

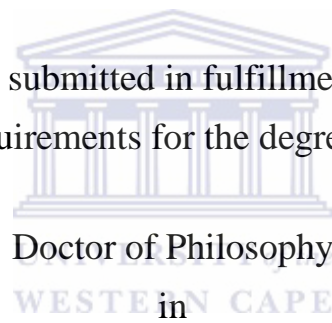
UNIVERSITY OF THE WESTERN CAPE

**Context-Awareness for Adaptive Information
Retrieval Systems**

by

Kehinde Kayode Agbele

a thesis submitted in fulfillment of the
requirements for the degree of



Doctor of Philosophy

in

Computer Science

Faculty of Science
Department of Computer Science

May 2014

Declaration of Authorship

I, Kehinde Kayode Agbele, declare that this thesis titled "Context-Awareness for Adaptive Information Retrieval Systems" is my own research work, that it has not been submitted before for any degree or examination in any other university, and that all the sources I have used or cited have been indicated and acknowledge by complete references.

Signed: 



KEHINDE KAYODE AGBELE

Date:..05-10-2014.....

Dedication

This project is dedicated to

The Almighty God the alpha & Omega,

my beloved wife (*Olufunke Omolara Agbele*),



the lovely children (Ayomiposi & Ayopelumi),

My beloved parent (*Mr. and Chief (Mrs.) Rachael Oyebanji Agbele*),

&

My beloved siblings,

For your prayers, understanding, and love for the time I was away to another shore due to this noble study.

Abstract

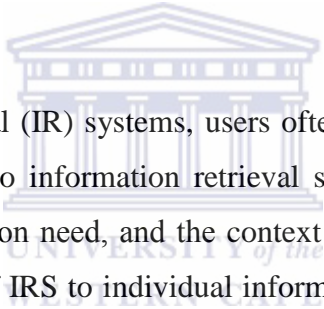
Faculty of Science
Department of Computer Science

Doctor of Philosophy

by

Kehinde Kayode Agbele

Supervisor: Prof. Henry O. Nyongesa



When using Information Retrieval (IR) systems, users often present search queries made of ad-hoc keywords. It is then up to information retrieval systems (IRS) to obtain a precise representation of user's information need, and the context of the information. This research study investigates optimization of IRS to individual information needs in order of relevance. The research addressed development of algorithms that optimize the ranking of documents retrieved from IRS. In this thesis, we present two aspects of context-awareness in IR. Firstly, the design of context of information. The context of a query determines retrieved information relevance. Thus, executing the same query in diverse contexts often leads to diverse result rankings. Secondly, the relevant context aspects should be incorporated in a way that supports the knowledge domain representing users' interests. In this thesis, the use of evolutionary algorithms is incorporated to improve the effectiveness of IRS. A context-based information retrieval system is developed whose retrieval effectiveness is evaluated using precision and recall metrics. The results demonstrate how to use attributes from user interaction behaviour to improve the IR effectiveness.

Keywords: Information retrieval (IR), Context awareness, Interactive reinforcement learning (IRL), Relevance, Parameters optimization, Performance measures, Contextual information, Personalization, Clustering, Evolutionary algorithm

Acknowledgement

The completion of this thesis is also attributed to several contributions. Firstly, above all I would like to acknowledge my Lord and Heavenly father God, for the health, ability, love, mercy, and grace He provided me to complete this academic feat.

Special thanks go to my supervisor Professor Henry Nyongesa for his continuous guidance and patient supervision throughout my PhD study. His rich knowledge, abundant experiences and incisive insight always make me feel that there could not have been a better PhD supervisor. His scrutiny of my work has considerably improved my critical analytical thinking for research. This particular thesis would not have been possible without his help. Also, I acknowledge the Director of the Division for Postgraduate Studies and Acting Deputy Dean of Postgraduate studies, Professors Lorna Holtman and Ralf Henkel, respectively.

Besides, I would like to thank my colleagues in the Machine Learning and Intelligent Systems Research Group (UWC), and my friends specifically, Olofintila Opeyemi, Dr Fadipe Seun, Elder Amosu Tobi, Sunday Vodah, Dr. Dele Seluwa, Abidoeye Ademola, Dr. Ademola Adesina, Dr Akinyemi Segun for their encouragement and motivation. Their corporation has made the journey easier. I would like to thank Hardley Scholtz, who helped to test my system. Also, I appreciate the HoD Mathematical Sciences EKSU, Prof. Emmanuel Ibijola.

I feel a deep sense of appreciation to my parents for their endless love that has guided all my visions and formed the most important part of growing-up. To my siblings: Idowu, Temitope, Olusola, Taiwo, Kehinde Junior, and Alaba for their sacrifices and unshakeable support during the tough period of this research project. Special thanks to my twin brother Taiwo for his love, understanding, support and who was always there when needed, helping me to face any difficulties. To my father in-law and late mother in-law I say a big thank you for your love and prayers. Further, I appreciate my father's in the Lord, Pastor Robert Akande and his wife, Pastor Titi Akande and the entire church congregation of Rainbow Family Church, Cape Town, South Africa and Prophet Bayo Olowa and Presiding Pastor Samson Ndukwe of Winner Chapel, Adebayo I, Ado-Ekiti for their unceasing prayers and love.

Last but not least I would like to thank my darling wife; Olufunke not only for her love, sacrifices, endurance, tolerance, and patience but also for her assistance with the immediate family needs while studying abroad, you are a wife in a million and my lovely children, Ayomiposi and Ayopelumi - to whom I can now give undivided attention.

List of Publications

1. **Kehinde Agbele**, Ademola Adesina, Ademola Abidoye, Nureni Azeez, (2012): Context-Aware Stemming Algorithm for Semantically Related Root Words, *African Journal of Computing and ICTs*, Vol. 5, Issue 4, pp. 33-42.
2. **Kehinde Agbele**, Henry Nyongesa and Ademola Adesina, (2010): ICT and Information Security Perspectives in E-Health Systems, *Journal of Mobile Technology*, Vol.4, No 1, pp. 17-22.
3. Ademola O. Adesina, **Kehinde K. Agbele**, Ronald Februarie, Ademola P. Abidoye, Henry O. Nyongesa, (2011): Ensuring the Security and Privacy of Information in Mobile healthcare Communication System, *South African Journal of Science*, Vol. 107, NO 9/10, pp. 1-7.
4. Ademola O. Adesina, **Kehinde K. Agbele** and Henry O. Nyongesa (2010): Text Messaging: a tool in E-health services, *SATNAC Conference 2010*, May 5-8, 2010, Stellenbosch, South Africa.
5. **Kehinde Agbele**, Henry Nyongesa, Ademola Adesina (2014): Personalization of Information Retrieval Model via User Search Behaviours for Ranking Document Relevance. (Under Review: Manuscript ID 140274- *Journals of Information Science and Engineering*).
6. **Kehinde Agbele**, Henry Nyongesa, Ademola Adesina (2013): Algorithm for Information Retrieval Optimization. (Under Review: Manuscript ID 2514- *Journal of Information Science*).
7. **Agbele Kehinde**, Ademola Adesina, Ekong Daniel, and Seluwa Dele (2012): Agent-Based Context-Aware Healthcare Information Retrieval Using DROPT Approach”, *International Journal of Information Retrieval*, Vol. 5, No. 2, pp. 109-118.
8. Ademola O. Adesina, **Kehinde. K. Agbele**, Ademola. P. Abidoye, Henry, O. Nyongesa (2014): Text-Messaging and Retrieval Techniques for mobile health information systems, *Journal of Information Science (SAGE)*, pp. 1-14.
9. **Agbele K.K**, Adesina A.O, Ekong D.O, and Seluwa Dele (2012). ICT a Tool in eHealth Priorities for Human Development and Poverty Reduction in the African Region, *IEEE African Journal of Computing and ICTs*, Special Issue Vol. 5, No. 4, Issue 2, pp: 100-120.
10. Ademola O. Adesina and **Kehinde K. Agbele**, Ademola P. Abidoye and Nureni A. Azeez (2011). A Query-Based SMS Translation in Information Access System. *International Journal of Soft Computing and Engineering*, Vol. 1, Issue 5, pp 13-18.
11. **Kehinde Agbele**, Ademola Adesina, Daniel Ekong, Oluwafemi Ayangbekun (2012). *State-of-the-Art-Review on Relevance of Internet Web Search to Genetic Algorithms*. *Journal of Applied Computational Intelligence and Soft Computing*, Volume 2012, Article ID: 152385, 7pages. Doi: 10.1155\2012\152385.
12. **Kehinde Agbele**, Ademola Adesina, Azeez Nureni, Ademola Abidoye, Ronald Febba (2011): *A Novel Document Ranking Algorithm that supports Mobile Healthcare Information Access Effectiveness*. *Research Journal of Information Technology*, Science Alert, USA, Vol. 3, No. 3, pp 153-166.

List of Figures

Figure 2-1: The Information Retrieval Process.....	7
Figure 2-2: The Multi-Faceted Concepts of Contextual IR.....	15
Figure 2-3: Contextual Information Retrieval Framework	17
Figure 3-1: Weight of a Context Cluster.....	28
Figure 5-1: Process Hierarchy for the Contextual Information Retrieval	43
Figure 5-2: The IR method (DROPT Technique) must always be applicable to the Web Perl Programming language chosen for knowledge domain. Likewise, the context-aware IR model must be able to interoperate with the knowledge domain and make its relevant context-aware information available for the retrieval method (DROPT Technique).	44
Figure 5-3: Sequence diagram for the two Interaction options the user has with the System: (1) Pose context-aware queries and (2) edit her profile.....	44
Figure 5-4: Deployment Diagram for the System, making Use of Existing Agent Repositories Framework and Search Engine Web Services.....	45
Figure 5-5: Overall Context-Aware Information Retrieval Agent System Architecture.....	47
Figure 5-6: The Generation of Context Personalized IR	49
Figure 5-7: Context Aware Personalized System	51
Figure 5-8: Design of Preference Value.....	55
Figure 5-9: Document Ranking Personalization Flow	57
Figure 6-1: Average Precision Graph for Ranking Performance Results	72
Figure 6-2: Precision Graph for Ranking Performance Results at known Relevant Documents	77
Figure 6-3: Ranking Performance Graph Results at the known Relevant Documents.....	84
Figure 6-4: Comparison of DROPT with BM25 and TF-IDF in the P@n Measure.	85
Figure 6-5: Personalized Search Results.....	86
Figure 6-6: Showing Values of 2.47 at $F_{0.05, 4, 97}$	87
Figure 6-7: Showing F-Distribution Table for 3.42	90

List of Tables

<i>Table 4-1: Summary of Ranking Notations</i>	31
<i>Table 5-1: Predictive Document Ranking Model (PDRM) Table for User Model Preference</i>	50
<i>Table 5-2: Relevance Judgments Model (RJM) Table for User Model Judgments</i>	53
<i>Table 5-3: Data Generated by DBD: Mysql, LWP and CAM::PDF for the Five System Users</i>	59
<i>Table 5-4: Derived Information for Ranking Prediction from Domain of Participant 1</i>	65
<i>Table 5-5: Derived Information for Ranking Prediction from Domain of Participant 2</i>	66
<i>Table 5-6: Derived Information for Ranking Prediction from Domain of Participant 3</i>	67
<i>Table 5-7: Derived Information for Ranking Prediction from Domain of Participant 4</i>	68
<i>Table 5-8: Derived Information for Ranking Prediction from Domain of Participant 5</i>	69
<i>Table 6-1: Documents Collection</i>	72
<i>Table 6-2: Mean Average Precision Results for Ranking Performance from 5 Domain of Participants</i>	77
<i>Table 6-3: Precision Results for Ranking Performance at Known Relevant Documents</i>	84
<i>Table 6-4: Feedback Weight Values</i>	78
<i>Table 6-5: Precision and Recall Values for Ranking Performance at Known Relevant Document of domain1</i>	79
<i>Table 6-6: Precision and Recall Values for Ranking Performance at Known Relevant Documents of domain2</i> ...	80
<i>Table 6-7: Precision and Recall Values for Ranking Performance at Known Relevant Documents of domain3</i> ...	81
<i>Table 6-8: Precision and Recall Values for Ranking Performance at Known Relevant Documents of domain4</i> ..	82
<i>Table 6-9: Precision and Recall Values for Ranking Performance at Known Relevant Documents of domain5</i> ...	83
<i>Table 6-10: The Values of Occurrences of Generated Keywords from Domains of Participants 1, 2,3, 4 & 5</i>	88

Contents

DECLARATION OF AUTHORSHIP	II
ABSTRACT	IV
ACKNOWLEDGEMENT	V
LIST OF PUBLICATIONS	VI
LIST OF FIGURES	VII
LIST OF TABLES	VIII
CHAPTER 1	1
INTRODUCTION	1
1.1 BACKGROUND	1
1.2 STATEMENT OF THE PROBLEM	3
1.3 MOTIVATION FOR THE RESEARCH	4
1.4 RESEARCH OBJECTIVES	4
1.5 RESEARCH METHODOLOGY	5
1.6 THESIS ORGANIZATION	5
CHAPTER 2	6
A REVIEW OF INFORMATION RETRIEVAL AND RELATED TECHNIQUES	6
2.1 INTRODUCTION	6
2.2 THE INFORMATION RETRIEVAL PROCESS	7
2.2.1 INDEXING AND SEARCHING.....	9
2.2.2 RANKING.....	10
2.3 INFORMATION RETRIEVAL AND ITS EVALUATION	11
2.3.1 INFORMATION RETRIEVAL EVALUATION APPROACHES.....	12
2.4 PERFORMANCE EVALUATION	14
2.5 CONTEXT-AWARE INFORMATION: A REVIEW	15
2.5.1 THE FRAMEWORK FOR CONTEXTUAL INFORMATION RETRIEVAL.....	16
2.5.1.1 User profile modelling.....	17
CHAPTER 3	19
CLUSTERING ALGORITHMS FOR CONTEXT-AWARENESS	19
3.1 INTRODUCTION	19
3.2 IN OTHER APPROACHES	19
3.2.1 HIERARCHICAL CLUSTERING TECHNIQUES.....	20
3.2.2 EVOLUTIONARY SEARCH-BASED CLUSTERING.....	21
3.3 CONTEXT INFORMATION ACQUISITION	22
3.3.1 CONTEXT-AWARENESS MODELS.....	23
3.3.2 CHARACTERISTICS OF CONTEXT INFORMATION.....	24
3.3.3 CONTEXT-CENTERED INFORMATION RETRIEVAL.....	25

