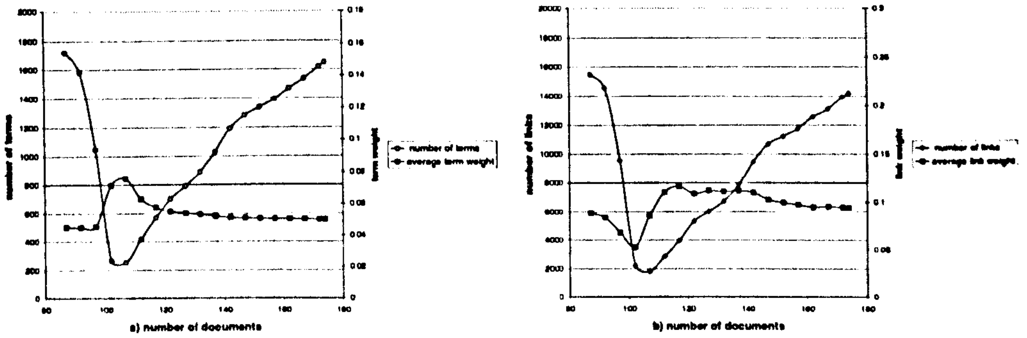

Figure 5-49: Profile statistics for task $\delta.1$: a) number of terms and average term weight b) number of links and average link weight

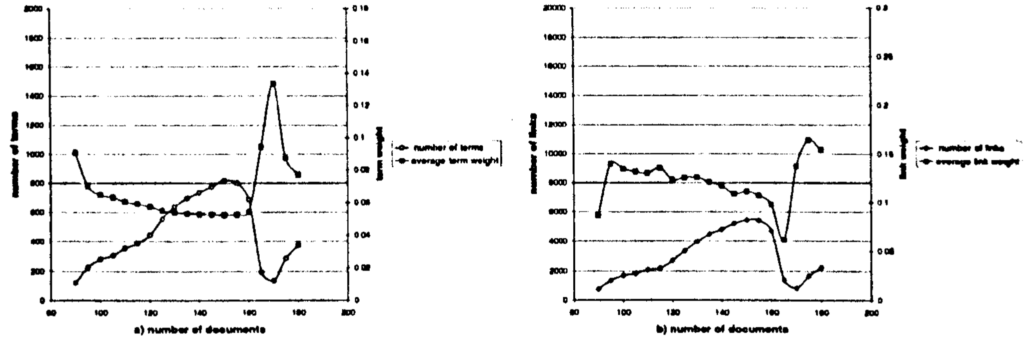

Figure 5-50: Profile statistics for task $\delta.2$: a) number of terms and average term weight b) number of links and average link weight

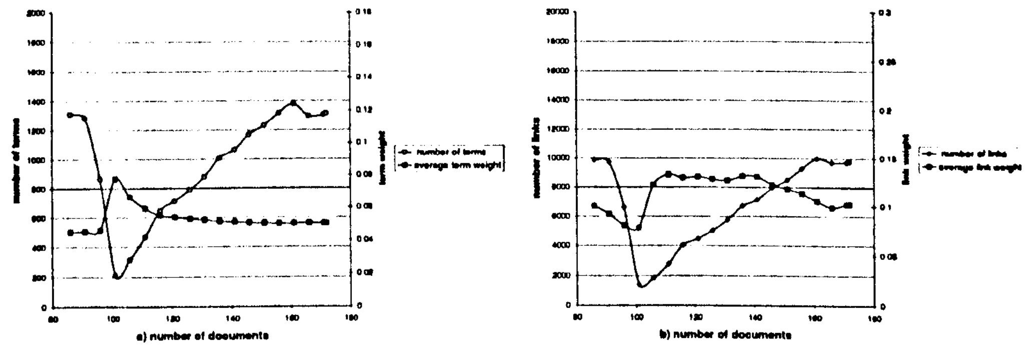

Figure 5-51: Profile statistics for task $\delta.3$: a) number of terms and average term weight b) number of links and average link weight

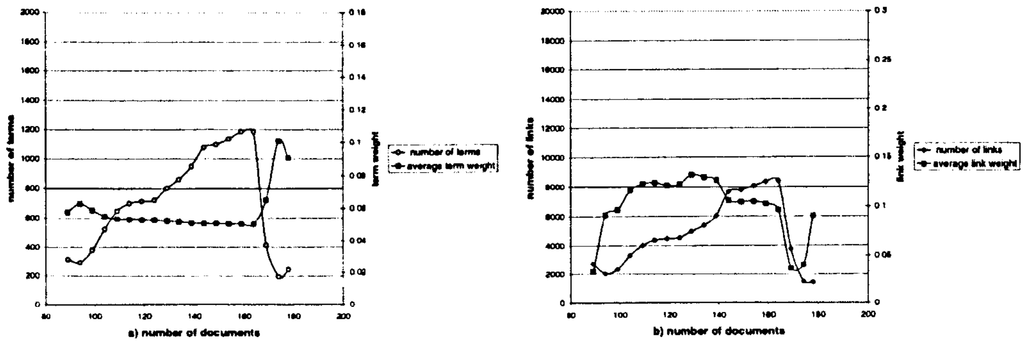

Figure 5-52: Profile statistics for task δ.4: a) number of terms and average term weight b) number of links and average link weight

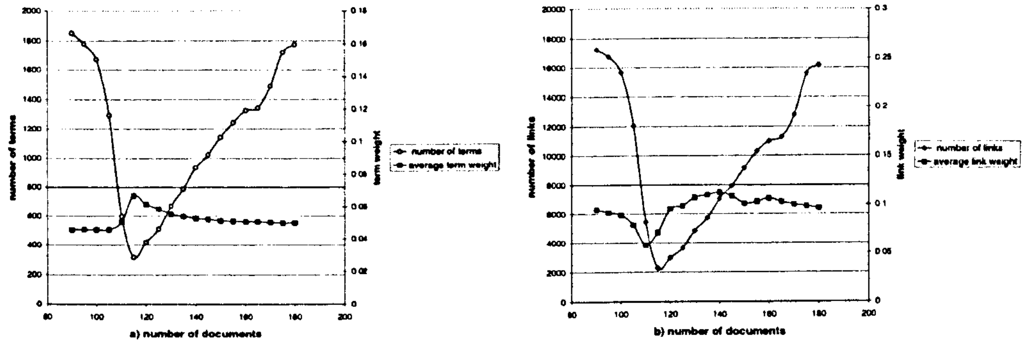

Figure 5-53: Profile statistics for task δ.5: a) number of terms and average term weight b) number of links and average link weight

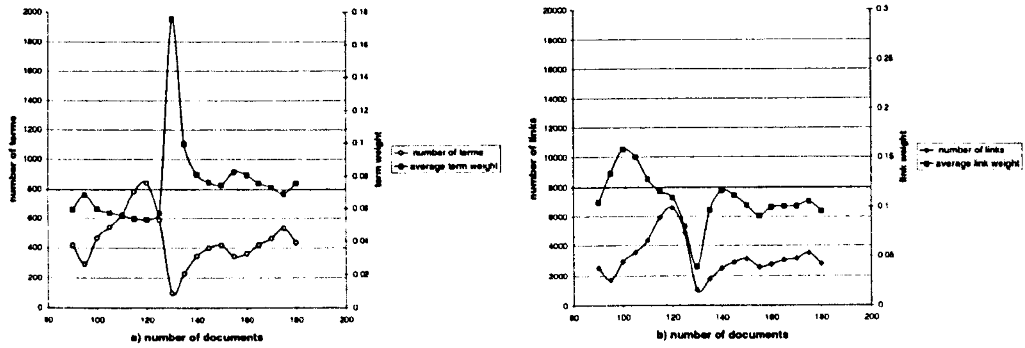

Figure 5-54: Profile statistics for task δ.6: a) number of terms and average term weight b) number of links and average link weight

the non-relevant topic and especially in effectively representing the persistent ones. S1 is at least as good as S0 and in some cases (tasks δ.4 and δ.6) slightly better.