3.4 EXTRACTING REPRESENTATIVE CONTEXT	25
3.5 ESTIMATION OF THE SEMANTIC IMPORTANCE OF TERMS	26
3.6 PROPOSAL FOR A CLUSTERING TECHNIQUE	29
3.7 CHAPTER SUMMARY	30
CHAPTER 4	31
CONTEXT-AWARE IR: THE DROPT TECHNIQUE	31
4.1 INTRODUCTION	31
4.2 PARAMETERS USED FOR RANKING PRINCIPLES.....	31
4.2.1 THE STATEMENT FORMULATION.....	33
4.3. DROPT TECHNIQUE.....	33
4.3.1 FORMALIZATION OF MATHEMATICAL MODEL DEFINITIONS.....	36
4.4 EVALUATION APPROACHES FOR CONTEXT-AWARE IR	40
4.4.1 EVALUATION OF THE RELEVANCE BY THE USER’S JUDGMENT	40
4.4.2 EVALUATION OF THE RELEVANCE COMPARED TO THE QUERY	41
4.4.3 EVALUATION OF PERFORMANCE OF THE SEARCH TOOL.....	41
4.5 CHAPTER SUMMARY	41
CHAPTER 5	42
SYSTEM DESIGN AND IMPLEMENTATION.....	42
5.1 REQUIREMENTS ANALYSIS.....	42
5.2. REQUIREMENTS DETERMINATION	42
5.3 DESIGN AND ARCHITECTURE.....	45
5.3.1 THE PROPOSED SYSTEM ARCHITECTURE.....	46
5.4 A CONTEXT PERSONALIZED INFORMATION RETRIEVAL MODEL	48
5.5 PROPOSED SYSTEM DESIGN.....	51
5.5.1 PREFERENCE RELEVANCE FEEDBACK OF USER JUDGMENTS ON DOCUMENTS	54
5.6 SEQUENTIAL DOCUMENT RANKING PERSONALIZATION.....	56
5.7 EXPERIMENTAL DESIGN	57
5.8 IMPLEMENTATION	58
5.8.1 Results.....	63
5.8.2 Discussion.....	70
5.9 CHAPTER SUMMARY	70
CHAPTER 6	71
SYSTEM EVALUATION	71
6.1 EVALUATION METHODOLOGY	71
6.2 EVALUATION METRICS.....	72
6.3 RANKING PERFORMANCE RESULTS	73
6.4 RETRIEVAL RESULTS	77
6.5 EXPERIMENTAL RESULTS OF DROPT TECHNIQUE.....	84
6.5 PERSONALIZING SEARCH RESULTS	85
6.6 STATISTICAL ANALYSIS.....	87

CHAPTER 7	91
CONCLUSION AND FUTURE WORK	91
7.1 CONCLUSION	91
7.2 FUTURE WORK	95
REFERENCES	97



Chapter 1

Introduction

1.1 Background

Recent years have witnessed ever-growing amount of online information. The development of the World Wide Web (WWW) led to increase in the volume and diversity of accessible information. The question that now arises is how access to this information can be effectively supported. Users require the assistance of tools aimed to locate documents that satisfy their specific needs. Information retrieval (IR) concerns searching documents for information that meet a user need. It is also concerned with the representation, storage, organization of, and access to information items that make retrieving information an easy and beneficial task [Baeza-Yates and Ribeiro-Neto 1999]. Traditionally, document representations are expressed by extracting meaningful keywords (index terms) from the documents. This set of keywords provides a logical view of the documents. When the user sends a search request, a representation of his/her information need will also be expressed in the same manner. Then the user query (request representation) and the representation of the document will be matched according to specific matching conditions (rules). Results are presented to the user in a form of a ranked list that contains the most relevant documents. Most of the documents that are retrieved however are irrelevant to the user because search engines cannot determine the user context. Diverse IR models have been developed for this purpose. Context-based IR systems are based on user models that describe the user's interest using commonly used terms in a specific domain [Zhou et al. 2012].

Spink and Cole [2005] argued that taking context into account is vital when solving IR tasks in order to produce insightful results and eventually cognitive-enabled IR. Context can be employed from the dimension of user's prior knowledge [Li et al. 2011], or user's interest [Chevalier et al. 2011]. A context-based system adapts the search results to the user's context to capture a specific information need [Islam et al. 2013 and Asfari 2009]. The main motivation of context-based retrieval systems is that users often fail to accurately represent their information need using query reformulation prediction [Ercan and Cicekli 2012], which

often lead to ambiguous queries [Gupta et al. 2013; Song et al. 2009]. Steichen *et al.* [2012] surveyed diverse personalisation IR techniques. They conclude that most existing IR systems base their retrieval judgment solely on query representation and document collections but, information about actual users and search context is largely ignored.

Ideally, the relevance of documents should be defined based on user context. Thus, the problem of ranking of retrieved documents should be based on user context and preferences. Relevance is a standard measure utilized in IR to evaluate effectiveness of an IR system based on the documents retrieved. The effectiveness of an IR system is determined primarily by the relevance assessment of the retrieved information [Setchi *et al.* 2011; Saracevic 2007; Borlund 2003]. The concept of relevance, however, is one that is subjective and influenced by diverse factors. To this end, user perception and user knowledge level are factors that influence the relevance of a retrieved document. Therefore, there has been a paradigm shift from a view of relevance as simple term matching between query and document, to a view of relevance as a cognitive and dynamic process involving interaction between the information user and the information source.

It is important for IR systems to obtain accurate representations of users' information needs and the context of information need. Context-based systems attempt to take into account factors and tailor various aspects of the search knowledge to individual users. There are many different ways to personalize IR systems, with respect to the particular aspects of search knowledge and different information sources. Search knowledge encompasses a wide variety of aspects of the search, such as the interaction mode by users.

One of the lessons learnt over the years, is that it is very difficult to achieve effective personalization solutions, without having considerable knowledge about the particular problem being addressed. Personalization approaches result in very specialized solutions that provide very limited personalization capabilities [Zhao *et al.* 2008]. This is because automatic personalization techniques are typically applied out of context. While users may have stable and recurrent overall preferences, not all of their interests are relevant all the time. In order to address some of the limitations of these personalization systems, researchers have examined a new emerging area, so-called context-awareness [Campos *et al.* 2013; Baltrunas *et al.* 2012; Adomavicious and Bamshad 2011; Yu and Jeon 2010].

Context-awareness is the ability of an entity to be aware of the conditions under which it operating current situations, and use the information to perform tasks. Context-awareness has

been acknowledged to be effectively functional in a wide range of fields, including mobile and pervasive computing [Emmanouilidis *et al.* 2013; Noh *et al.* 2012], computational linguistics [Glushko *et al.* 2013], and IR [Carrevas and Botia 2013; Steichen *et al.* 2012; Dumitrescu and Santini 2009]. Context has also been applied to a wide variety of applications, ranging from physical user location [Melucci 2005] to desktop information [Saparova *et al.* 2013; Sease 2008; Dumais *et al.* 2003], and visited Web pages [Palomino *et al.* 2013 and Sugiyama *et al.* 2004]. Knowing more about what features are important in a context and what they are used for, can help design more beneficial and successful IR systems. The idea of context personalization, relates to the fact that human preferences are multiple, heterogeneous, changing, even contradictory, and should be understood with the user goals in mind. Aiming to address the discrepancies, the question how search can affect the information seeker's interaction with IR system, his expectations and judgments about retrieved documents can be supported effectively restricting by notion of context-awareness.

1.2 Statement of the problem

Context is a common notion in IR. This is not surprising since it is known that the relevance of information is strongly dependent on context. The term context and context-awareness, denotes a general class of systems that can sense a continuously changing physical environment and provide relevant services to users on this basis [Dey 2001]. Based on this fundamental definition, various authors [Emmanouilidis *et al.* 2013; Jara *et al.* 2013; Noh *et al.* 2012; Xue and Deng 2012] focus on different aspects of context-awareness, including modelling interactions between users and IR systems nature, and how to modelling context. The research reported in [Nyongesa and Maleki-dizaji 2006] showed that based on preferences of users, genetic algorithms (GA) could be applied to improve the search results. Similarly, the work reported in [Koorangi and Zamanifar 2007] proposed improvement of internet engines using multi-agent systems. In this work, a meta-search engine gives a user documents based on an initial query while a feedback mechanism returns to the meta-search engine the user's suggestions about retrieved documents. This leads to a new query formed using a Genetic Algorithm (GA).

From these previous studies, two aspects emerge: context representation, and document ranking. This research then addresses the following:

a. How a combination of relevance feedback, interactive reinforcement learning and context-awareness can be applied to improve IR effectiveness?

In this respect, qualitative feedback from users is combined with a fitness measure of competing models of information needs. We also apply context-awareness to reformulate queries in order to improve the predicted relevance of retrieved documents.

b. How can retrieved information be ranked with regards to the context of the information seeker?

We propose a technique to quantify the context of retrieved information. The technique aims to avoid the drawback of manually scanning through and selecting from a long list of documents.

1.3 Motivation for the research

The emergent growth of the WWW has necessitated a need for tools that address problems associated with access to vast information sources. In many situations the information seeking experience is less than satisfactory and often searchers have difficulty finding relevant information from the huge number of information sources that has not matched this rapid growth. The state-of-the-art tools are often ineffective for all but the most simple search tasks. Often users need to refine the search query several times and search through large document collections to find relevant information.

1.4 Research objectives

Context-awareness is proposed as a technique that can be employed to reformulate queries in order to improve the predicted relevance of retrieved documents. Context-awareness creates a user profile through their interaction with IR systems. Thus, IR systems learn how to use user interaction to adapt information seeking context. The following are the research goals:

- To develop algorithms that optimize the ranking of documents retrieved from search engines.
- To build user profile models through interaction with IR systems.
- To rank document relevance in accordance with user profile.
- To improve the effectiveness of IR systems using evolutionary algorithms.
- To build a system capable of modelling evolving user information needs.

1.5 Research methodology

The research methodology followed in the study is as follows:

- Review of state-of-art literature on context-awareness, and context-aware IR systems.
- Conceptualization of context-aware IR system.
- Conceptual design of a context-aware IR system.
- Proposal for a framework for context-aware IR system.
- Architectural design of a context-aware IR system.
- Develop an algorithm for context-aware information retrieval.
- Develop models for user profiling.
- Design, develop, implement and test a "prototype" context-aware IR system.
- Validate the proposed system using test cases.

1.6 Thesis organization

The rest of the thesis is organized as follows.

Chapter Two gives an overview of the principles of IR, related work on context-aware information retrieval, context modelling, reviews of existing IR systems and approaches for evaluation of IR systems.

Chapter Three reviews information techniques and proposes a clustering algorithm for context aware information.

Chapter Four introduces and formalizes the DROPT algorithm according to information relevance.

Chapter Five presents the design and implementation of the proposed prototype IR system.

Chapter Six discusses an evaluation of the proposed system and the developed algorithm. The experimental results include the comparison of the retrieval efficiency of different prototype implementations.

Finally, Chapter Seven presents a discussion of the results Conclusion, Significance and Contribution of the research, and proposal for future work.

Chapter 2

A Review of IR and Related Techniques

2.1 Introduction

Information Retrieval (IR) has been a well-established discipline in Computer Science since the 1950s. It has however recently enjoyed increased significance because of the information explosion caused by the WWW and its related technologies. Not only the absolute amount of information, but also new types of information formats have drawn attention to this field [Lally 2006]. While IR used to be a restricted field with specialized users like librarians and information professionals, today millions of people use IR every day to search the web or search their email, resulting in the need for new user interfaces and query languages [Manning *et al.* 2008; Hearst 2011; Danica *et al.* 2013].

According to [Baeza-Yates and Ribeiro-Neto 1999], IR *"deals with the representation, storage, organization of, and access to information items."* While this is a very broad and generic definition over a more precise definition of IR as a field of academic study according to is: *"IR is finding material (e.g. documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)."*

The most common task in IR is informally ad hoc retrieval: a user expresses an information need by submitting a query to the system, which tries to return documents relevant to this query. Other tasks in IR include support of users in browsing or filtering document collections, text classification, text clustering, cross-language retrieval, and multimedia retrieval [Manning *et al.* 2008; Lew *et al.* 2006; Roul and Sahay 2012].

- *Documents* are the basic information items that IR systems operate on. While traditional IR mostly dealt with text documents, modern IR deals with such diverse items as semi-structured, multimedia, and hypertext documents.
- A *corpus or document collection* is a set of documents.
- *Index terms* are keywords, fragments of words, or phrases that are used to describe the content of a document.

- A *vocabulary* is a set of all keywords.

2.2 The information retrieval process

The overview of the IR process is presented in the abstract schema in figure 2.1. The schema serves as basis for the discussion of the retrieval process and its components. The information process requires a collection which is indexed. In typical applications documents are not held by the IR system, but rather representations of the documents. Documents undergo a series of pre-processing operations to obtain this representation. The other side of the process is represented by the user, who has a certain information need that has to be satisfied. An information need is often expressed as a set of keywords. To allow for correct matching, the query is usually treated with the same operations used for indexing of documents. The IR system matches the keywords against the document index in order to retrieve matching documents. The system then ranks the documents applying some algorithm that measures the similarity between the query and the document representation.

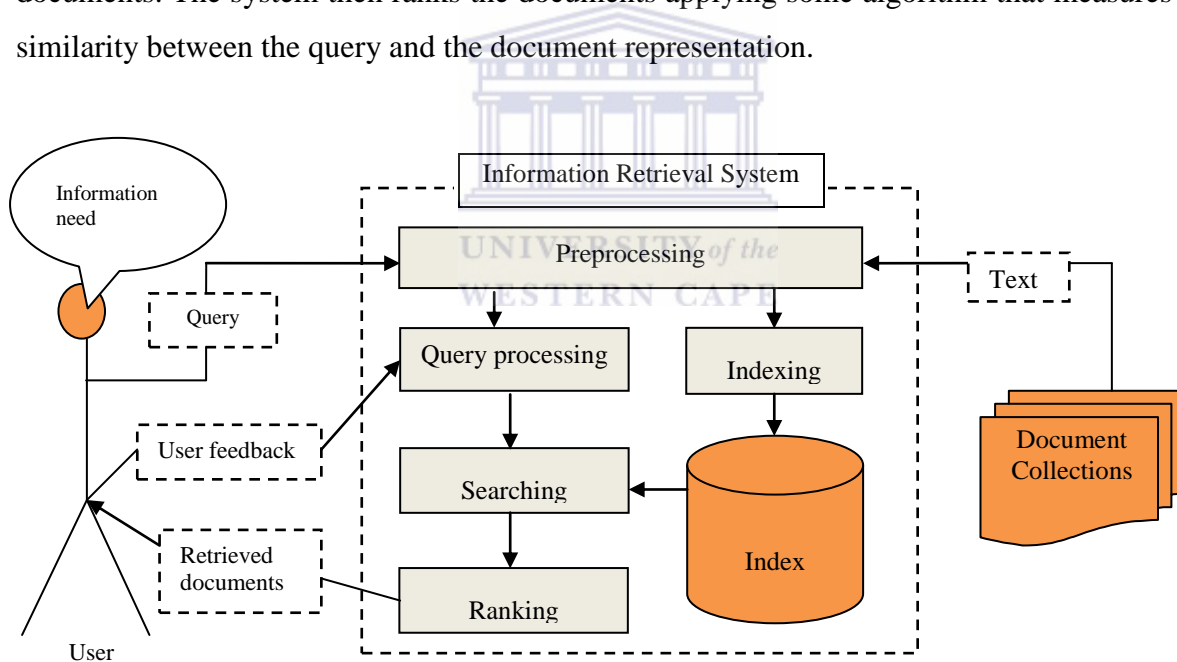


Figure 2-1: The Information Retrieval Process adapted from [Belew 2000].

The length of the individual documents, so-called indexing granularity, depends on the given collection and can range from a few single sentences or paragraph, to large file. The preparatory phase can involve cleaning the collection, which means removing unnecessary documents, duplicates, and other documents that should not be indexed. Belew [2000] for example mentions filters that remove unnecessary syntactical and structural information such as formatting mark-up.

After the document preparation has been carried out, the individual documents can be treated as a stream of characters. This stream has to be transformed into terms, before the actual indexing can occur. As transformations may make it harder for the user to interpret the results of the retrieval process, some modern IR systems such as web search engines try to avoid extensive pre-processing [Baeza-Yates and Ribeiro-Neto 1999]. Despite this trend, document pre-processing still plays an important role in many IR systems today. Operations typically involved in this process are described in the following.

Tokenization is the process of breaking the stream of characters into keywords pieces called tokens. Even though keywords and tokens appear similar, tokens have to undergo a set of transformations before they become terms that are indexed. Different strategies exist for deriving tokens, ranging from splitting on whitespace and removing all punctuation, to more sophisticated solutions that make numeric, hyphenation, and domain-specific aspects into account. The same operations are also to be carried out for queries submitted by the user, in order to guarantee correct matching.

Token normalization allows for matches to occur even in the presence of superficial differences between queries and index terms. The two most common forms of token normalization are case-folding, the conversion of all uppercase letters into lowercase or vice-versa, and the removal of accents, diacritics, and other peculiarities related to specific languages [Manning *et al.* 2008].

Words that appear too frequently in a document corpus are not very helpful for matching. These words are known as stop words and are often filtered out [Baeza-Yates and Ribeiro-Neto 1999]. Typical stop words are articles, prepositions, and conjunctions; however, this varies from domain to domain. Besides speeding up query processing, an important benefit of discarding stop words is a reduction of index size.

Stemming is another transformation that reduces words to their morphological roots by stripping off suffixes and other modifiers [Witten *et al.* 1999]. Lemmatization has the same goal, but instead of using heuristics as in stemming, it takes advantage of vocabularies and morphological analysis to reduce words to their base or dictionary form known as lemma. The main idea behind both methods is that the retrieval performance can be improved by collapsing variants of a keyword that would otherwise be treated independently. Searches for the plural form of a term, for example, also yield documents containing only the singular form of the term and vice-versa.

2.2.1 Indexing and searching

During document preparation, the raw corpus is processed into a set of individually retrievable documents. These raw documents are transformed into index-able terms. Indexing stores the mapping between keywords terms and documents in a data structure that allows for fast lookup. Even though there are several possible structures for indices, the most suitable one for text applications is the so-called inverted index [Witten *et al.* 1999].

In an inverted index, terms are kept in a dictionary, sometimes also referred to as a lexicon. For each term, a posting list or inverted list stores references to all documents the keywords occur in. Thus, an inverted index, much like an index in a book, maps terms to documents or document parts in which they occur. Each record consists of an inverted key value and a string of elements which identify those main file records which contain the cited key. Inverted index records are alternatively referred to as ‘associative key lists’. In addition to the key lists, a key directory is usually maintained to provide the start of key list address, given the attribute value form of the inverted key.

The size of the index largely depends on its granularity [Witten *et al.* 1999]. While in a non-positional, index only the document in which the term occurs is stored in the posting list, in a positional index; the location of the term in the document is specified as well.

The process of creating an inverted index takes a list of normalized tokens for each document as input. The most important step in index construction is the sorting and grouping of the terms. In the simplest case, terms are sorted alphabetically and multiple occurrences of the same term in a document are merged. Instances of the same term across documents are then grouped together, and the resulting list of terms and their occurrences is split into dictionary and postings. In addition to a pointer to the posting list, each term in the dictionary can contain certain pieces of statistical information such as document frequency. The postings are then sorted by document number to allow efficient query processing. The dictionary file is much smaller than the postings file and is usually kept in memory to optimize response time.

Processing a query over an inverted index can be divided into three general steps [Baeza-Yates and Ribeiro-Neto 1999]. First, the individual words and patterns in the query are isolated and looked up in the dictionary. Second, the posting list for each match is retrieved and decoded. Third, the occurrences are processed for different query operations such as Boolean, phrase and proximity operations. Simple Boolean queries can be carried out by

merging the retrieved posting lists using the intersection, union, or complement [Witten *et al.* 1999]. The same general merge technique is used to process phrase or proximity queries. However, in addition to checking for the presence of the terms in a document, their relative position is taken into account as well [Manning *et al.* 2008].

2.2.2 Ranking

Ranking is the process of ordering the results of a query according to measures of relevance to the user. Although not essential to IR, ranking can greatly simplify the interaction with large document collections and is employed widely, especially in the context of web search. Shen *et al* [2012] proposed a ranking technique for multi-search projections on the Web for results aggregation model based on query words, search results, and search history to achieve user's intention. To this end the Web can offer a rich context of information which can be expressed through the relevancy of document contents. Shivaswamy and Joachims [2011] proposed a model for online learning that is specifically adequate for user feedback. The experiment conducted shown retrieval effectiveness for web search ranking. In the context of web search ranking, these techniques aim at finding the best ordering function over the returned documents is important. The authors argue that, regression on labels may be adequate and, indeed, competitive in the case of large numbers of retrievals. To make the web more interesting, there is need to develop a good and efficient ranking algorithm to deliver more suitable results for users.

The need for query operations arises from the user's difficulty to formulate queries without a full understanding of the underlying collection and the IR environment [Baeza-Yates and Ribeiro-Neto 1999]. Over the years, various techniques to deal with this problem have been proposed. [Manning *et al.* 2008] divide these techniques into two broad categories: global methods that use information independent of the query, and local methods that adjust a query relative to the documents that initially seem to match it. Global methods use a thesaurus to expand or reformulate a query with similar terms. Thesauri can be generated either manually or automatically by leveraging word co-occurrences or grammatical analysis. Local methods are normally based on relevance feedback (RF). Relevance feedback requires the user to assess documents returned for an initial query as either relevant or non-relevant. Based on this feedback, the system then tries to compute a more accurate representation of the user's information need and returns a new result set. These actions can be carried out iteratively, forming a feedback cycle. Two modifications of RF have been developed that are not based

on interactive feedback from the user. Pseudo-relevance feedback builds on the assumption that the top-ranking documents for a query are relevant and thus uses them for relevance feedback. Implicit relevance feedback, on the other hand, re-ranks documents based on implicit relevance judgments. A user's click on a document in the result list can for instance be interpreted as such an implicit relevance statement.

2.3 Information retrieval and its evaluation

Information retrieval (IR) is the key technology for knowledge management which guarantees access to large corpora of unstructured data. Very often, text collections need to be processed by retrieval systems. IR is the basic technology behind Web search engines and an everyday technology for many Web users. IR deals with the storage and representation of knowledge and the retrieval of information relevant to a specific user problem. IR systems respond to queries which are typically composed of a few words taken from a natural language. The query is compared to document representations which were extracted during the indexing phase. The most similar documents are presented to the users who can evaluate the relevance with respect to their information needs and problems.

In the 1960s, automatic indexing methods for texts were developed. They implemented the bag-of-words approach at an early stage, and this still prevails today. Although automatic indexing is widely used today; many information providers and even internet services still rely on human information work. In the 1970s, research shifted its interest to partial match retrieval models and proved superior compared to Boolean retrieval models. Vector space and later probabilistic retrieval models were developed. However, it took until the 1990s for partial match models to succeed in the market. The Internet accelerated this development. All Web search engines were based on partial match models and provided results as ranked lists rather than unordered sets of documents. Consumers got used to this kind of search system and eventually all big search engines included partial match functionality. The growing amount of machine-readable documents available requires more powerful IR systems for different applications and user needs.

The evaluation of IR systems is a tedious task. Evidently, a good system should satisfy the needs of a user. However, the users' satisfaction requires good performance in several dimensions. The quality and relevance of the results with respect to the information need, system speed and the user interface are major dimensions. To make things more difficult, the

most important dimension, the level to which the search result documents help the user to solve the information need, is very difficult to evaluate. User-oriented evaluation is extremely difficult and requires many resources. In order to evaluate the individual aspects of searches and the subjectivity of user judgments regarding the usefulness of searches, an impracticable effort would be necessary.

As a consequence, information retrieval evaluation experiments try to evaluate only the system. In order to calculate performance measures, a test collection consisting of three parts is required: a document collection, a set of information needs transformable into queries and a set of relevance judgments for each query-document pair [Manning et al. 2008; Efron 2009; Samini and Ravana; 2014]. The user is an abstraction and not a real user. In order to achieve that, the users are replaced by objective experts who judge the relevance of a document to one information need. This evaluation methodology is called the Cranfield paradigm, based on the first information retrieval system evaluation in the 1960's [Cleverdon1997]. This is still the evaluation model for modern evaluation schemes. The first main modern evaluation scheme was the Text Retrieval Conference (TREC). TREC had a huge impact on the field. The emphasis on evaluation in IR research was strengthened. System development and the exchange of ideas was fostered by TREC and systems greatly improved in the first few years. Recent evaluation efforts try to keep their work relevant for the real world and make their results interesting for practical applications. In order to cope with these new heterogeneous requirements and to account for the changing necessities of different domains and information needs, new approaches and tasks need to be established. A measure of the effectiveness of the search, a test collection, and a test of statistically significant between methods are the major rudiments of a meaningful IR experiments.

2.3.1 Information retrieval evaluation approaches

The first criterion is concerned with the different quantitative and qualitative metrics used for evaluation. The metrics of importance in this study is described as follows:

1. *Retrieval effectiveness* can be quantitatively measured in a number of ways using well-known metrics in the IR community [Baeza-Yates and Ribeiro-Neto 2011; Manning et al. 2008]: (i) *Precision*, which is the number of retrieved relevant documents over the total number of retrieved documents; (ii) *Recall*, which is the number of relevant documents that are retrieved over the total number of known relevant documents in the document collection;

(iii) *Precision at K*, which measures the fraction of retrieved relevant documents within the top K retrieved documents; (iv) *Recall at K*, which measures the fraction of retrieved relevant documents within the top K documents over the total number of relevant documents in the document collection; (v) *Mean Average Precision (MAP)*, which is a single-valued metric that serves as an overall figure for directly comparing different retrieval systems. It is the average Precision at K values computed after each relevant document has been retrieved for a query, where the mean of all these averages is calculated across all the test queries; (vi) *Normalised discounted Cumulative Gain (NDCG)*, which is a precision metric that is designed for experiments where documents are judged using a non-binary relevance scale (e.g. highly relevant, relevant, or not relevant). It gives higher scores for more relevant documents being ranked higher in the ranked list of results; (vii) *R-precision*, which measures precision with respect to a given number of documents that are known to be relevant;

2. In PIR, two kinds of datasets are used in the second criterion: document collections and search logs. Document collections (corpora) are datasets that comprise a large number of documents in one or more languages. Examples of these are the collections provided by TREC, CLEF, and NTCIR, which are widely used in the IR community. These collections, together with a set of manually selected information needs, are used as a test-bed for comparing retrieval and adaptation algorithms developed by researchers in the community. Not all experiments in PIR are conducted on standard test collections; several experiments were conducted on open Web corpora using retrieval components that are wrapped around live Web search engines. The advantage of this approach, over the use of standard test collections, is that the experiments are not over-fitted on the domain or characteristics of a specific test collection. However, the disadvantage of this approach is that it becomes hard to perform apples-to-apples comparisons between the results of different studies in the literature. Search logs, are datasets that comprise the history of user interactions with a system over a period of time. Search logs serve a very important role in PIR experiments since they hold usage information (aggregate or per user) which is a crucial element in search personalisation. When this information is analysed and represented in user models it becomes the basis of user-focused adaptation algorithms. Larger datasets of search logs can contribute towards more reliable results.