Figures 5-49 to 5-54 show for each task, the number of profile terms, the average term weight, the number of links and their average weight. Similar to task γ, two kinds of behaviour in response to negative feedback can be identified. In tasks δ.1, δ.3 and δ.5 the initial number of terms is large (above 1000 terms) and subsequently drops quickly to around 200 terms. Then it starts increasing progressively to reach values close to the initial level. At the same time, the average term weight initially increases quickly and then decreases progressively and appears to stabilise at values larger than the initial. Similar observations can be made for the number of links and their average weight.

On the other hand, for tasks δ.2, δ.4 and δ.6 the initial number of profile terms is small (smaller than 400 terms) and after a small decrease in tasks δ.4 and δ.6, it increases progressively. Then it drops suddenly to values close to, or smaller than, the initial value. Finally, it increases again, especially in tasks δ.2 and δ.6. While the number of terms increases their average weight drops. Following the sudden decrease in the number of terms, the average term weight increases suddenly and then drops again to values that, for tasks δ.4 and δ.6, exceed the initial value. The number of links follows the changes in the number of terms and the average link weight varies accordingly.

We again attribute these two types of behaviour to differences in the number of terms. How the profile responds depends on its current state. It is however important to note that, here, the results indicate that in most cases the average term weight has an overall increase. Less topics have to be represented, but it appears, that the profile takes into account information provided through negative feedback. Terms, including non-informative, that appear in the non-relevant documents are kept out of the profile and hence the average term weight increases.

5.3.6 Discussion of the Results

We may argue that the experimental results have been positive. They indicate that the profile responds to short-term variations and occasional radical changes in the composition of a stream of feedback documents. As a result the following adaptive behaviours are observed:

1. More than one topic of interest may be learned from scratch and in parallel with a single profile (task α).

2. The relative importance of topics in the profile varies in response to short-term variations in the distribution of relevant documents in the training set (task α).

3. An existing profile representing more than one topic of interest may learn an emerging topic of interest, without what is already represented being significantly affected (task β).

4. A profile representing more than one topic of interest may forget a topic that, in contrast to the rest of the topics, no longer receives positive feedback (task γ). No explicit negative feedback is required.

5. A profile representing more than one topic of interest, forgets a topic faster and more effectively if it receives negative feedback (task δ).

The above findings do not account for a fixed behavioural repertoire, but rather the profile's ability to respond through self-organisation to a variety of changes in a stream of feedback documents. Macroscopic statistics, show that the profile responds structurally through tuning and alteration. The weight of existing terms and links is updated, new terms and links are generated, and incompetent ones are removed. How the profile reacts depends on its current structure. We may therefore argue, that new structures and new modes of behaviour, new modes of document evaluation in our case, are created in the self-organisation process. The introduced process exhibits the third characteristic of self-organisation.

The results have been more clear for tasks comprising unrelated topics with a small number of test documents. There are no significant changes in score, when the topic to be learned or forgotten is related to the persistent two topics, due to the common vocabulary that this implies. The results are also less clear for tasks that comprise topics with a large number of test documents, due to the fixed number (3000) of evaluation documents used.

Furthermore, with the exception of task .3 in each case, the hierarchical profile using S3 is the best performing approach. We should remember that S3 is the approach most sensitive to the hierarchical structure, which it takes into account explicitly through breadth b and size d (equation 4.5). The results highlight the importance of the profile's hierarchical structure in determining and adapting the document evaluation process. The positive effect of links is also reflected by the fact that the hierarchical profile with S1 is at least as good as the unconnected profile using S0.

5.4 Summary and Conclusions

Combinations of linear profile representations with linear learning algorithms are forced to break down the problem of profile adaptation into different single-topic profiles and separate adaptation levels. Therefore, they cannot fully account for the dynamic nature of user interests. Genetic Algorithms represent a more dynamic, essentially probabilistic, approach to profile adaptation that employs a population of single-topic representations and is computationally expensive.

To achieve adaptation of our single, multi-topic profile to a variety of changes in the user interests, we have been inspired by biological theories of self-organisation and by autopoiesis in particular. In the previous chapter we complied with the first characteristic of self-organisation, non-linearity. Here we introduced a process comprising five deterministic, but interrelated, steps that bring plasticity to the profile's hierarchical structure. The profile becomes open to its environment with the addition and removal of terms. The result is a profile that operates far from equilibrium, constantly changing structurally in response to changes in a stream of feedback documents. Existing components are calibrated, new structures grow and existing structures disintegrate. How may the profile evaluate document changes accordingly.

To test our approach to adaptive document filtering, we have synthesised virtual users based on RCV1's classification schema and established a methodology based on a variation of TREC's routing subtask. We made the assumption that a user's interests and changes in them are reflected by the feedback that the user provides. On these grounds we evaluated both hierarchical and unconnected profiles against two learning and two forgetting tasks. The results have been positive. They indicate the profile's ability to respond to frequent short-term variations in the feedback stream and occasional radical changes. Furthermore, they highlight the importance of taking into account the profile's hierarchical structure and confirm the significance of term dependence representation.

In conclusion, if our assumption is true, then we may argue that adaptation to both variations in a user's short-term needs and radical changes in long-term interests has been achieved with a single, multi-topic profile, through a process that exhibits all three characteristics of self-organisation (section 5.1).

Chapter 6

Conclusions and Future Research

This thesis started with a discussion of the practical and scientific issues that motivated our PhD research on personalised information delivery and more specifically adaptive filtering of textual information. In this final chapter, we summarise our approach and results, we draw conclusions, and we carve out our future research directions.

6.1 Linearity in Information Filtering

The problem of information overload is not only present but is here to stay. Scientific and technological innovations consistently contribute to the accessibility and exponential increase of online information. The need to make use of this information glut is pressing. Personalised Information Delivery (PID) is an important aspect of this trend. It can be modelled as a series of focusing processes which aim at providing an individual with the information that is most likely to be relevant to the individual's interests.

In the case of textual information, IR, TC and more recently IF, are the main research fields that tackle various aspects of PID. Traditionally, IF has been viewed as a specialisation of the former, more well established, disciplines. However, the long-term nature of user interests differentiates IF from IR and TC. On one hand, a user may be interested in more than one topic in parallel. On the other hand, the user's multiple interests change dynamically over time.

Despite the above characteristics of a user's long-term interests, our exploration of the state-of-the-art revealed that, typically, IF research inherits the dominant term independence assumption. The result is profile representations that support linear document evaluation functions and hence, can only effectively represent a single topic of interest. A separate profile is built for each one of a user's multiple topics of interest. So far, a single, multi-topic profile has not been proposed.

Furthermore, profile adaptation of single-topic profiles was sought using linear learning algorithms. Although it was realised that the steady adaptation pace that these algorithms achieve cannot account for the dynamic nature of changes in the user interests, the proposed alternative was again to break down the problem into discrete adaptation levels with different learning coefficients. Profile adaptation using GAs or MAs represents a different more dynamic approach, but suffers from computational cost and the linearity of the single-topic profiles that are employed. The adaptation of a single, multi-topic profile to dynamic changes in the user interests has not yet been addressed.

Our contributions to adaptive document filtering derive from a novel approach towards profile representation and adaptation, that was founded on non-linearity and self-organisation. In the following sections, we describe in detail how we have achieved term dependence representation, multi-topic information filtering with a single profile, profile adaptation through self-organisation and other novel aspects of our work.

6.1.1 Term Dependence Representation

The term independence assumption has always been recognised as wrong and some efforts have indeed been made to incorporate term dependencies into content representation structures. Connectionist approaches are a natural route towards this end. Associative graphs that express the stochastic dependencies between terms have been suggested and applied mainly for query expansion. More recently, concept hierarchies that represent topic-subtopic relations between terms have also been proposed. Nevertheless, no existing content representation has tackled all three dependence dimensions that Doyle identified in 1961 [48].