2.4 Performance evaluation

Ranking search results is a fundamental problem in IR. Most common search personalization approaches in the context of the web use both the result re-ranking and result scoring for implementation and compared results to each other [Agichtein et al. 2006a]. The authors showed that result scoring approach was more effective, thus they recommended performing personalization by result scoring rather than by result re-ranking. In this research study, the result scoring approach is adapted as our pilot guide. However, with increasing popularity of search engines, implicit relevance feedback (i.e. the actions users take when interacting with the search engine) and information collected from the user explicitly for example by asking for feedback such as preferences can be used to improve the rankings. The research proposed a Document Ranking OPTimization (DROPT) technique for solving an IR problem. The problem is capability to retrieve from a search engine only those documents that are relevant to a user's information needs and rank them at the top of the list, rejecting documents that are irrelevant. The system was tested for two different learning techniques, namely relevance feedback (RF) adopting result scoring approach to score the documents according to users' need for result adaptation and interactive reinforcement learning (IRL), which is a combination of user's feedback and context awareness. This is desired so that the relevance of the returned document can be adapted to individual users. In both techniques context awareness can be utilized for preference user feedback. The retrieval effectiveness and the proposed ranking performance technique experiments will be carried out. Generally, the effectiveness of a traditional IR model is evaluated using the well-known recall and precision metrics that can allow measuring its ability to select relevant documents at the top. We outline, however, that they focus on topical relevance which is user independent. According to this view, a laboratory evaluation model has been proposed through TREC that provides data collections (document, queries and judgments) allowing comparative evaluation of algorithms, models and techniques in IR). This standard evaluation approach is studied for evaluating our model.

2.5 Context-aware information: A review

In Allan [2002], contextual information retrieval (CIR) is defined as:

"combine search technologies and knowledge about query and user context into a single framework in order to provide the most appropriate answer for user's information needs".

CIR intends to optimize the retrieval accuracy by involving two related steps: appropriately defining the context of user information needs, commonly called *search context*, and then adapting the search by taking it into account in the information selection process.

One of the primary questions is: which facets of context should be considered in the retrieval process. Several studies have addressed context specification within and across application domains [Jara *et al.* 2013; Dinh and Tamine 2012; Kebler 2009; Goker *et al.* 2008; Vieira *et al.* 2007]. Device, user, task, document and spatio-temporal are the five context specific dimensions that have been explored in context-based information retrieval literature [Li *et al.* 2012; Asfari *et al.* 2011; Mylonas *et al.* 2008; Anand and Mobasher 2007; Emmanouilidis *et al.* 2013; Marco *et al.* 2013; Lukowicz *et al.* 2012; Zhoul *et al.* 2012; Kebler 2009]. Figure 2-2 shows the five context specific dimensions.

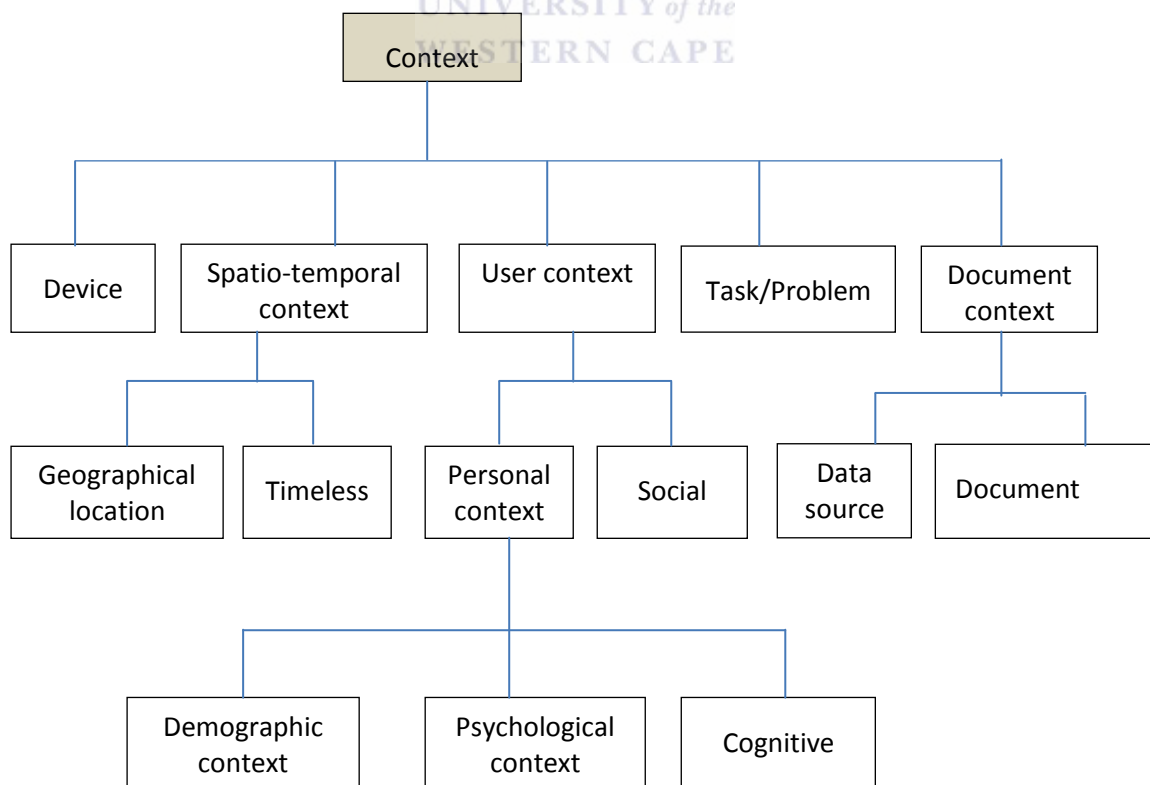


Figure 2-2: The Multi-faceted Concepts of Contextual IR

A review of context-aware IR in mobile environments show that contextual IR aims to tackle the problem of information overload by providing appropriate results according to the resource constraints in one hand and user's location, time and interests on the other hand. In the same essence with personalized IR, contextual retrieval is achieved by exploiting the mobile context during query reformulation and document re-ranking steps. For example, in [Emmanouilidis *et al.* 2013], the authors combine situation-based adaptation and profile-based personalization into the IR model.

2.5.1 The framework for contextual information retrieval

An information system is context-aware if it exploits context data in order to deliver relevant information to the user. CIR aims at optimizing the retrieval accuracy by involving two related steps: appropriately defining the context of user information needs, commonly called *search context*, and then adapting the search by taking it into account in the information selection process. New search services that incorporate context, and further incorporation of context into existing search services, may increase the retrieval effectiveness, and help mitigate any negative effects of biases in access to information on the Web [Bhatia and Kumar 2008a]. The CIR paradigm has the primary goal to acquire a user's information seeking behaviour, such as their search activities and responses, and incorporate this information into a search system.

CIR aims at delivering the right information to the user, in response to his query, within the right context. Numerous approaches - employing contextual user profiles, concept-based query formulation and relevance filtration and relevance feedback/suggestion - already exist today. Previous studies in the area of CIR has focused on three main themes, namely, ***User Profile Modelling*** [Gladun *et al.*, 2013; Steichen *et al.* 2012; Agosto 2012; Speretta and Gauch, 2005] ***Query Expansion*** [Chellatamilan and Suresh 2013; Carpineto and Romano 2012; Dinh and Tamine 2012; Bouramoul and Kholadi 2010; Asfari *et al.* 2010] and ***Relevance Feedback*** [Gupta *et al.* 2013; Belhajjame *et al.* 2011; Jones *et al.* 2011; Tamine *et al.* 2010]. Figure 2-3 presents the basic architecture of a context-aware called also Contextual Information Retrieval system.

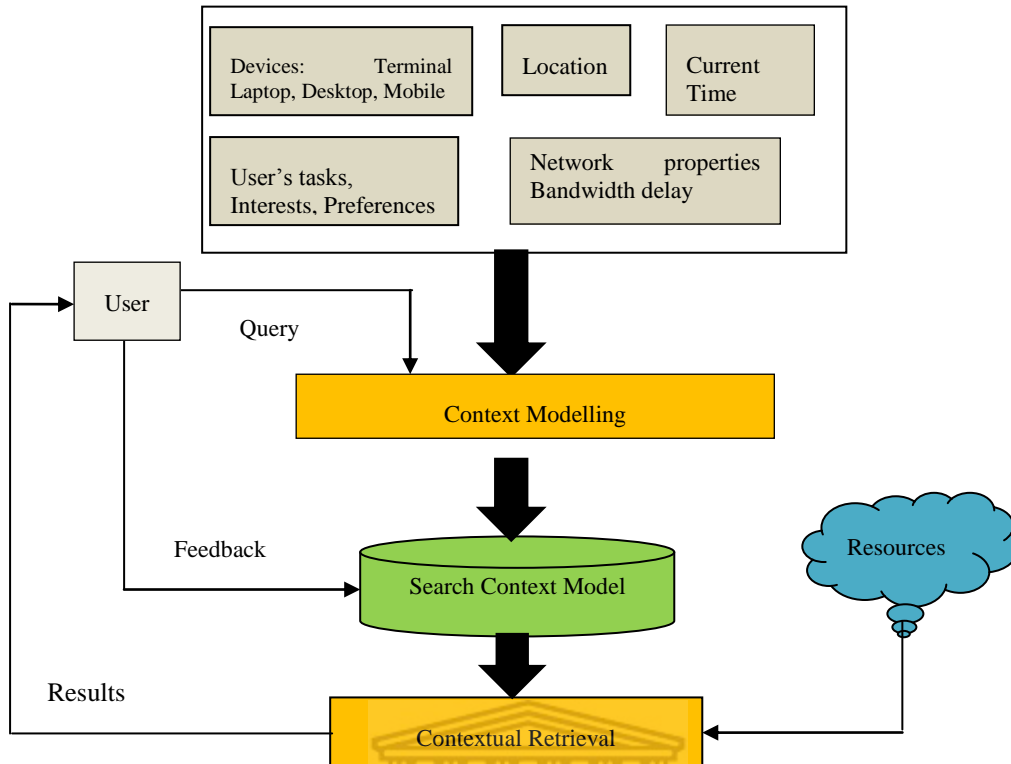
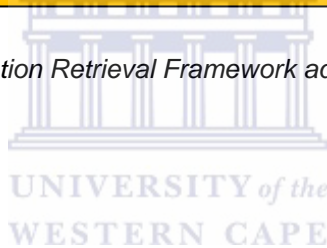


Figure 2-3: Contextual Information Retrieval Framework adapted from [Tamime et al 2010]



2.5.1.1 User profile modelling

Research works have focused on exploiting the sources of evidence that more precisely include approaches to build the user profile that allow learning user's context by implicitly inferring the information from the user's behaviour and from external or local context sources. Several pertinent studies on Web IR systems have examined various user modelling approaches to improve the personalization of a users' Web search experience. Steichen *et al.* [2012] also personalizes the user's IR system content. The user profile, learned from the documents in the user's system, is composed of adaptive hypermedia terms. Gupta *et al.* [2013] constructed an approach based on user profile to reformulate a query as refinement process which integrates elements of the user profile into the user query. A review of these user modelling approaches reveals that in order to construct a contextual profile these techniques utilize either user behaviour or preferences. However, none of the approaches have used a combination of user behaviour and preferences and do not have the capability to share a user's contextual profile information with other users, thereby potentially leading to

suboptimal performance when the user needs access to information outside their original context (with an exception of WebMate) [Chen and Sycara 1998]. While showing promise, prior conventional IR approaches employing user profile modelling have had limited success. Fundamental challenges remain, specifically:

- How to acquire, maintain and represent information about a user's interests with minimal intervention?
- How to deliver personalized search results using the user information acquired?
- How to use information acquired from various users as a knowledge base for interest communities or groups?



Chapter 3

Clustering Algorithms for Context-Awareness

3.1 Introduction

Searching for useful nuggets among huge amounts of data has become known as the field of data mining. Data mining can be applied to relational, transaction, and spatial databases, as well as large stores of unstructured data such as the World Wide Web (WWW). Data mining is an exploratory activity, in which clustering techniques are often applied. Clustering is an important initial step in the data mining process. In information retrieval documents are clustered on the basis of the information that they contain. Clustering has also been used on retrieval documents to provide structure to retrieved documents. Roul and Sahay [2012] used an effective clustering technique to aggregate clusters of cited Web documents by successively linking together all selected pairs of cited Web documents that have at least one Web cited document in common. Context clusters are weighted lexical chains that represent aspects of the meaning of a document and express the semantic importance within the document. Lexical chains have two-fold significance for computational understanding of text. First, they determine the context of the discourse within a document. Second, they provide clues about the topicality of a document. Conversely, in order to estimate the semantic importance of terms within a document, the representative context clusters need to be identified. To achieve this, two weight functions are defined; one for each context cluster and the other for terms within the cluster. Context clusters, thus, quantify and preserve the relevant information within a document.

3.2 In Other Approaches

Clustering has a long history starting from a statistical pattern recognition viewpoint, Jain *et al.* [1999] reviewed clustering algorithms and other issues related to cluster analysis. Hansen and Jaumard [1997] described clustering problems under a mathematical programming scheme. In another approaches, Kolatch [2001] and He [1999] investigated applications of clustering algorithms for spatial database systems and IR, respectively. Clustering has also

been used on retrieval documents to provide structure to retrieved documents, such as co-citation analysis [Croft 1997]. Berkhin [2001] expanded the topic to the general field of data mining. Rauber *et al.* [2000], presented empirical results for five typical clustering algorithms. Wei *et al.* [2000] placed emphasis on the comparison of fast algorithms for large databases. Steinbach *et al.* [2000] reviewed applications and experimental evaluations on document clustering techniques, based on hierarchical and k-means clustering algorithms.