In chapter 3, we presented a methodology that, through a series of three processes, generates out of a set of user specified documents, a hierarchical term network that takes into account topical and lexical correlations between terms and distinguishes topic-subtopic relations between them. All three dependence dimensions are tackled.

The first of these processes tackles documentation redundancy through stemming, stop word removal and then the weighting and selection of the most informative terms in the specified documents. It was realised that according to the traditional view of IF, the common practice is to adopt an existing term weighting method based on its successful application in the context of IR or TC. However, we argued that the characteristics of the relevance information that is available for profile construction may affect the effectiveness of existing term weighting methods. Therefore, we introduced a term weighting method called Relative Document Frequency (RelDF), that was devised with these characteristics in mind. We have then evaluated it in comparison to a large number of existing methods, using a methodology that reflects the characteristics of IF. The results indicated that term weighting methods for IF should take into account, and achieve a balance between, both the relevance information that a user provides and information derived from a general collection. Favoured by the characteristics of the experimental setup, IG was the best performing approach followed by RelDF, which appears to be a promising and flexible alternative. On these grounds, we have concentrated on these two methods for the realisation of the first process in the methodology.

The second process involved the identification and weighting of topical and lexical correlations between the extracted terms. For this purpose, we used a sliding window of 10 terms and introduced a novel link weighting method that combines the statistical dependencies caused by both lexical and topical correlations into a single measure. The result of this second step is an associative graph that expresses term dependencies with symmetric weighted links between terms.

In the third and final step, we identified topic-subtopic relations between terms by ordering the terms according to relevant document frequency, or even better, their assigned weights. Therefore, the hierarchical term network that this process generates has the same applicability as existing concept hierarchies, with the additional advantage that both lexical and topical correlations are taken into account. Furthermore, it may represent and distinguish between more than one topic of interest. The hierarchical network's topology reflects the topics discussed in the user specified documents and provides evidence for their identification.

6.1.2 Multi-Topic Document Filtering with a Single Profile

To achieve multi-topic document filtering with a single profile we have introduced in chapter 4 a layered and a continuous approach for establishing non-linear document evaluation functions on the hierarchical term network. The latter can hence be considered a single, multi-topic user profile. Both approaches resided on similar directed, spreading activation models which allowed the dependencies and topic-subtopic relations between terms to be taken into account.

In the first case however, the partitioning of the hierarchy into discrete layers had a negative effect on filtering performance, which comparative experiments revealed. The continuous approach overcomes the layered approach's drawbacks. Using a slightly different spreading activation model, we have introduced a series of three non-linear document evaluation functions that in addition to the dependencies and topic-subtopic relations between terms, exploit additional evidence derived from the topology of the subhierarchies that a document activates. Initial single-topic experiments indicated that this continuous approach is indeed superior to the layered approach. But, possibly due to the characteristics of test set and the small number of training documents used, the continuous approach does not clearly outperform an unconnected profile representation that evaluates documents using the linear, inner product. This is especially true for large numbers of extracted terms and consequently of links.

Nevertheless, experiments on multi-topic document filtering with a single profile, conducted using combinations of two and three topics, produced promising results. Hierarchical profiles perform on average better for most topics and their combinations. For the two-topic experiments in particular, it was observed that the hierarchical profile performs particularly well for combinations that produce profiles with large average link weight. On the other hand, it was again observed that the difference between hierarchical and unconnected profiles is smaller when a large number of terms, and hence of links, is extracted.

Our research on multi-topic document filtering with a single user profile highlights an unexplored niche that we hope will attract the attention of other researchers in the field. A whole new domain of non-linear document evaluation functions on the hierarchical profile representation can be envisioned. Further research in this direction may even challenge the dominance of the traditional vector space model in IF.

6.1.3 Profile Adaptation through Self-Organisation

To achieve adaptation of our single, multi-topic profile to changes in a user's interests, we have been inspired by biological theories of self-organisation. In chapter 5, we presented a process comprising five deterministic, but interrelated steps that collectively cause the profile's structural self-organisation in response to changes in a stream of feedback documents. We assumed, that such changes reflect changes in the user interests.

Experiments using virtual users produced positive results. Through self-organisation, the profile appears to adapt to a variety of changes ranging from frequent variations in a user's short-term needs, to occasional radical changes like the emergence of a new topic of interest and the loss of interest in a certain topic. The profile can learn interesting topics, or forget topics that are no longer interesting. In the latter case negative feedback is not required, but intensifies the process when available. Some less clear results were produced in cases where the topic to be learned or forgotten is related to the persistent topics of interest, but these results have been exaggerated due to the large number of test documents per topic and the fixed number of evaluation documents.

The proposed adaptation of a single, multi-topic profile through self-organisation represents a significant innovation over existing practices, which adapt single-topic profiles with a steady pace, or using discrete adaptation levels. A fixed learning coefficient is not employed. Complex adaptive behaviour has been achieved with a combination of deterministic processes governed essentially by a single parameter, the weight threshold used to extract informative terms from feedback documents.

Furthermore, our approach represents a more efficient alternative to the application of GAs and MAs for AIF. Global adaptation to radical changes in the user interests is now achieved with a single user profile that represents multiple user interests. There is no need for a population of individual profiles. Furthermore, the relative importance of topics and subtopics of interest is not reflected by an external indicator, like the fitness of individual profiles in a population, but is ingrained in the way they are being represented.

6.1.4 Experimental evaluation

The definition of a novel standard evaluation methodology has not been our major research goal. However, it was realised that existing evaluation standards were not well suited to the novelty of our approach. No existing methodology targets single, multi-topic profiles. The well established TREC filtering track is influenced by the traditional view of IF as a specialisation of IR or TC. In addition, its adaptive filtering subtask resides on loosely controlled changes in the content of documents over time. To test our innovative approach to adaptive document filtering we had to devise new evaluation methodologies.

To avoid reinventing the wheel, we conducted our experiments using variations of TREC's routing subtask. In chapter 3, only a small number of training documents per topic, that more accurately reflects the number of documents that a user is expected to specify for profile initialisation, was allowed to train profiles for our comparative evaluation of term weighting methods. In chapter 4, we proposed another variation that allowed the evaluation of single, multi-topic profiles. Finally, to test the adaptive mechanism (chapter 5), we synthesised virtual users out of the RCV1 topic categories and introduced a methodology for testing the ability of profiles to perform a number of learning and forgetting tasks.

Although the above variations allowed us to experiment using as much as possible an existing evaluation standard, the experimental results were unfortunately influenced by an acknowledged drawback of the adopted RCV1 corpus. That is, the large number of test documents per topic (see section 3.3.2). The results were usually better when topics with a small number of test documents were used. We expect improved results if a more semantically focused corpus is used.

Nevertheless, the proposed methodological variations are not constrained to RCV1. Any preclassified corpus could be used. Considering the removal of the filtering track from TREC 2003, due to the above drawback of RCV1 and other reasons, there is obviously a niche for a new approach to the evaluation of IF systems. Our novel experimental methodology provides suggestions towards this end.

6.2 Future Work

Two different, but compatible directions for future research can be followed. One explores the applicability of our innovative approach and the other its theoretical implications.

6.2.1 Towards Nootropia

We wish to further develop Nootropia into a complete system, into an intelligent information assistant with a broad application scope. Our PhD research has focused on Nootropia's adaptive, document filtering profile. Our experimental results have been encouraging, despite the fact that certain parameters have not been fine tuned. For example, the results indicate that further improvements can possibly be achieved by controlling the quantity and quality of generated links. This can be done by maintaining only those links with weight over a certain threshold. Experiments for fine tuning this threshold are required. Other system aspects that require further investigation include the threshold for selecting the most informative terms, the effect of using non-symmetric links and the effect of stop word removal and stemming on the filtering performance. Further experiments should of course overcome the disadvantages of the current experimental methodology.