Some data mining approaches which use clustering are database segmentation [Wang and Fan 2010; McCarty and Hastak 2007], predictive modeling [Bellazzi and Zapan 2008], and visualization of large databases [AbdulRaham and AbdulAziz 2012]. Clustering algorithms have been used in a large variety of applications. These include IR [Bordogna and Pasi 2012; Bordogna and Posi 2011; Fet *et al.* 2007], data mining [Padmapriya and Sabitha 2013; Luo *et al.* 2009], character recognition [Yousri *et al.* 2008; Nafiz and Yarman-Vural; 2001] image segmentation [Chaira 2011; Yang 2009; Yang and Huang 2007], object recognition [Awad 2012]. Text clustering algorithms partition document into distinct groups or categories. Everitt *et al.* [2001] argued that there is no universally agreed upon definition of what constitutes a text cluster. Some researchers [Hansen and Jaumard 1997; Jain and Dubes 1988] describe a cluster by considering their internal homogeneity and external separation. That is, patterns in the same cluster should be similar to each other, while patterns in different clusters should likewise be dissimilar.

Diverse starting points and criteria usually lead to different taxonomies of clustering algorithms. A rough but widely agreed framework is to classify clustering techniques as either hierarchical clustering or partitional clustering, based on the properties of clusters generated [Everitt 2001; Jain *et al.* 1999].

3.2.1 Hierarchical clustering techniques

Hierarchical clustering (HC) algorithms organize data into a structure according to a proximity matrix. The results of HC are usually depicted by a binary tree or dendrogram. The root node of the binary tree represents the whole data set and each leaf node is regarded as a data object. The intermediate nodes, thus, describe the degree that the objects are proximal to each other; and the height of the binary tree usually expresses the distance between each pair of objects or clusters, or an object and a cluster. The ultimate clustering results can be obtained by cutting the binary tree at different levels. HC algorithms are mainly classified as agglomerative techniques and divisive techniques. Based on the different definitions for

distance between two clusters, there are many agglomerative clustering algorithms. The simplest and most popular methods include single linkage and complete linkage techniques.

The common criticism for classical hierarchical clustering algorithms is that they lack robustness and are, hence, sensitive to noise and outliers. Once an object is assigned to a cluster, it will not be considered again, which means that HC algorithms are not capable of correcting possible previous misclassification.

In recent years, with the requirement for handling large-scale data sets in data mining and other fields, many new hierarchical clustering techniques have appeared and greatly improved the clustering performance. Typical examples include CURE [Guha *et al.* 1998], ROCK [Guha *et al.* 2000], BIRCH [Zhang *et al.* 1996] and RHC [Mollineda and Vidal 2000]. Guha *et al.* [2000] also proposed another agglomerative HC algorithm, ROCK (Robust Clustering using linKs), to group data with qualitative attributes. In their approach, they used a novel measure “link” to describe the relation between a pair of objects and their common neighbors. Like CURE, a random sample strategy is used to handle large data sets.

RHC (Relative hierarchical clustering) [Mollineda and Vidal 2000] is another exploration that considers both the internal distance (distance between a pair of clusters which may be merged to yield a new cluster) and the external distance (distance from the two clusters to the rest), and uses the ratio of them to decide the proximities. Liand Biswas [2002] extended agglomerative HC to deal with both numeric and nominal data.

3.2.2 Evolutionary search-based clustering

Evolutionary approaches, motivated by natural evolution, make use of evolutionary operators and a population of solutions to obtain the globally optimal partition of the data. Candidate solutions to the clustering problem are encoded as chromosomes. The most commonly used evolutionary operators are: selection, recombination, and mutation. Each transforms one or more input chromosomes into one or more output chromosomes. A fitness function evaluated on a chromosome determines a chromosome’s likelihood of surviving into the next generation. Clustering can be regarded as a category of optimization problems. Given a set of data points, clustering algorithms aim to organize them into subsets that optimize some criterion function. Hall *et al.* [1999] proposed a GA that can be regarded as a general scheme for center-based (hard or fuzzy) clustering problems. Fitness functions are reformulated from the standard sum of squared error criterion function in order to adapt the change of the

construction of the optimization problem. Other GA-based clustering applications have appeared based on a similar framework. They are different in the meaning of an individual in the population, encoding methods, fitness function definition, and evolutionary operators [Maulik and Bandyopadhyay 2000; Tseng and Yang 2001].

3.3 Context information acquisition

The goal of context information acquisition is to determine what a user is trying to accomplish. Because the user's objective is difficult to determine directly, context clues are used to help infer this information and inform an application on how best to support the user. Context awareness represents a generalised model of input (both implicit and explicit), allowing almost any application to be considered more or less context aware insofar as it reacts to input and the environment. However, there is divergent opinion as to whether context should only comprise automatically acquired information or also include manually acquired information. In an ideal setting context would be obtained automatically and there would be no need for manual acquisition. However, in the real world not all context information can be sensed automatically and applications must rely on the user to provide it manually. We define the term *contextual information* as the retrieved and relevant documents that encompass the *context* of the information seeker. Context-aware applications are often distributed because they acquire context information from a number of different sources [Dey 1998]. As much as the models for application distribution are well known, they are not always appropriate for distributed context information acquisition. Indeed, context awareness is most relevant when the environment is highly dynamic, such as when the user is mobile. Thus context-aware applications can be implemented on very diverse kinds of computing platforms, ranging from handheld devices to wearable computers to custom-built embedded systems [Bauer et al. 1998]. As a result context-aware applications require lightweight, portable and interoperable systems that can be implemented across a wide range of platforms. Dey [2001] proposes three basic functions that should be implemented by any context-aware application: presentation of information and services, automatic execution of services and storage (and retrieval) of context information. Presentation of information and services refers to functions that either present context information to the user, or use context to propose appropriate selections of actions to the user. Example is showing a user their location on a map and possibly indicating nearby sites of interest. The second function, automatic

execution of services, describes functions that trigger a command or reconfigure the system on behalf of the user according to context changes. Example includes a car navigation system that recomputed driving directions when the user misses a turn. In the third type of function, storage and retrieval of context information, applications tag captured data with relevant context information.

3.3.1 Context-awareness models

A context-awareness model defines and stores context information in a machine-readable form. Strang and Linnhoff-Popien [2004] summarised the most relevant context-modelling approaches based on data structures used for representing and exchanging contextual information in their respective systems. These are highlighted below.

1.) Key-value models: These represent the simplest data structure for context modelling. They are frequently used in various service frameworks, where key-value pairs are used to describe the capabilities of a service. Service discovery is then applied with matching algorithms which use these key-value pairs.

2.) Object-oriented models: Modelling context using object-oriented techniques offers the full power of object orientation (e.g. encapsulation, reusability and inheritance). Existing approaches use various objects to represent different context information (such as temperature, location, etc.), and encapsulate details of context processing and representation. Access to the context and context-processing logic is provided by well-defined interfaces like the hydrogen model [Hoffer *et al.* 2002].

3.) Logic-based models: These models have a high degree of formality, and typically facts, expressions and rules are used to define a context model. A logic-based system is used to manage the aforementioned terms and allows addition, updating or removal of new facts. The inference process is used to derive new facts based on existing rules in the systems. Contextual information is then represented in a formal way as facts.

4.) Ontology-based models: Ontology represents a description of concepts and their relationships. These models are very promising for modelling contextual information due to their high and formal expressiveness and possibilities for applying ontology reasoning techniques.

5.) User-context perception model: This is a model created to help the system designer understand the challenge(s) faced in creating context-aware systems. As an example, a car

navigation system works very well if one is in a new city; however, when using it around a familiar area one may sometimes be surprised at the route it tries to direct one to.

3.3.2 Characteristics of context information

In this sub-section, we summarize the obvious characteristics of context information used in mobile computing systems according to work reported in [Hendrickson *et al.* 2002]. These characteristics can determine the design requirements for our proposed context-aware IR model described in Chapter 5.

a.) *Context information displays a range of temporal characteristics:* Context information can be characterized as either static or dynamic. Static context information describes those aspects of a pervasive system that are invariant, such as person's date of birth. As pervasive systems are typically characterized by frequent change, the majority of information is dynamic. The persistence of dynamic context information can be highly variable; for example, relationships between colleagues typically endure for months or years, while a person's location and activity often change from one minute to the next. The persistence characteristics influence the means by which context information must be gathered.

b.) *Context information is imperfect:* Imperfection is another second feature of context information in pervasive systems. Information may be incorrect if it fails to reflect the true state of the world it models, inconsistent if it contains contradictory information, or incomplete if some aspects of the context are not known. These problems may have their roots in a number of causes. For example: context producers, such as sensors, agent's technology, derivation algorithms and users, may provide faulty information.

c.) *Context information has many alternative representations:* Much of the context information involved in pervasive systems is derived from agents. For this reason, there is usually a significant gap between sensor output and the level of information that is helpful to applications, and this gap must be bridged by various kinds of processing of context information.

d.) *Context information is highly interrelated:* For instance, in a given context aware medical knowledge information management scenario, relationships are evident between healthcare providers, their devices and their services communication channels (for example, ownership of devices and channels of devices and proximity between healthcare providers and their devices). Other less obvious types of relationship may be related by derivation of

rules which describe how information is obtained from one or more other pieces of information.

3.3.3 Context-centred information retrieval

Context-centred IR is an expression which can be used to encompass tools, techniques and algorithms aimed at producing an outcome (in response to a user's query), which is tailored to the specific context. When context is referred to the user context, we may talk about personalized IR. To personalize search results means to explicitly make use of the user preferences to tailor search results.

The possible use of context in IR requires a context dependent IR strategy, involving two main activities: modelling the context (representation problem) and using the context to enhance search quality (definition of processes which make use of context representation). The requirement activity is of a knowledge representation type, and is aimed at the definition of the context model. Such an activity comprises sub-activities such as the identification of the basic knowledge which characterizes the context, the choice of a formal language by which to represent this knowledge, and a strategy to update this knowledge (to adapt the representation to context variations). The second activity is aimed at defining processes (algorithms), based on the knowledge represented in the context representation and the user query, are formulated to produce as a search result of appropriate relevance.

3.4 Extracting representative context

Context clusters are lexical chains that represent the context or topic of a document. Morris and Hirst [1991] were able to use various kinds of syntactic categories when composing lexical chains, because they used Roget's Thesaurus as a knowledge base. However, in this thesis we use synonyms dictionary as our text corpus. However, we followed the approach of previous researchers and limited our research to noun [Budanitsky 1999; Fellbaum *et al.* 1998]. Hirst and St-Onge [1998] adapted the Roget's-based relations of Morris and Hirst to WordNet-based relations. They limited the chaining process only to nouns in their study. By the same reason, in this thesis, the approach is studied to cluster only nouns of clustering candidate.

Fellbaum *et al.* [1998] defined five relations in the order listed: identity, synonym, hypernym and meronym to compose context clusters by related lexical items. The notion of a context

cluster proposed here not only groups related lexical items, but also assigns each lexical item and context a weight that represents its semantic importance degree within a document. Thus, we define a context cluster as a weighted lexical chain that represents an aspect of the meaning of a document and expresses the semantic importance within the document using the following definition.

Definition 1 Let $N = \{N_1, N_2, \dots, N_l\}$ be the set of nouns in a document, and $R = \{\text{identity, synonym, hypernym, meronym}\}$ be the set of lexical relations. Let $C = \{C_1, C_2, \dots, C_m\}$ be the set of context clusters in a document. Context cluster C_j is composed of N_i and R_k . Each N_i and C_j has a weight that represents their respective extents of semantic importance in a document.

To construct a context cluster, we group related terms based on the notion of lexical chains, and then we assign the context cluster a weight based on the relations that a word has with other words. Firstly, to illustrate the notion of a context cluster, we apply our clustering approach using context awareness to the extracted text. In the extracted text, the nouns used in constructing context clusters must be distinguished from the other words. It is for this reason, that context clusters with more representative clusters of terms are considered over other clusters, and lexical relations are exploited to search for the semantically important terms within a document. The method for determining the representative contexts in a given document is described in detail in the following sub-section.

3.5 Estimation of the semantic importance of terms

To estimate the semantic importance of terms within a given document, the representative contexts should first be identified for a given document. To achieve this, we define two weight functions for each context cluster and the terms in that context cluster as definitions 2 and 3.

Definition 2 (Score of a term). Let $T = \{T_1, T_2, \dots, T_l\}$ be the set of terms in a context cluster. Let $R = \{\text{identity, synonym, hypernym, meronym}\}$ be the set of lexical relations. Let $N = (R_i, T_i)$ be the number of relations $R_k \in R$, that term $T_l \in T$ has with other terms, and let $W(R_k)$ be the weight of the relation R_k . Then the score $S_{\text{TERM}}(T_l)$ of term T_l in a context cluster is defined

$$\text{as: } S_{\text{TERM}}(T_l) = \sum_{k=1}^5 N(R_k, T_l) \times W(R_k) \quad (3-1)$$

$S_{TERM}(T_l)$ is determined by the relations that T_l has with the other terms and their weights. A large value of $S_{TERM}(T_l)$ indicates that T_l is a semantically important term in a document. The relation weight $W(R_k)$ is in the order listed: identity, synonym, hypernym and meronym (i.e., identity highest and meronym lowest) [Fellbaum *et al.* 1998]. Based on Definition 2, we now define the weighting function of a context cluster in definition 3.

Definition 3 (Score of context cluster). Let $C = \{C_1, C_2, \dots, C_m\}$ be the set of context clusters in a document. Let $T = \{T_1, T_2, \dots, T_k\}$ be the set of terms in a context cluster $C_i \in C$. Let $S_{TERM}(T_l)$ be the sum of $T_l \in T$. Then the score $S_{CONTEXT}(C_i)$ of a context cluster C_i in a document is defined as:

$$S_{CONTEXT}(C_i) = \sum_{l=1}^n S_{TERM}(T_l) \quad (3-2)$$

Thus, $S_{CONTEXT}(C_i)$ is obtained by summing the scores of all the terms in C_i . A large value of $S_{CONTEXT}(C_i)$ indicates that C_i is semantically important context within the document. From the weighted context clusters, we extract a set of representative context clusters for the document.