Once a satisfactory performance is achieved we can then turn to other more interactive system aspects and start considering possible applications. In chapter 4 we have argued that multi-topic document filtering cannot be performed on the basis of a quantitative relevance score alone. We have suggested ways of exploiting additional evidence of a document's aboutness, derived from the profile's topology, to support enhanced topical presentation of the filtering results and document summarisation. We have also argued that document evaluation can be performed on a per paragraph or per sentence basis, thus identifying specific parts of a document that might be of interest.

Furthermore, we described additional personalisation services that can be supported by the proposed hierarchical profile representation. These include, automated query formulation, expert finding and collaborative filtering. Nootropia's scope is therefore broader that the mere evaluation of documents. A possible application domain is knowledge management. Of course, any such application implies interface issues that have to be resolved and maybe the need for thresholding, for making the binary decision between accepting or rejecting a document.

Finally, since syntactic rules have been purposely avoided, it is possible in principle that the proposed hierarchical representation and profile adaptation through self-organisation, could be applied to other media, like audio and image, for which features can be statistically extracted. Nootropia's application for personalised music delivery is one of our future research goals.

6.2.2 Projections

We devised Nootropia's adaptive filtering core to be a hierarchical, self-organising network of terms, that can be used computationally for document evaluation. In other words, we synthesised a complex adaptive network, an autopoietic network we might say. Inspired by biology, we stressed non-linearity and self-organisation. Adaptive document filtering and biology have been brought closer and interesting questions arose in the process. These include:

- Does the hierarchical profile exhibit the common characteristics of biological, social, language and other complex adaptive networks?

- What is its theoretical importance as an alternative evolutionary model to GAs?
- What is the underlying computational model and what does it tell us about complexity, self-organisation and adaptation?

These and other theoretical questions already occupy us. They suggest that in addition to its important application for adaptive document filtering and other personalisation services, Nootropia may be used as a testbed for experimenting with concepts such as complexity, evolvability, self-organisation and adaptation that have attracted the attention not only of biologists, but also of social scientists, linguists and experts from a variety of other disciplines.

Our future will abound with digital information. Personalisation will necessarily become an intrinsic part of our interaction with the information overloaded environment. We believe that with Nootropia, we contribute to this imminent trend, and we highlight the important role biologically inspired computing can play. Our new perspective on adaptive document filtering invites multi-disciplinary research and expands its scientific scope.

Appendix A

Summary of Topic Codes

Table A.1: Summary of codes, thematic subjects and statistical characteristics of the topics involved in the experiments.

			Number of documents in	
Topic	Code	Description	Test Set	Training Set
R1	C11	STRATEGY/PLANS	23651	597
R2	C12	LEGAL/JUDICIAL	11563	351
R3	C13	REGULATION/POLICY	36463	821
R4	C14	SHARE LISTINGS	7250	146
R5	C1511	ANNUAL RESULTS	22813	352
R6	C16	INSOLVENCY/LIQUIDITY	1871	42
R7	C171	SHARE CAPITAL	17876	403
R8	C172	BONDS/DEBT ISSUES	11202	251
R9	C173	LOANS/CREDITS	2560	68
R10	C174	CREDIT RATINGS	5625	212
R20	C313	MARKET SHARE	1074	38
R21	C32	ADVERTISING/PROMOTION	2041	39
R29	E12	MONETARY/ECONOMIC	26402	630
R32	E131	CONSUMER PRICES	5492	140
R41	E311	INDUSTRIAL PRODUCTION	1658	35
R50	E71	LEADING INDICATORS	5104	149
R58	G157	EC COMPETITION/SUBSIDY	1991	41
R68	GJOB	LABOUR ISSUES	16770	419
R79	GWELF	WELFARE, SOCIAL SERVICES	1818	42

Appendix B

List of Publications

N. Nanas, V. Uren and A. de Roeck. Nootropia: a User Profiling Model based on a Self-Organising Term Network. In *3rd International Conference on Artificial Immune Systems*, Springer-Verlag, 2004.

N. Nanas, V. Uren and A. de Roeck. A comparative evaluation of term weighting methods in information filtering. In *4th International Workshop on Natural Language and Information Systems (NLIS '04)*, pages 13–17, IEEE Computer Science, 2004.

N. Nanas, V. Uren, A. de Roeck, and J. Domingue. Beyond trec's filtering track. In *4th International Conference on Language Resources and Evaluation (LREC 2004)*, 2004.

N. Nanas, V. Uren, A. de Roeck, and J. Domingue. Multi-topic information filtering with a single user profile. In *3rd Hellenic Conference on Artificial Intelligence*, pages 400–409, Springer-Verlag, 2004.

N. Nanas, V. Uren, A. de Roeck, and J. Domingue. Adaptive document filtering with a self-organising population of terms. In *Symposium on Evolvability & Interaction: Evolutionary Substrates of Communication, Signalling,* *and Perception in the Dynamics of Social Complexity*, October 2003.

N. Nanas, V. Uren, A. de Roeck, and J. Domingue. Building and applying a concept hierarchy representation of a user profile. In *26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 198–204. ACM press, 2003.

References

[1] K. Aas. A survey on personalized information filtering systems for the world wide web. Technical Report 922, Norwegian Computing Center, 1997.

[2] J. Allan. Incremental relevance feedback for information filtering. In *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 270–278, 1996.

[3] G. Amati, F. Crestani, F. Ubaldini, and S. Nardis. Probabilistic learning for information filtering. In *RIAO 1997,Computer-Assisted Information Searching on Internet*, Montreal, Canada, 1997.

[4] G. Amati, D. D' Aloisi, V. Giannini, and F. Ubaldini. A framework for filtering news and managing distributed data. *Journal of Universal Computer Science*, 3(8):1007–1021, 1997.

[5] P. Anick and S. Tipirneri. The paraphrase search assistant: Terminological feedback for iterative information seeking. In M. Hearst, F. Gey, and R. Tong, editors, *22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 153–159, 1999.

[6] A. Arampatzis. Personal communication, 2003.

[7] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell. Webwatcher: A learning apprentice for the world wide web. In *1995 AAAI Spring Symposium on Information Gathering from Heterogeneous Distributed Environments*, Mar. 1995.

[8] P. E. Baclace. Personal information intake filtering. In *Bellcore Information Filtering Workshop*, 1991.

[9] P. E. Baclace. Competitive agents for information filtering. *Communications of the ACM*, 35(12):50, 1992.

[10] M. Balabanović. An Adaptive Web Page Recommendation Service. In *Conference on Autonomous Agents*, Marina del Rey, CA, 1997.

[11] M. Balabanovic. Exploring versus exploiting when learning user models for text recommendation. *UMUAI*, 8(1-2):71–102, 1998.

[12] M. Balabanović and Y. Shoham. Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, Mar. 1997.

[13] A.-L. Barabási. *The New Science of Networks.* Perseus Publishing, Cambridge, Massachusetts, 2002.

[14] C. D. Batty. The automatic generation of index languages. *Journal of Documentation*, 25(2):142–149, 1969.

[15] R. K. Belew. Adaptive information retrieval: using a connectionist representation to retrieve and learn about documents. *ACM*, 1989.

[16] N. J. Belkin and W. B. Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29-38, 1992.

[17] P. J. Bentley. *Digital Biology.* Headline Book Publishing, 2001.

[18] S. K. Bhatia. Selection of search terms based on user profile. In *SIGAPP Symposium on Applied Computing: Technological Challenges of the 1990's*, volume 2, pages 224–233, Kansas City, MO USA, 1992.

[19] D. Billsus and M. Pazzani. A hybrid user model for news story classification. In *7th International Conference on User Modeling*, Banff, Canada, June 1999.

[20] D. Billsus and M. Pazzani. A personal news agent that talks, learns and explains. In *3rd International Conference on Autonomous Agents*, Seattle, WA, 1999.

[21] A. Bookstein and D. R. Swanson. Probabilistic models of automatic indexing. *Journal of the American Society for Information Science*, 25:312–318, 1974.