The weights of the five relations used in the clustering of terms were set from 0.1 to 1.5 (identity highest and meronymy). For the weights of the basic relations, no general guidelines exist except for the research in WordNet [Fellbaum *et al.* 1998]. In this thesis, we followed the principle of WordNet when assigning the relation weights.

For example, consider the context cluster containing four terms shown in Figure 3-1, in which the identity relation weight, $W(iden)$, is set to 0.8 and the synonym relation weight, $W(syn)$, is set to 0.6. Given that term T_1 has one identity relation, $N(iden, T_1) = 1$, and two synonym relations, $N(syn, T_1) = 2$, the score of T_1 is found to be 2.0 by the following calculation:

$$\begin{aligned} S_{TERM}(T_1) &= \sum_{K=2}^2 N(R_k, T_1) \times W(R_k) \\ &= N(iden, T_1) \times W(iden) + N(syn, T_1) \times W(syn) \\ &= 1 \times 0.8 + 2 \times 0.6 = 2.0 \end{aligned}$$

Similarly, we found that $S_{TERM}(T_2) = 0.8$, $S_{TERM}(T_3) = 0.6$, $S_{TERM}(T_4) = 0.6$. Thus the score of the context cluster C_1 is calculated as

$$S_{CONTEXT}(C_1) = \sum_{j=1}^4 S_{TERM}(T_j) = (2.0 + 0.8 + 0.6 + 0.6) = 4.0$$

From the weighted context clusters, we extract a set of representative clusters for the document. For example, in the system shown in Figure 3-1, the clusters that best represent the context of the text are C_3 and C_4 .

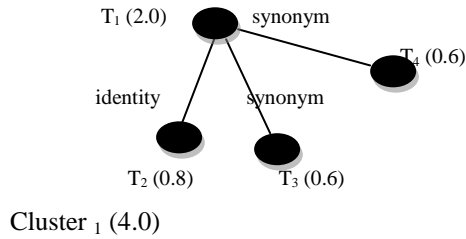


Figure 3-1: Weight of a context cluster

Definition 4 (representative context cluster). Let $C = \{C_1, C_2, \dots, C_m\}$ be the set of context clusters in a document, and Let $C^R = \{C_1^R, C_2^R, \dots, C_n^R\}$ ($n \leq m$) be a set of representative context clusters that satisfy the following criterion:

$$S_{CONTEXT}(C_i^R) \geq \alpha \cdot \frac{1}{m} \sum_{j=1}^m S_{CONTEXT}(C_j) \quad (i = 1, 2, \dots, n) \quad (3-3)$$

Where m is the number of contexts in a given context document and α is a weighting coefficient that is used to control the number of the representative context clusters to be considered. The criterion for representative context clusters in Definition 4 is designed to extract the main contexts in a document. It does this by using the average score of the context clusters in conjunction with the weighting coefficient α . After extracting the representative context clusters of a document, we extract the terms in each representative context cluster as index terms that capture the aboutness of the document, and regard the scores assigned to those terms as the index weights that represent the semantic importance within the document.

Definition 5 (Semantic index). Let $C^R = \{C_1^R, C_2^R, \dots, C_n^R\}$ be the set of representative context clusters for a document. Let $T = \{T_{i1}, T_{i2}, \dots, T_{ik}\}$ be the set of terms in context cluster $C_i^R \in C^R$. Then the index terms and their weights for the document are $(T_{il}, S_{TERM}(T_{il}))$ for $1 \leq i \leq n$.

3.6 Proposal for a clustering technique

Obviously, there is no clustering algorithm that can universally be used to solve all problems. Usually, algorithms are designed with certain assumptions and favour some type of biases. In this sense, it is not accurate to say ‘*best*’ in the context of clustering algorithm, although some comparisons are possible. The comparisons are mostly based on some applications under certain conditions, and the result may be quite different if the condition changes. The choice of the clustering technique will determine the outcome, and the choice of algorithms will determine the efficiency with which it is achieved. Focusing on text retrieval application, this research study addressed the problem of improving relevance of documents retrieval by a context-based approach. In document retrieval, little prior information is available about the text, and we must make few assumptions about the text as possible. On one hand, ranking strategy can provide a valuable base of information for clustering in a dynamically organized hierarchy. On the other hand, a cluster strategy can provide a valuable base for altering the rank of the retrieved results.

Clustering may help with grouping into a much smaller number of groups of related documents, ordering them by information relevance, and returning only the relevant documents from the most relevant group or several most relevant groups. It is for this reason that individual users need to guide the clustering process so that the clustering will be more relevant to the users’ specific contextual interest. Besides, the WWW delivers huge number of documents in response to a user query. However, due to lack of structure, the users are at a loss to manage the information contained in these documents efficiently. The importance of text mining is used to gather meaningful relevance information from text and includes tasks like Text Categorization, Document Clustering (also referred to as Text Clustering), Text Analysis and Document Summarization. Thus, text mining examines unstructured textual information in an attempt to discover structure and implicit meanings within the text.

One main problem in this area of study is regarding organization of text document. Text clustering is one of the most important text mining methods that are developed to help users effectively navigate, summarize, and organize text documents. This can be achieved by developing nomenclature or topics to identify different documents. However, assigning topics to documents in a large collection manually can prove to be a difficult task. Consequently, we propose a technique to automatically cluster these documents into related topics according to information relevance. Clustering is the proven technique for document grouping and

categorization based on the similarity between these documents [Song and Li 2006]. Documents within one cluster have high similarity with each another, but low similarity with documents in other clusters.

Conversely, in order to provide a valuable base of information for clustering in a dynamical organized hierarchy, we employ a ranking strategy to provide a limited number of ranked documents in response to a given query. It is for this reason, that we propose DROPT technique to provide a solution to the problem of ranking of retrieved documents.

3.7 Chapter Summary

In this Chapter we place focus on the clustering algorithms and review approaches appearing in literature. These algorithms evolve from different research communities, aim to solve diverse problems, and have their own benefits and issues. Thus we have already seen many examples of successful applications of cluster analysis. There still remain many open problems due to the existence of many intrinsic uncertain factors. In this thesis, we have presented an approach to document searching to extract and weight index terms. From among the concept clusters obtained from a document, representative context clusters were identified using the weights of each context cluster and terms in the context cluster. The terms in each representative context cluster were used as index terms, and the assigned term weights were used as the index weights. Conversely, we proposed an efficient clustering technique for document retrieval. The proposed clustering algorithm is suitable for applications in which the context is an important factor and the number of clusters is not known prior; an example of such application is user profiling and, more specifically, the mining of user context for the enhancement of an IR system performance. Clustering is an interesting, useful and challenging problem. It has great potential in diverse applications areas. However, it is possible to exploit this potential only after making several designs choices meticulously.

We discuss the issues of context representation, categorisation and acquisition of context information. Different researchers have approached these issues from individual perspectives in proposing frameworks to describe and handle context. In conclusion, considering the issues and aspects of context awareness highlighted here, more understanding of requirements in the development of context-aware applications is essential. In our study we examine context awareness in modelling user information needs.

Chapter 4

Context-Aware IR: The DROPT Technique

4.1 Introduction

This Chapter introduces the document ranking technique for context-aware IR known as a document ranking optimization (DROPT) according to information relevance. A document ranking technique is an algorithm that tries to match documents in the corpus to the user, and then ranks the retrieved documents by listing the most relevant documents to the user at the top of the ranking. Regrettably, despite the exposure of individual users to domain of Web retrieval and online documentation systems with document ranking features; it rarely addresses the information relevance of ranked output as core issue.

4.2 Parameters used for ranking principles

In this section we study the problem of ranking of retrieved documents. For example, we desire to rank a set of scientific articles such that those related to the query 'information retrieval' are retrieved first. The basic assumption we make is that such a ranking can be obtained by a weighting function $w(tf \times idf)$ which conveys to us how relevant document d is for query q . The document ranking will be done by taking a weighted average of all determined parameters. Table 4-1 depicts the summary of notations.

Table 4-1: Summary of ranking notations

Parameter Name	Description
d_j	indexed document
q_i	i -th query vector
(q, d)	document-query pair
$W = D \times Q$	convolution matrix
$w(tf \times idf)$	weighting function
tf	term frequency

idf	index term frequency
$Val_i = \max \{t_{ij}\}$	maximum relevance weight value added to matrix G
$D = \{d if \ val_i > 0\}$	documents sorted in ascending order of relevance value
$v \in \{0,1\}$	relevance numerical weight values normalization interval
$G = [g_{ij}]_{n \times l}$	query vector defined as a matrix G
$\bar{w} = \prod_{i=1}^n \frac{1}{l} \sqrt{\sum_{j=1}^l w_{ij}^2}$	weighted root mean square (RMS) to determine the overall relevance fitness of all documents with respect to a given query
n	number of queries for self-learning
N	size of the corpus
w_{ij}	Weights of terms in the document vectors

The necessity of being able to deal with each (document, query) pair independently arises from details of practicality on search engines back end prototype (which was wrapped around Google). To search through a large collection of documents efficiently it is preferable to assign a numerical weight to each document individually. In this respect, users are often only interested in the most relevant documents rather than the entire ranked list. For example, for web search, it is likely that users will only want to look at the first 10 retrieved search results. Similarly, when retrieving documents, a user may only be interested to consider viewing the best top n documents. Our focus in this present thesis is to provide limited number of ranked documents to the user in response to a given query. Alternatively, user's satisfaction with the system may depend on how many documents he needs to scrutinize through until user finds a relevant one. Therefore, in this thesis we are concerned primarily about the retrieval of the most relevant documents according to information relevance rather than all of them.

Our approach to ranking of retrieved documents is centered on self-learning the weighting function $tf \times idf$ with required adaptivity properties. This is in contrast to past strategies in IR which rely on viewing the documents as information overloads to obtain weighting function without considerations for underlying semantic analysis. The semantic similarities between terms in documents, which attracted the interest of many researchers who realized viewing query terms as relevance information is limiting. Therefore, in this thesis we take advantage of query terms occurrences and self-learning to guide us in finding a weighting

function that can automatically adjust its search structure to a user's query behaviour. In this regards, a good ranking criterion remains the choice of an IR system expert.

4.2.1 The statement formulation

Let us define this problem in the semantics analysis of the documents itself by self-learning. Assume that for a query q we have a set of documents $D := \{d_1, \dots, d_l\}$ with associated relevance numerical weight values $v = \{v_1, \dots, v_n\}$ where $v_i \in \{0,1\}$ as normalization interval which prompt a relevance order among the documents d_i . Here 1 is the maximum relevance numerical weight value corresponding to '*highly relevant*' and value 0 corresponds to '*irrelevant*'. Often the relevance weight values are generated by IR experts who retrieve information ranked according to relevance. For example, using the relevance context-aware information $v_i > v_j$ implies that document d_i is preferable to document d_j . This will express user's degree of interest by pairwise comparison of documents. It is our goal to rank retrieved document according to relevance numerical weight such that the documents with relevance value v_i will show up at the commencement of the ranked list than documents with relevance value v_j . This optimization of information retrieval is obtained by ranking the documents according to a relevance numerical weight value $\omega(tf \times idf)$ which is obtained from the weighting function w in descending order. Then we wish to return a relevance numerical weight subset v_i of v such that for each $d_i \in D$, we optimize the following weighting function:

$$\omega = (tf \times idf) \tag{4-1}$$

Where tf is the term frequency in the query-document pair, $idf = \log \left(\frac{N}{n_i} \right)$, n_i is the number of documents indexed containing term j ; N is the total number of documents in the corpus.

Based on work reported in [Baeza-Yate and Ribeiro-Neto 2011; Salton and Buckley 1988], equation (4-1) notations suggest diverse approaches to this weighting function problem involve statistics and VSM to enhance the retrieval effectiveness. We propose a solution to this problem of ranking of retrieved documents based on our DROPT technique.

4.3. DROPT technique

An adaptive DROPT technique must be able to personalize to individual user interests, adapt as context changes according to individual user's interest and capable to explore new

domains for potentially relevant context-aware information. The DROPT technique can be applied to concept-based knowledge domain used as the basis for representing user's interest where context is formalized as matching-rules adapting the knowledge domain of IR experts. A DROPT technique for document retrieved from a corpus is defined with respect to document index keywords and the query vectors. The calculated value represents the search context that take a relevance weight's into consideration in the retrieval of information process. It is therefore important for a system to assign numerical weights to the returned documents and provide a ranked list to the user. In other words, documents that are more relevant are ranked ahead of documents that are less relevant. This requires us to consider relevance value to the normalization interval $v \in \{0,1\}$.

One of the lessons learnt from previous studies using $tf \times idf$ concepts, in particular with IR effectiveness, is that it is very difficult to obtain relevance, because documents are viewed as bag-of-words without any consideration to the underlying semantically and syntactical structure or term proximity in the text. To this end, the user perception and user knowledge level are factors that influence the relevance of a retrieved document. In this regards, current ranking algorithms have low precision in average and are not adaptive to user needs and thus resulted in poor performance. So, a DROPT algorithm seems better ranking algorithms that can take the role of the user into consideration in web-based retrieval system. We can achieve this by user context on retrieved documents who indicate documents that are relevant and otherwise from the designated document database.

User can play the most essential role in the system and the basic goal should be to satisfy the user by a good ranking according to relevance information. The DROPT technique has taken a new approach that allows the combination of the context-aware clustering and context-aware. The context-aware clustering will be suitable for applications such as user profiling and mining user preferences for the enhancement of IR system performance in which the context is an important factor and the number of clusters is not known prior. Also context-aware is suitable for applications that can use contexts to provide relevant information to user, where information relevance depends on the retrieval of document ranked. The goal of context information awareness acquisition should be to determine what a user is trying to achieve while performing his ad hoc retrieval tasks. Due to the fact that user's intention is difficult to determine directly, context indications can be used to help infer this information and inform an application on how best to support the user's task effectively.

Consequently, we propose an approach that looks at result rankings instead of adaptations on the input. This ranking-driven approach also adheres to the cognitive aspect of IR, as the top of a ranking, presenting the best results for a specific query, is in the user's focus [Agichtein *et al.* 2006]. For DROPT technique, this means that changes at the top of a ranking need to be emphasized by a higher numerical weight as more relevant documents are ranked ahead of documents that are less relevant according to information relevance. This aspect is handled by the mathematical formalization of weighting technique introduced in *Sub-section 4.3.1*

The DROPT technique can adapt the ranking of the search result set of documents. Most existing search engines compute a ranking value of information relevance between the document and the information needs (e.g. the user's query). Hence, the measure takes normalization to the interval $v \in \{0,1\}$. A personalized search engine back end (which was wrapped around Google) can then compute a relevance numerical weight for every document in the ranked result set by DROPT technique. The benefit of this approach is that this relevance numerical weight value has only to be computed for the returned top search result set of documents. The main drawback is that this value has to be computed at query time. This DROPT algorithm is also suitable for Meta search engines by [Shivaswamy and Joachim 2011], as the user-dependent DROPT algorithm can focus on a limited number of the top returned documents.

The DROPT technique, has taken an approach of the context search-based. The relevance numerical weight of a document is calculated as a function of the occurrence of the keyword across a document. For example a search string like "*Information Retrieval*" may be considered. Let in document, d_1 the string "*Information Retrieval*" appears. In document, d_2 only the word "*Retrieval*" appears. Now it may happen that d_2 refers to "*Retrieval Performance*" which is not at all related to the search string context "*Information Retrieval*". So it can be inferred that, in a document where the entire search string appears as a whole is more relevant to the search topic than a document where only part of the string appears. In the proposed DROPT technique the words in "*Stop List*" are removed first from the search string. After proper stemming, the relevant index terms are extracted from the search string. Next, the occurrence of each keyword is found out, and a numerical weight is calculated accordingly. So for the above example the term "*Information Retrieval*" will get a relevance weight due to matching results whereas the term "*Retrieval*" will not get a relevance weight due to no matching result.

The presentation of the search results to the user is an important aspect in human-computer information retrieval (HCIR). The presentation method lists result rankings in ascending order according to relevance numerical weights in response to a given query request. The essence of HCIR to the DROPT technique is to give the user an impression on how good the search results are. In particular, they allow the individual users to judge the retrieval of ranked documents according to information relevance how much better search result 1 is than search result 2. An assumption that suggests itself is that presentation of information about the relevance of documents can influence user's judgments in result rankings. The intention behind DROPT technique is to use relevance judgment for a query to explicitly take user interests into consideration about retrieved documents to meet their information needs under context changes. Depend on the user information needs and context, the approach adapts itself with the environment to present an appropriate ranking for the user's satisfaction. The proposed technique query terms may give different rankings to a document depending on the semantic characteristics terms of the document itself which is not possible in any existing traditional $tf \times idf$ algorithms. DROPT is a user's behaviour source that can be used for ranking of retrieved document to influence the IR process. The mathematical definition of the DROPT technique is introduced in the following subsection.

4.3.1 Formalization of mathematical model definitions

This is based on equation (4-1), a DROPT measure for documents retrieved from a corpus is developed with respect to document index keywords and the query vectors. Naturally, given the notation we present for the problem, the use of statistical methods has proven both popular and efficient in responding to the problem [Salton and Buckley 1988; Baeza-Yate and Ribeiro-Neto 2011]. This mathematical model definition is based on calculating the weight (w_{ij}) of keywords in the document index vector, calculated as a function of the frequency of a keyword k_j across a document d_i .

The DROPT technique is based on IR result rankings, where a ranking R consists of an ordered set of ranks. Each rank consists of a relevance numerical weight value $v \in \{0,1\}$ where v represents the relevance numerical weights of the retrieved documents. Each rank is assigned an ascending rank number n , such that:

$$R = [\{1, v_1\}, \{2, v_2\}, \dots, \{n, v_n\}], \text{ where } v_1 > v_2 > \dots > v_n. \quad (4-2)$$

Our technique, DROPT is composed of six steps.

Step 1: Initialization of Parameters

(a) Let a query vector, Q , be defined as:

$$Q = [q_1, q_2, q_3, \dots, q_l] \quad (4-3)$$

Where, $q_i = (x_i, 1)$, x_i being a term string with a weight of 1.

(b) Let the indexed document corpus be represented by the matrix:

$$D = \begin{bmatrix} d_{11} & d_{12} & d_{13} & \cdots & d_{1j} \\ d_{21} & d_{22} & d_{23} & \cdots & d_{2j} \\ & \vdots & & & \\ & \vdots & & & \\ d_{N1} & d_{N2} & d_{N3} & \cdots & d_{Nj} \end{bmatrix} \quad (4-4)$$

Where $d_{jk} = (y_{jk}, w_{jk})$, y_{jk} being an index string, with weight w_{jk} .

(c) We compute the convolution matrix, representing:

A convolution matrix (*convmtx*) is a matrix formed from a vector, whose product with another vector is the convolution of the two vectors. $W = \text{convmtx}(D, Q)$ returns the convolution matrix, W such that the product of D and a vector, x is the convolution of D and x . If D is a column vector of length l , W is $(l+Q-1)$ -by- Q and the product of W and a column vector, x of length Q is the convolution of W and x . If D is a row vector of length n , W is n -by- $(n+Q-1)$ and the product of a row vector, x , of length n with W is the convolution of D and x . The convolution matrix (*convmtx*) is calculated from equation (4-5) given below:

$$W = D \star Q = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{1l} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{2l} \\ & \vdots & & & \\ & \vdots & & & \\ w_{n1} & w_{n2} & w_{n3} & \cdots & w_{nl} \end{bmatrix} \quad (4-5)$$

$w_{ij} = w_{jn} \times \text{IsEqualStringIgnoreCase}(q_{ij}, d_{kj})$, where q_{ij} are query vectors, d_{kj} are document vectors, w_{ij} are weights of terms in the document vectors, and w_{jn} are weights of terms in the query vectors, while n is the number of retrieved documents that are indexed by at least one keyword in the query vector. The matrix W gives a numeric measure with no context information.

Step 2: Search String Processing (Matching Mechanism)

The comparison of the issued query term against the document representation is called the query process. The matching process results are a list of potentially relevant context-aware information. Individual users will scrutinize this document list in search of the information they needs. The goal of context-aware information acquisition should be to determine what a user is trying to achieve while performing his\her matching tasks. The context-aware search agent is used to retrieve documents in response to an issued query and return the best matching documents according to specific matching rules (Table 5-2).

Step 3: Calculate Relevance Weight

Retrieved documents that are more relevant are ranked ahead of other documents that are less relevant. It is important to find relevance numerical weights of the retrieved documents and provide a ranked list to the user according to their information requests as follows:

- (a) Based on equation (4-1), the relevance weight is obtained according to document content.
- (b) Subsequently we calculate the average mean weight (\bar{w}) using the weighted root mean squares (RMS) to determine the overall fitness value of retrieved documents with respect to a given query calculated as:

$$\bar{w} = \prod_{i=1}^n \frac{1}{l} \sqrt{\sum_{j=1}^l w_{ij}^2} \quad (4-6)$$

Where w is the average relevance mean weight of each retrieved document, n is the number of keywords terms occurrences in each retrieved document, l is the total size of the keywords in the corpus, and w_{ij} are the sum weights of terms of the document vectors.

Step 4: User Feedback about Retrieved Documents

User's feedback about retrieved documents is based on overall relevance weights \bar{w} to construct a personalized user profiling interests. The relevance weight of a document will be measured according to the degree of fitness of the document with respect to the query vector with small-operator defined as matrix G in equation (4-7) below:

$$(a) \quad G = [g_{ij}]_{n \times l} \quad (4-7)$$

where $g_{ij} = \min(w_{ij}, \bar{q}_{l,j})$

$$1 \leq i \leq n, 1 \leq j \leq l$$

Then we retrieve the documents by a specific average mean weight (\bar{w}) given by the system calculated from equation (4-6). Where G is a query vector with a small-operator defined as a matrix, w_{ij} are weights of terms of the document vectors, and q_{ij} are queries vectors.

(b) Any numerical weight component of matrix G greater than the average mean weight, \bar{w} will be retained which adds to a matrix T shown in equation (4-8).

$$T = [t_{ij}]_{n \times l} \quad (4-8)$$

Where
$$\begin{cases} t_{ij} = g_{ij}, & \text{if } g_{ij} \geq \bar{w} \\ t_{ij} = 0, & \text{if } g_{ij} < \bar{w} \end{cases} \quad 1 \leq i \leq n, 1 \leq j \leq l$$

(c) Based on matrix T , we calculate relevance numerical weight values, for all set of documents D , which are defined as the largest weighting values for each corresponding vector as equation (4-9).

$$Val_i = \max \{t_{ij}\}, 1 \leq i \leq n \quad (4-9)$$

$$1 \leq j \leq l$$

(d) Document d_i is retrieved if value Val_i is greater than zero and added into the retrieved documents set, D shown in equation (4-10). Hence, d is a subset of D (documents in the corpus). The average relevance mean value within the normalization interval $v \in \{0,1\}$ is computed for each document. After a query is made by a user, the system ranks the retrieved documents in such order:

$$D = \{d \mid \text{if } val_i > 0, 1 \leq i \leq n\} \quad (4-10)$$

Step 5: Relevance Judgment

Individual user is asked to judge contextual factor (e.g. information relevance) influence on ranking task given a certain contextual dimension (*numerical weight is relevant or irrelevant*)

(a) If the ranked document is relevant to user information needs, the user finishes his/her query search context, then GO to **Step 4** according to user's document preference.

(b) Otherwise, user continues to search the document databases by reformulating the query or stop querying the designated database until relevant documents are ranked. GO to **Step 6**.

Step 6: Update Term Weight and Keywords Set

The keyword term set n provided by the ranked documents and the relevance numerical weight values will be updated by the user's feedback.

- (a) Any new query term not belonging to n will be added and a new column of relevance weight value will be computed and expanded for ranked documents routinely.
- (b) If any ranked document d_i is retrieved by the users, the corresponding relevance weight values with respect to the query keywords will be increased by equation (4-11). The default of β is set to increase the corresponding relevance numerical weight values.

$$w_{ij} = (w_{ij})^\beta, \quad (4-11)$$

Where $0 < \beta < 1, i \in \{i | d_i \in D\}$ and $j \in \{j | q_j = 1\}$

We coined the acronym DROPT to name our new adaptive algorithm that provides a limited number of ranked documents in response to a given query. Also it can improve the ranking mechanism for the search results in an attempt to adapt the retrieval environment of the users and amount of relevant context-aware information according to each user's request. Finally, the DROPT measure must be self-learning that can automatically adjust its search structure to a user's query behaviour.

UNIVERSITY of the
WESTERN CAPE

4.4 Evaluation approaches for context-aware IR

We summarize the evaluation approaches for context-aware IR used in adaptive IR systems according to work reported in [Bouramoul *et al.* 2011].

4.4.1 Evaluation of the relevance by the user's judgment

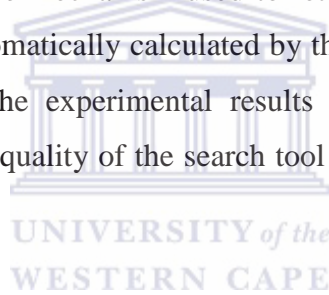
This is a user context based evaluation measure. It raises the question how user appreciates search results when an IR system returns a document to the user. There is recovery of information and the information is crucial for a given user context [Bouramoul *et al.* 2011]. The information relevance for a given user in a given context and the user determine the actual adequacy of results returned by the search tool with its information needs. Based on the principles of search adequacy and to allow consideration of the user's judgments during evaluation [Bouramoul *et al.* 2011], and adapted the approach from the work reported in [Bouramoul *et al.* 2010] to modelled the user by a static and dynamic context.

4.4.2 Evaluation of the relevance compared to the query

This is a contextual evaluation weighting approach by increasing the number of terms, of the query words compared to the words of the returned documents [Bouramoul *et al.* 2011]. The authors chose the weighted terms in the first time, then apply the proposed formula by an incremental way versus the number of words forming the query. This can be achieved by increasing query terms instead of a classic weighting of each word separately allowing better consideration for the query context during evaluation.

4.4.3 Evaluation of performance of the search tool

[Bouramoul *et al.* 2011] proposed an evaluation approach based on a number of criteria summarizing the problems generally encountered by users during a search session. The authors explained the criteria to include the nature of the manipulated information, the source of the information, and finally the mechanism used to retrieve this information. The values assigned to these criteria are automatically calculated by the system shortly obtaining results provided by the search tool. The experimental results estimated these values and give subsequently an overview of the quality of the search tool independently of the relevance of the results that it returns.



4.5 Chapter Summary

This Chapter has introduced a document ranking technique to IR with the intention to retrieve context-aware information ranked according to information relevance. The technique demonstrated in providing limited number of ranked documents in response to a given users' query. The DROPT approach combined context-aware clustering and context-aware suitable for user profiling and mining of user context for the enhancement of an IR system performance, which satisfy the focus of this thesis. Current ranking algorithms suffered from low precision and recall. DROPT technique adapts itself with individual user information needs based on environment and search context. The technique is designed purposely to overcome some of the limitations of existing traditional ranking $tf \times idf$ algorithms that ignore the semantic analysis of the document itself. In Chapter 6, the evaluation of the developed technique show performance improvements using $P@n$ over the chosen baseline algorithms runs. The next Chapter discusses the design and implementation of context-aware IR model.

Chapter 5

System Design and Implementation

5.1 Requirements analysis

The success of a software development inherently depends on whether the developed system works the way users expect it to work. While the full software development lifecycle covers the phases of analysis, design, implementation, integration, deployment, operation and maintenance [Maciaszek 2007], we will restrict the considerations in this thesis to the analysis phase, design and implementation. In the following, we will go through the phases of requirements determination as well as the design and architecture.