[22] A. Bookstein and D. R. Swanson. A decision theoretic foundation for indexing. *Journal of the American Society for Information Science*, 26(1):45–50, 1975.

[23] U. M. Borghoff and R. Pareschi. Information technology for knowledge management. *Journal of Universal Computer Science*, 3(8):835–842, 1997.

[24] M. Boughanem, C. Chrisment, and M. Tmar. Mercure and Mercure-Filtre applied for web and filtering tasks at TREC-10. In *TREC-10*, 2001.

[25] J. Budzik and K. Hammond. Watson: Anticipating and contextualizing information needs. In *ASIS 1999 Annual Conference*, 1999.

[26] U. ¸Cetintemel, M. J. Franklin, and C. L. Giles. Self-adaptive user profiles for large-scale data delivery. In *International Conference on Data Engineering*, pages 622–633, San Diego, CA, Feb. 2000.

[27] J. Callan. Learning while filtering documents. In *21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*, pages 224–231, 1998.

[28] J. P. Callan, Z. Lu, and W. B. Croft. Searching distributed collections with inference networks. In *18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1995.

[29] F. Capra. *The Web of Life.* Harper Collins, 1996.

[30] H. Chen. Machine learning for information retrieval: Neural networks, symbolic learning and genetic algorithms. *Journal of the American Society for Information Science*, 46(3):194–216, 1995.

[31] H. Chen, K. J. Lynch, K. Basu, and T. D. Ng. Generating, integrating, and activating thesauri for concept-based document retrieval. *IEEE Expert*, 8(2):25–34, 1993.

[32] L. Chen and K. Sycara. Webmate : A personal agent for browsing and searching. In *2nd International Conference on Autonomous Agents*, 1998.

[33] B. Chiu and G. Webb. Using decision trees for agent modeling: Improving prediction performance. *User Modeling and User-Adapted Interaction*, 8(1/2), 1998.

[34] Y.-M. Chung, W. M. Pottenger, and B. R. Schatz. Automatic subject indexing using an associative neural network. In *3rd ACM conference on Digital libraries*, pages 59–68. ACM Press, 1998.

[35] K. W. Church. One term or two? In *18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*, pages 310–318, Seattle, WA USA, 1995.

[36] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. In *27th Meeting of the Association for Computational Linguistics*, pages 76–83, Vancouver, B.C, 1989.

[37] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin. Combining content-based and collaborative filters in an online newspaper. In *ACM SIGIR Workshop on Recommender Systems*, 19 Aug. 1999.

[38] W. W. Cohen and Y. Singer. Context-sensitive learning methods for text categorization. In H.-P. Frei, D. Harman, P. Schäuble, and R. Wilkinson, editors, *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 307–315, Zürich, CH, 1996. ACM Press, New York, US.

[39] W. S. Cooper, A. Chen, and F. C. Gey. Experiments in the probabilistic retrieval of full text documents. In *3rd Text Retrieval Conference (TREC-3) Draft Conference Papers*, Gaithersburg, MD : National Institute of Standards and Technology, 1994.

[40] F. Crestani. Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6):453–482, 1997.

[41] C. J. Crouch and B. Yang. Experiments in automatic statistical thesaurus construction. In *15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–87, 1992.

[42] N. J. Davies, R. S. Stewart, and R. Weeks. Knowledge sharing agents over the world wide web. *British Telecom Technology Journal*, 16(3):104–109, 1998.

[43] R. Dawkins. *The Selfish Gene.* Oxford University Press, 1990.

[44] O. de Vel and S. Nesbitt. A collaborative filtering agent system for dynamic virtual communities on the web.

[45] S. Deerwester, S. T. Dumais, G. W. Landauer, and R. Hashman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[46] P. J. Denning. ACM president's letter: electronic junk. *Communications of the ACM*, 25(3):163–165, 1982.

[47] G. Desjardins and R. Godin. Combining relevance feedback and genetic algorithm in an internet information filtering engine. In *RIAO 2000*, 2000.

[48] L. B. Doyle. Semantic road maps for literature searchers. *Journal of the ACM*, 8:553–578, 1962.

[49] H. P. Edmundson and R. E. Wyllys. Automatic abstracting and indexing - survey and recommendations. *Communications of the ACM*, 4(5):226–234, 1961.

[50] D. Eichmann and P. Srinivasan. Adaptive filtering of newswire stories using two-level clustering. *Information Retrieval*, 5:209–237, 2002.

[51] J. L. Fagan. Automatic phrase indexing for document retrieval: An examination of syntactic and nonsyntactic methods. In *10th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 91–108, June 1987.

[52] P. W. Foltz and S. T. Dumais. Personalized information delivery: an analysis of information filtering methods. *Communications of the ACM*, 35(12):51–60, 1992.

[53] R. Forsyth and R. Rada. *Machine Learning: applications in expert systems and information retrieval.* Ellis Horwood Limited, 1986.

[54] N. Fuhr and C. Buckley. A probabilistic approach for document indexing. *ACM Transaction on Information Systems*, 9(3):223–248, 1991.

[55] N. Fuhr and P. Muller. Probabilistic search term weighting - some negative results. In *10th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 13–18, New Orleans, Louisiana, United States, 1987. ACM Press.

[56] N. Fuhr and U. Pfeifer. Probabilistic information retrieval as combination of abstraction, inductive learning and probabilistic assumptions. *ACM Transactions on Information Systems*, 11(1), 1994.

[57] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971, 1987.

[58] N. Glance, D. Arregui, and M. Dardenne. Knowledge pump: Supporting the flow and use of knowledge. In U. Borghoff and R. Pareschi, editors, *Information Technology for Knowledge Management.* Springer Verlag, 1999.

[59] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70, 1992.

[60] H. Kautz and B. Selman and M. Shah. Combining social networks and collaborative filtering. *Communications of the ACM*, 40(3):63–65, Mar. 1997.

[61] Y. S. Han, Y. K. Han, and K. Choi. Lexical concept acquisition from collocation map. In *Workshop on Acquisition of Lexical Knowledge from Text, 31st Annual Meeting of the ACL*, Columbus, Ohio, 1993.

[62] U. Hanani, B. Shapira, and P. Shoval. Information filtering: Overview of issues, research and systems. *User Modeling and User-Adapted Interaction*, 11:203–259, 2001.

[63] D. Harman. An experimental study of factors important in document ranking. In *9th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 186–193, 1986.

[64] S. P. Harter. A probabilistic approach to automatic keyword indexing. part i: On the distribution of specialty words in a technical literature. *Journal of the American Society for Informaiton Science*, 26(4):197–206, 1975.

[65] S. P. Harter. A probabilistic approach to automatic keyword indexing. part ii: An algorithm for probabilistc indexing. *Journal of the American Society for Informaiton Science*, 26(4):280–289, 1975.

[66] D. A. Hull. The trec-7 filtering track: Description and analysis. In E. M. Voorhess and D. K. Harman, editors, *The 7th Text Retrieval Conference (TREC-7)*, pages 33–56, Gaithesrburg, MD, 1998. NIST Special Publication 500-242.

[67] R. F. i Cancho and R. V. Sole. The small-world of human language. In *Proceedings of the Royal Society of London SeriesB- biological Sciences*, pages 2261–2265, 2001.

[68] N. Ide and J. Veronis. Word sense disambiguation: The state of the art. *Computational Linguistics*, 24(1):1–40, 1998.

[69] F. Idris and S. Panchanathan. Review of image and video indexing techniques. *Journal of Visual Communication and Image Representation*, 8(2):146–166, 1997.

[70] P. Jacobs and L. Rau. Scisor: Extracting information from on-line news. *Communications of the ACM.*, 33(11):88–97, 1990.

[71] C. Jamie. Document filtering with inference networks. In *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 262–269, 1996.

[72] A. Jennings and H. Higuchi. A personal news service based on a user model neural network. In *IEICE Transactions on Information and Systems*, 1992.

[73] T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *14th International Conference on Machine Learning ICML97*, pages 143–151, 1997.

[74] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In C. Nédellec and C. Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[75] S. Johnson. *Emergence.* Penguin Books Ltd, 2001.

[76] W. P. Jones and G. W. Furnas. Pictures of relevance: A geometric analysis of similarity measures. *Journal of the American Society of Information Science*, 38(6):420–442, May 1986.

[77] K. Kageura and B. Umino. Method of automatic term recognition. *Terminology*, 3(2):259–290, 1996.

[78] T. Kamba, H. Sakagami, and Y. Koseki. Anatagonomy: a personalized newspaper on the world wide web. *International Journal of Human-Computer Studies*, 46(6):789–803, 1997.

[79] H. Kautz, B. Selman, and M. Shah. The hidden web. *AI Magazine*, 18(2):27–36, 1997.

[80] F. Kilander. A brief comparison of news filtering software, 1995.

[81] S. Kirby. Natural language from artificial life. *Artificial Life*, 8(2):185–215, 2002.

[82] E. Kirkwood and C. Lewis. *Understanding Medical Immunology*. John Wiley & Sons, Chichester, 1989.

[83] R. Klinkenberg and I. Renz. Adaptive information filtering: Learning in the presence of concept drifts. In *Learning for Text Categorization*, pages 33–40, Menlo Park, California, 1998. AAAI Press.

[84] M. Klusch, editor. *Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet.* Springer-Verlag, 1999.

[85] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela. Self organization of a massive document collection. *IEEE Transactions on Neural Networks*, 11(3), May 2000.

[86] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. GroupLens: Applying collaborative filtering to usenet news. *Communications of the ACM.*, 40(3):77–87, 1997.

[87] B. Krulwich and C. Burkey. The InfoFinder agent: Learning user interests through heuristic phrase extraction. *IEEE Expert*, pages 22–27, 1997.

[88] K. L. Kwok. A neural network for probabilistic information retrieval. In *12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–30, Cambridge, Massachusetts, USA, 1989.

[89] K. Lagus. Text retrieval using self-organized document maps. *Neural Processing Letters*, 15(1):21–29, 2002.

[90] W. Lam, S. Mukhopadhyay, J. Mostafa, and M. Palakal. Detection of shifts in user interests for personalized information filtering. In *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 317–325, Zurich, Switzerland, 1996.

[91] K. Lang. NewsWeeder: Learning to filter netnews. In *12th International Conference on Machine Learning (ICML95)*, 1995.

[92] C. Lanquillon and I. Renz. Adaptive information filtering: Detecting changes in text streams. In *ACM CIKM International Conference on Information and Knowledge Management, Kansas City, Missouri, USA, November 2-6, 1999*, pages 538–544. ACM, 1999.

[93] D. Lawrie and B. W. Croft. Discovering and comparing topic hierarchies. In *RIAO 2000 Conference*, pages 314–330, 2000.

[94] D. Lawrie, B. W. Croft, and A. Rosenberg. Finding topic words for hierarchical summarization. In *24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 349–357, New Orleans, Louisiana, United States, 2001. ACM Press.

[95] D. Lewis and W. Croft. Term clustering of syntactic phrases. In *13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 385–404, 1990.

[96] D. D. Lewis and M. Ringuette. A comparison of two learning algorithms for text categorization. In *Symposium on Document Analysis and Information Retrieval.*, 1994.

[97] H. Lieberman. Letizia: An agent that assists web browsing. In *International Joint Conference on Aritifical Intelligence*, Montreal, CA, 1995.

[98] S. Loeb. Architecting personalized delivery of multimedia information. *Communications of the ACM*, 35(12):39–47, 1992.

[99] R. M. J. Losee. Minimizing information overload: the ranking of electronic messages. *Journal on Information Science*, 15(3):179–189, 1989.

[100] J. B. Lovins. Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics*, 11:22–31, 1968.

[101] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[102] P. Maes. Agents that reduce work and information overload. *Communications of the ACM*, 37(7):30–40, 1994.

[103] T. W. Malone, K. R. Grant, F. A. Turbak, S. A. Brobst, and M. D. Cohen. Intelligent information-sharing systems. *Communications of the ACM*, 30(5):390–402, 1987.

[104] C. D. Manning and H. Schutze. *Foundations of Statistical Natural Language Processing.* MIT Press, 1999.

[105] M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing and information retrieval. *Journal of the ACM*, 7:216–244, 1960.

[106] H. R. Maturana and F. J. Varela. *Autopoiesis and Cognition.* Dordrecht, Holland, 1980.

[107] M. McElligott and H. Sorensen. An emergent approach to information filtering. *UCC Computer Science Journal*, 1(4), 1993.

[108] M. McElligott and H. Sorensen. An evolutionary connectionist approach to personal information filtering. In *4th Irish Neural Networks Conference '94*, University College Dublin, Ireland, 1994.

[109] F. Menczer. ARACHNID: Adaptive retrieval agents choosing heuristic neighborhoods for information discovery. In *Machine Learning: Proceedings of the Fourteenth International Conference*, pages 227–235, 1997.

[110] F. Menczer and R. Belew. Adaptive information agents in distributed textual environments. In *2nd International Conference on Autonomous Agents*, Minneapolis, MN, 1998.

[111] F. Menczer and A. E. Monge. Scalable web search by adaptive online agents: An infospiders case study. In M. Klusch, editor, *Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet.*, pages 323–347. Springer-Verlag, 1999.

[112] C. Michel. Diagnostic evaluation of a personalized filtering information retrieval system. Methodology and experimental results. In *RIAO 2000*, Collège de France, Paris, 2003.

[113] G. Miller. Special issue, wordnet: An on-line lexical database. *International Journal of Lexicography*, 4, 1990.

[114] M. S. Miller and K. E. Drexler. Markets and computation: Agoric open systems. Technical report, Agoric Inc., 2000.

[115] J. Mingers. Embodying information systems: the contribution of phenomenology. *Information and Organization*, 11:103–128, 2001.

[116] M. Mitchell and R. Belew, editors. *Adaptive Individuals in Evolving Population Models.* Santa Fe Institute, Studies in the Sciences of Complexity. Addison-Wesley, 1996.

[117] T. M. Mitchell. *Machine Learning.* McGraw-Hill, 1997.

[118] S. Mizzaro. How many relevances in information retrieval? *Interacting With Computers*, 10(3):305–322, 1998.

[119] D. Mladeni'c. Text-learning and related intelligent agents. Technical report, Department of Intelligent Systems, J.Stefan Institute, Slovenia, 1999.

[120] D. Mladeni'c. Using text learning to help web browsing. In *9th International Conference on Human-Computer Interaction (HCI International 2001)*, New Orleans, LA, 2001.

[121] M. E. Müller. Machine learning based user modeling for www search. In *7th International Conference on User Modeling*, 1999.

[122] M. Moens and J. Dumortier. Text categorization: the assignement of subject descriptors to magazine articles. *Information Processing and Management.*, 36(6):841–861, 2000.

[123] M. Morita and Y. Shinoda. Information filtering based on user behavior analysis and best match text retrieval. In *17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 272–281. Springer-Verlag New York, Inc., 1994.

[124] T. Morrison and U. Aickelin. An artificial immune system as a recommender for web sites. In *1st International Conference on Artificial Immune Systems*, 2002.

[125] J. Mostafa, S. Mukhopadhyay, M. Palakal, and W. Lam. A multilevel approach to intelligent information filtering: model, system, and evaluation. *ACM Transactions on Information Systems (TOIS)*, 15(4):368–399, 1997.

[126] A. E. Motter, A. P. S. de Moura, and P. Dasgupta. Topology of the conceptual network of language. *Physical Review E*, 65(065102(R)), 2002.

[127] A. Moukas and P. Maes. Amalthaea: An evolving multi-agent information filtering and discovery system for the www. *Autonomous Agents and Multi-Agent Systems.*, 1(1):59–88, 1998.

[128] A. Moukas, G. Zacharia, and P. Maes. Amalthaea and Histos: Multiagent systems for www sites and reputation recommendations. In M. Klusch, editor, *Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet*, pages 293–322. Springer-Verlag, 1999.

[129] C. G. Nevill-Manning, I. H. Witten, and G. W. Paynter. Lexically-generated subject hierarchies for browsing large collections. *International Journal on Digital Libraries*, 2(2-3):111–123, 1999.

[130] H. T. Ng, W. B. Goh, and K. L. Low. Feature selection, perception learning, and a usability case study for text categorization. In *20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*, pages 67–73, New York, 1997.

[131] K. Nigam and A. McCallum. Pool-based active learning for text classification, working notes of learning from text and the web. In *Conference on Automated Learning and Discovery CONALD-98*, 1998.

[132] M. Oakes, R. Gaizauskas, H. Fowkes, A. Jonsson, V. Wan, and M. Beaulieu. Comparison between a method based on the chi-square test and a support vector machine for document classification. In *ACM Special Interest Group on Information Retrieval (SIGIR'01)*, 2001.

[133] D. W. Oard. The state of the art in text filtering. *User Modeling and User-Adapted Interaction: An Internation Journal*, 7(3):141–178, 1997.

[134] D. W. Oard and Marchionini. A conceptual framework for text filtering. Technical Report CS-TR-3643, University of Maryland, 1996.

[135] Y. C. Park and K.-S. Choi. Automatic thesaurus construction using bayesian networks. *Information Processing and Management.*, 32(5):543–553, 1996.

[136] V. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

[137] M. Pazzani, J. Muramatsu, and D. Billsus. Syskill & webert: identifying interesting web sites. In *13th National Conference on Artificial Intelligence*, Portland, Oregon, 1996.

[138] M. Porter. An algorithm for suffix stripping. *Program*, 14(3), 1980.

[139] M. F. Porter. Implementing a probabilistic information retrieval system. *Information Technology: Research and Development*, 1:131–156, 1982.

[140] A. Pretschner and S. Gauch. Personalization on the web. Technical report, Information and Telecommunication Technology Center, 1999.

[141] T. Quick, K. Dautenhahn, C. L. Nehaniv, and G. Roberts. The essence of embodiment: A framework for understanding and exploiting structural coupling between system and environment. *CASYS'99*, 1999.

[142] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A.-L. Barabási. Hierarchical organization of modularity in metabolic networks. *Science*, 297:1551–1555, August 2002.

[143] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

[144] B. J. Rhodes and T. Starner. Remembrance agent: A continuosly running information retrieval system. In *1st International Conference on the Practical Applications of Intelligent Agents and Multi Agent Technology (PAAM '96)*, pages 487–495, 1996.

[145] E. Riloff. Little words can make a big difference for text classification. In *18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 130–136, 1997.

[146] S. Robertson and I. Soboroff. The TREC 2001 filtering track report. In *TREC-10*, 2001.

[147] S. Robertson and S. Walker. Threshold setting in adaptive filtering. *Journal of Documentation*, 56(3):312–331, 2000.

[148] S. E. Robertson. The probability ranking principle in ir. *Journal of Documentation*, 33(4):294–304, December 1977.

[149] S. E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.

[150] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *17th Annual International ACM SIGIR Conference on Research and Development in*

Information Retrieval, pages 232–241, Dublin Ireland, 1994. Springer-Verlag.

[151] J. Rocchio. *Relevance Feedback in Information Retrieval*, chapter 14, pages 313–323. Prentice-Hall Inc., 1971.

[152] T. Rose, M. Stevenson, and M. Whitehead. The Reuters Corpus Volume 1 - from yesterday's news to tomorrow's language resources. In *3rd International Conference on Language Resources and Evaluation*, 2002.

[153] J. Rucker and M. J. Polanco. Siteseer: Personalized navigation for the web. *Communications of the ACM*, 40(3):73–75, Mar. 1997.

[154] M. E. Ruiz and P. Srinivasan. Hierarchical text categorization using neural networks. *Information Retrieval*, 5:87–118, 2002.

[155] G. Salton. Recent studies in automatic text analysis and document retrieval. *Journal of the ACM*, 20(2):258–278, 1973.

[156] G. Salton. Another look at automatic text-retrieval systems. *Communication of the ACM*, 29(7):648–656, July 1986.

[157] G. Salton and C. Buckley. On the use of spreading activation methods in automatic information retrieval. In *11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*, pages 147–160, Grenoble France, 1988.

[158] G. Salton and M. Lesk. Computer evaluation of indexing and text processing. *Journal of the ACM*, 15(1):8–36, 1968.

[159] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill Inc., 1983.

[160] G. Salton and H. Wu. The measurement of term importance in automatic indexing. *Journal of the American Society for Information Science*, 32:175–186, 1981.

[161] G. Salton and C. S. Yang. On the specification of term values in automatic indexing. *Journal of Documentation.*, 29(4):351–372, 1973.

[162] G. Salton, C. S. Yang, and C. T. Yu. A theory of term importance in automatic text analysis. *Journal of the American Society for Information Science*, 26(1):33–44, 1975.

[163] M. Sanderson and B. W. Croft. Deriving concept hierarchies from text. In *22nd Annual Internation ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–213, Berkeley, California, United States, 1999. ACM Press.

[164] R. Schapire and Y. Singer. Boostexter: A system for multiclass multilabel text categorization, 1998.

[165] R. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In *21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998.

[166] H. Schütze, D. A. Hull, and J. O. Pedersen. A comparison of classifiers and document representations for the routing problem. In *18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 229–237, 1995.

[167] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), 2002.

[168] Y. Seo and B. Zhang. A reinforcement learning agent for personalized information filtering. In *Intelligent User Interfaces*, pages 248–251, New Orleans, LA, 2000.

[169] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication.* University of Illinois Press, 1949.

[170] U. Shardanand and P. Maes. Social information filtering: algorithms for automatic word of mouth. In *Conference on Human Factors in Computing Science.*, pages 210–217, 1995.

[171] B. D. Sheth. *A Learning Approach to Personalized Information Filtering.* Master of Science, Massachusetts Institute of Technology, 1994.

[172] A. Singhal, G. Salton, M. Mitra, and C. Buckley. Document length normalization. *Information Processing and Management*, 32(5):619–633, 1996.

[173] H. Sorensen and M. McElligott. An online news agent. In *BCS Intelligent Agents Workshop, British Computer Society*, Britain, 1995.

[174] H. Sorensen, A. O' Riordan, and C. O' Riordan. Profiling with the informer text filtering agent. *Journal of Universal Computer Science*, 3(8):988–1006, 1997.

[175] K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–20, 1972.

[176] A. M. Steier and R. K. Belew. Exporting phrases: A statistical analysis of topical language. In R. Casey and B. Croft, editors, *2nd Symposium on Document Analysis and Information Retrieval*, 1994.

[177] C. Stevens. Automating the creation of information filters. *Communications of the ACM*, 35(12):48, 1992.

[178] D. R. Tauritz, J. N. Kok, and I. G. Sprinkhuizen-Kuyper. Adaptive information filtering using evolutionary computation. *Information Sciences*, 122(2–4):121–140, 2000.

[179] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter. Phoaks: a system for sharing recommendations. *Communications of the ACM*, 40(3):59–62, 1997.

[180] A. M. Tjoa, M. Höfferer, G. Ehrentraut, and P. Untersmeyer. Applying evolutionary algorithms to the problem of information filtering. In *8th International Workshop on Database and Expert Systems Application*, pages 450–458, Toulouse, France, 1997. IEEE Computer Press.

[181] H. Turtle and W. B. Croft. Inference networks for document retrieval. In *13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1–24, 1990.

[182] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5), 2002.

[183] K. Tzeras and S. Hartmann. Automatic indexing based on bayesian inference networks. In R. Korfhage, E. M. Rasmussen, and P. Willett, editors, *16th ACM International Conference on Research and Development in Information Retrieval*, pages 22–34, Pittsburgh, PA USA, 1996. ACM.

[184] C. J. van Rijsbergen. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–199, 1977.

[185] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 2nd edition, 1979.

[186] R. C. Veltkamp, H. Burkhardt, and H. Kriegel, editors. *State-of-the-Art in Content-Based Image and Video Retrieval.* Kluwer Academic Publishers, 2001.

[187] C. L. Viles and J. C. French. On the update of term weights in dynamic information retrieval systems. In *ACM CIKM Conference on Information and Knowledge Management*, 1995.

[188] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small world' networks. *Nature*, 393, June 1998.

[189] G. I. Webb, B. C. Chiu, and M. Kuzmycz. Comparative evaluation of alternative induction engines for feature based modelling. *International Journal of Artificial Intelligence in Education*, 8, 1997.

[190] G. I. Webb and M. Kuzmycz. Feature based modelling: A methodology for producing coherent, consistent, dynamically changing models of agents' competency. *User Modelling and User Assisted Interaction*, 5(2):117–150, 1996.

[191] G. I. Webb, M. J. Pazzani, and D. Billsus. Machine learning for user modeling. *User Modeling and User-Adapted Interaction*, 11:19–29, 2001.

[192] A. Weigend, E. Wiener, and J. O. Pedersen. Exploiting hierarchy in text categorization. *Information Retrieval*, 1(3):193–216, 1999.

[193] S. Wermter. Neural networks agents for learning semantic text classification. *Information Retrieval*, 3:87–103, 2000.

[194] G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23:69–101, 1996.

[195] D. H. Widyantoro, T. R. Ioerger, and J. Yen. An adaptive algorithm for learning changes in user interests. In *ACM/CIKM'99 Conference on Information and Knowledge Management*, Kansas City, MO, 1999.

[196] D. H. Widyantoro, T. R. Loerger, and J. Yen. Learning user interests dynamics with a three-descriptor representation. *JASIS*, 2000.

[197] D. H. Widyantoro, J. Yin, M. S. E. Nasr, L. Yang, A. Zacchi, and J. Yen. Alipes: A swift messenger in cyberspace. In *Spring Symposium on Intelligent Agents in Cyberspace*, pages 62–67, Palo Alto, Mar. 1999.

[198] E. Wiener, J. O. Pedersen, and A. S. Weigend. A neural netword approach to topic spotting. In *4th Annual Symposium on Document Analysis and Information Retrieval (SDAIR'95).*, 1995.

[199] R. Wilkinson and P. Hingston. Using the cosine measure in a neural network for document retrieval. In *14th Annual Internation ACM SIGIR conference on Research and Development in Information Retrieval*, pages 202–210. ACM Press, 1991.

[200] W. Winiwarter. PEA - a personal email assistant with evolutionary adaptation. *Internation Journal of Information Technology*, 5(1), 1999.

[201] J. Xu and B. W. Croft. Query expansion using local and global document analysis. In *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, 1996.

[202] T. Yan and H. Garcia-Molina. SIFT – a tool for wide-area information dissemination. In *1995 USENIX Winter Technical Conference*, New Orleans, Jan. 1995.

[203] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *14th International Conference on Machine Learning (ICML '97)*, 1997.

[204] Y. Yang and J. Wilbur. Using corpus statistics to remove redundant words in text categorization. *Journal of the American Society for Information Science*, 47(5):357–369, 1996.

[205] C. T. Yu, K. Lam, and G. Salton. Term weighting in information retrieval. using the term precision model. *Journal of the ACM*, 29(1):152–170, 1982.

[206] C. T. Yu and G. Salton. Effective information retrieval using term accuracy. *Communications of the ACM*, 20:135–142, 1977.

[207] T. Zhang and C. J. Kuo. *Content-Based Audio Classification and Retrieval for Audiovisual Data Parsing.* Kluwer Academic Publishers, 2001.

[208] G. K. Zipf. *Human behaviour and the principle of least effort. An introduction to human ecology.* New York: Hafner reprint, 1949.

Index

Adaptive Document Filtering, 5, 163
agent, 10, 44
Ant Colonies, 167
Artificial Immune Networks, 167
Artificial Intelligence, 3
autopoiesis, **167**
Average Uninterpolated Precision (AUP), **72**

Bayes' theorem, 35

connectionist approaches
- inference networks, 19
- neural networks, 19, 112
- semantic networks, 19, 112
- spreading activation, 36, 53, 112

contingency table, **28**

digital information, 1
dimensionality reduction, 17, 27
- stemming, 17
  - Lovins, 17
  - Porter, 17
- stop word removal, 17

document indexing, **16**, 18, 19
- "bag of words", 17
- inverted index, 19
- latent semantic indexing, **18**
- n-gram analysis, 16
- phrase-based, 18

documentation redundancy, **15**, 38, 83

expert finding, 46

Genetic Algorithms (GAs), 49, 65–69, 165, 167

Hebbian learning, 68, 165

Information Filtering (IF), 2, 43
- adaptive, **60**
- clustering, 49
- collaborative, 43, **44**, 46, 55
- content-based, 43, **44**, 46, 55
- social, 43

information overload, **1**, 9
Information Retrieval (IR), 2, **15**, 26
- ad-hoc, 25
- best match, 32
- document-based, 26
- exact match, 32
- probabilistic, **34**, 36
- query, 25
  - automatic formulation, 46
  - expansion, 25, 41

information sources
- dynamic, 11, **12**
- static, 11, **12**, **13**

intelligent information assistant, 44

Knowledge Management (KM), 55

language networks, 15
language redundancy, **14**, 38, 83
learning algorithms, 60, 62–64
- dual profiles, 63, 164
- linear, 61, 164
- machine learning, 62, 164

reinforcement, 164
reinforcement learning, 62
Rocchio's, **61**
lexical correlations, **14**, 18, 83
living systems, 166

Memetic Algorithms (MAs), 68–69, 165

Nootropia, **4**, 5

personal information assistant, 10
Personalised Information Delivery (PID), 2, 7–10, 69
precision, 71
probability ranking principle, **34**
profile, **2**

RCV1, 13, **72**
reality redundancy, **14**, 83
recall, 71
Relative Document Frequency (RelDF), **89**
relevance feedback
explicit, **57**
implicit, **58**
relevance information, 10
relevant information, **1**
Rocchio's algorithm, 61

self-organisation, **167**
shared interest matrix, 158
singular value decomposition, 18
structural coupling, 168
subject hierarchies, 12

term dependence, 38
collocation map, 40
concept hierarchies, 19
lexical hierarchies, 41, **41**, 42, 84
subsumption hierarchies, **41**, 42, 83
dependence tree, 26, 40
term independence, 18
term weighting, 17, 20–24, 28–32, 51, 86–90
χ^2, 31
exhaustive, 20
information gain, 30
inverse document frequency, 22
mutual information, 31
query term weighting, 29
rdf, 30
relative frequency technique, 22
ridf, 24
specificity, 20
term frequency, 21
tfidf, 23
term window, 39
document context, 40
local context, 39
topical context, 39
Text Categorisation (TC), 2, **15**, 27, 36
classifier, 27, 54
decision trees, 27
linear, 28, 37, 111
naive Bayes, 27
nearest-neighbour, 27
Neural Networks, 27
non-linear, 28, 37
rule-based, 27
Support Vector Machines, 27
thesauri, 18
global, 25
local, 26
WordNet, 19
threshold, 56
topical correlations, **14**, 41, 83
TREC, 60, 70, 72–75
filtering track, 73

user feedback, 10

user interests
  long-term, **59**, 60, 163
  short-term, **59**, 60, 163
user profile, 8, 43
  adaptation, 59
  initialisation, 50–52
  representation, 46–50
    connectionist, 49, 111, 165

Vector Space Model, **17**, 27, 32, 111
  cosine similarity, **34**, 52
  inner product, **33**, 54
virtual users, 75, 76
vocabulary problem, **18**
  polysemy, 19
  synonymy, 19

Zipf's law, 14