5.2. Requirements determination

The goal of the requirements determination phase is to analyse and document the hierarchy of processes underlying the application to be developed. Figure 5-1 shows an overview of the context-aware IR process from notion of context-awareness. From a user's perspective, the retrieval process splits down to four sub processes. Management of the knowledge domain is a general prerequisite which makes sure that a conceptualization and context-aware information annotated for retrieval with this conceptualization is at hand. This process may or may not be part of the functionality accessible to the user; an implementation for end users should try to hide most of this complexity from the user. Management of context dependencies based on input (retrieved documents) from search engines and the user herself enables context-aware IR based on the documents retrieved using IR method (DROPT technique). This process requires user input, at least at the level of building a user search profile. Query formalisation and result interpretation are not further analysed here, since they fall into the category of user interface issues that are too application specific. In the following, we detail the system services that are required to a software representation of the processes underlying context-aware IR optimization. Moreover, we discuss constraints imposed on the implementation.

The core functionality of the system is to allow the user to present queries to a knowledge domain that take the current context into account when they are processed. In particular, the system must be able to (i) adapt the search results to the user preferences and (ii) adapt the search results to external context-aware information retrieval provided by search engines. Moreover, the system allows the user to change her search profile file and thus the way the system reacts to context changes.

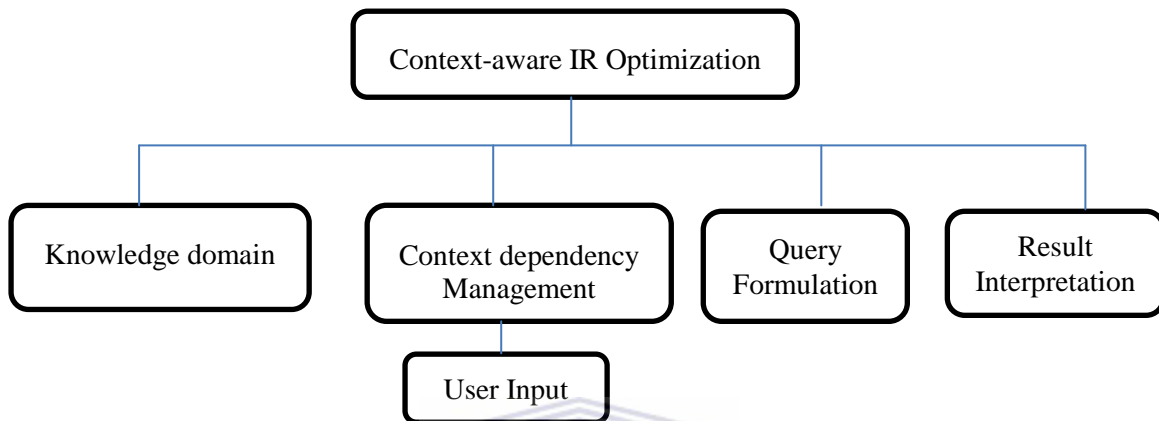


Figure 5-1: Process hierarchy for the contextual information retrieval

The system component of the application that enables the context-awareness must therefore act as a see-through representation between the IR method (DROPT technique) and the knowledge domain. Figure 5-2 shows the relationship between the context-aware IR model, the knowledge domain, and the IR method.

The constraints for the system implementation are imposed by the agent technology standards it should adhere to. The Resource Description Framework (RDF) [Kirn *et al.* 2006] and the Web Perl programming language are the most important standards to mention here. The system component handling context-awareness must therefore be compatible with knowledge domains expressed in Web Perl programming language. The retrieval of relevant context-aware information from search engine is an important area where exciting standards need to be considered.

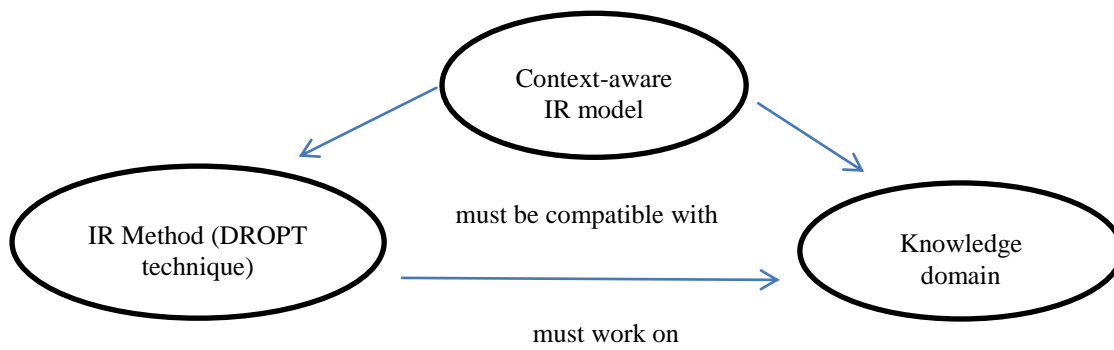


Figure 5-2: The IR method (DROPT technique) must always be applicable to the Web Perl programming language chosen for knowledge domain. Likewise, the context-aware IR model must be able to interoperate with the knowledge domain and make its relevant context-aware information available for the retrieval method (DROPT technique).

The different use cases have been transformed into a sequence diagram in Figure 5-3 that shows the order in which different steps are completed, and which components are involved at which step of the task.

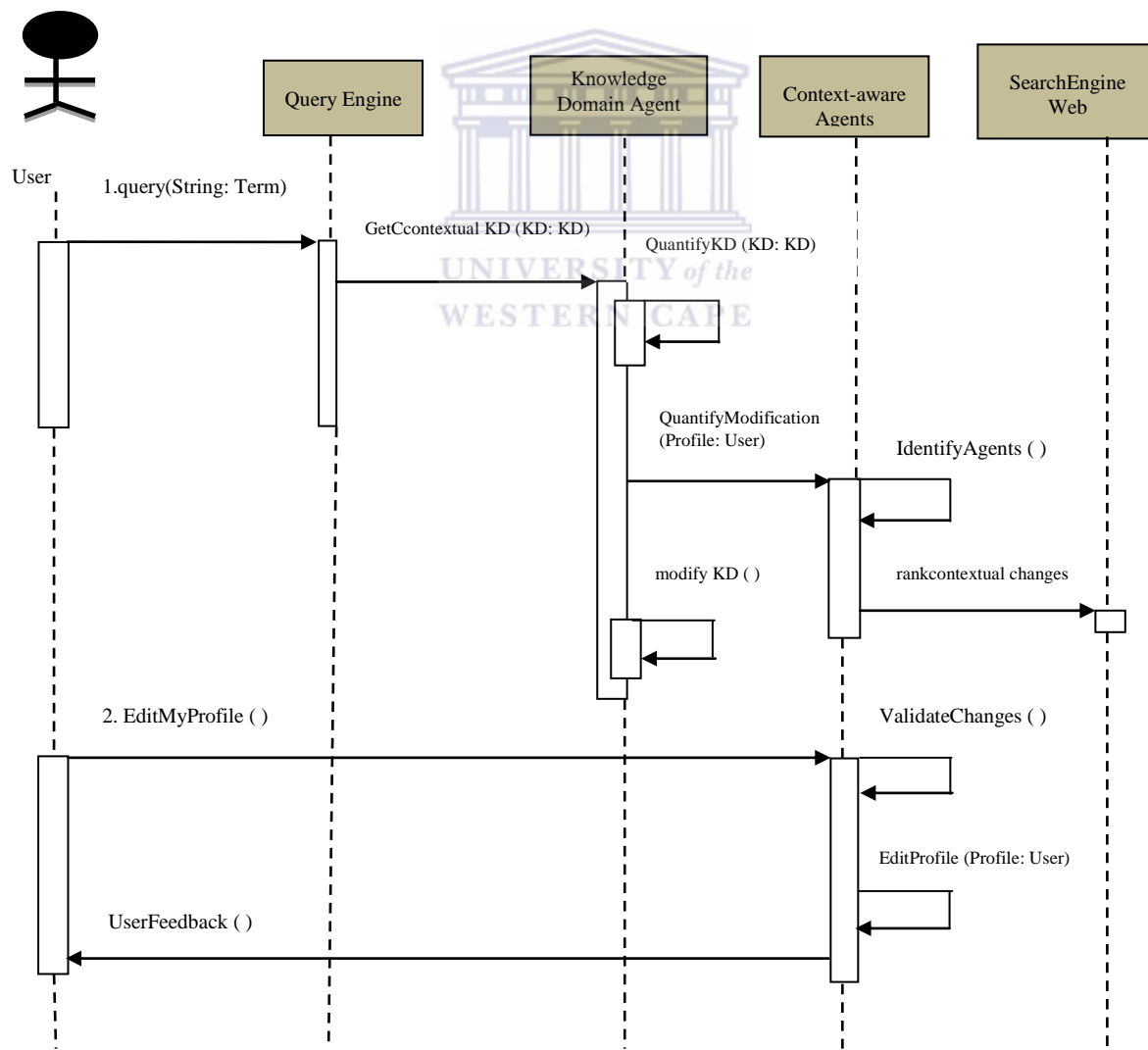


Figure 5-3: Sequence diagram for the two interaction options the user has with the system: (1) present context-aware queries and (2) edit her profile

A sequence diagram for context aware IR system plan, shown in Figure 5-3 above, is a diagram that shows actual events and interactions between events in the horizontal direction and sequence in the vertical direction. The vertical dotted lines represent the lifetime of the events and the horizontal arrows the interactions of messages between events based on environment and current context. Narrow elongated boxes on the event lifelines represent the activation of the event when interactions are sequential and represent calls to operations. The operation remains active until all the sequential operations, which it calls, have completed and returned, thus, allowing it to return control to its caller.

5.3 Design and architecture

The requirements determination and specification phases provided an abstract view on the system to be developed. They concentrated on identification of information (retrieved documents) processes the system needs to represent, and one of the components that are required to realize this system. The development of design and architecture, in contrast, provides the foundation for the actual implementation of the system. The deployment diagram in Figure 5-4 shows an overview of the components of the system and their interaction.

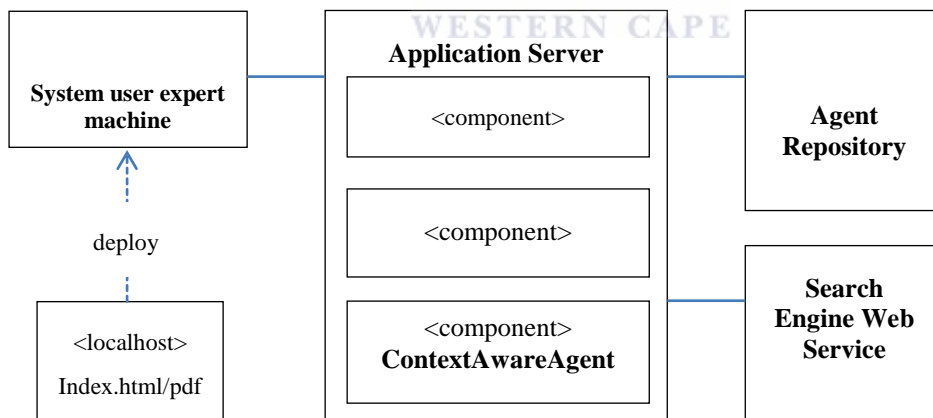


Figure 5-4: Deployment diagram for the system, making use of existing agent repositories framework and Search engine Web services

We assume that the system user expert runs in a search engines is therefore independent of any specific prerequisites on the user side concerning a specific operating system or software that needs to be installed. The main components to be developed are found on the application server side. The goal is to reuse existing software wherever possible that supports the

development of Perl Programming Language (PPL) with application user interface. We demonstrate the system implementation in Section 5.8 for context-aware information retrieval optimization. In the following sub-section, we present the architecture of the proposed context-aware agent-based systems guided by our design goals, especially the need for a robust and extensible system that supports information retrieval effectiveness.

5.3.1 The Proposed System Architecture

In this present study, in order to support the design and to ease the implementation of context-aware system, we proposed architecture with characteristics related to the IR application and techniques, where context-aware users are agents that act pro-actively on behalf of users in a given environment and context. The architecture of the proposed system as illustrated in figure 5-5 groups Use Cases into seven autonomous task categories, which are allocated to context-aware agents. There are seven types of agents recognized in the system to represent the IR solution: context-aware user interface agent, context-aware reformulate agent, context-aware search agent, context-aware document agent, context-aware match agent, context-aware user model agent, and context-aware display agent. Context-aware non-agent components include search engines Web services and data resources. Each of these agents is discussed in more detail as follows:

Context-aware user interface agent is designed for interaction with humans and responsible for mediating between the external user and the rest of the system (other agents). The context-aware interface provide means for creating a user profile that is tailored specifically to each context, and central to building context aware systems that conforms to the users' expectations, as well as allows the user to enter keyword based query terms. This agent allows the user to evaluate the relevance of the ranked documents, by giving a score to each document as a function of the frequency of keyword across a document. The interface notifies users the availability of search results and in turn provides feedback. ***Context-aware reformulate agent*** processes the input raw query from the *context-aware user agent* by pre-processing techniques (i.e. stemming operations). The refined queries are then sent to the context-aware match agent. ***Context-aware search agent*** carries out the task of submitting queries in the correct form and gathering context information from the Web. This agent uses the keywords to retrieve documents, hence the results of this task are then sent to the context-aware document agent. ***Context-aware document agent*** goal is to index the documents using normalized keywords. The representation typically used is a set of common features derived

from the document collection, which is a vector weighted keywords. Consequently, the highest indexed documents are sent to the context-aware match agent. **Context-aware match agent** performs the task (matching process) of comparing the refined queries against the indexed documents. The matching result is a list of potentially relevant documents that are then sent to the context-aware display agent, according to the user information needs. **Context-aware display agent** displays the results of the matching process (relevant documents) and performs ranking processes. The ranking function provides more accurate matching according to the representation of the current user model of user information need. Consequently, the ranked documents, according to their relevance numerical weights are then shown to the user through the context-aware user interface agent.

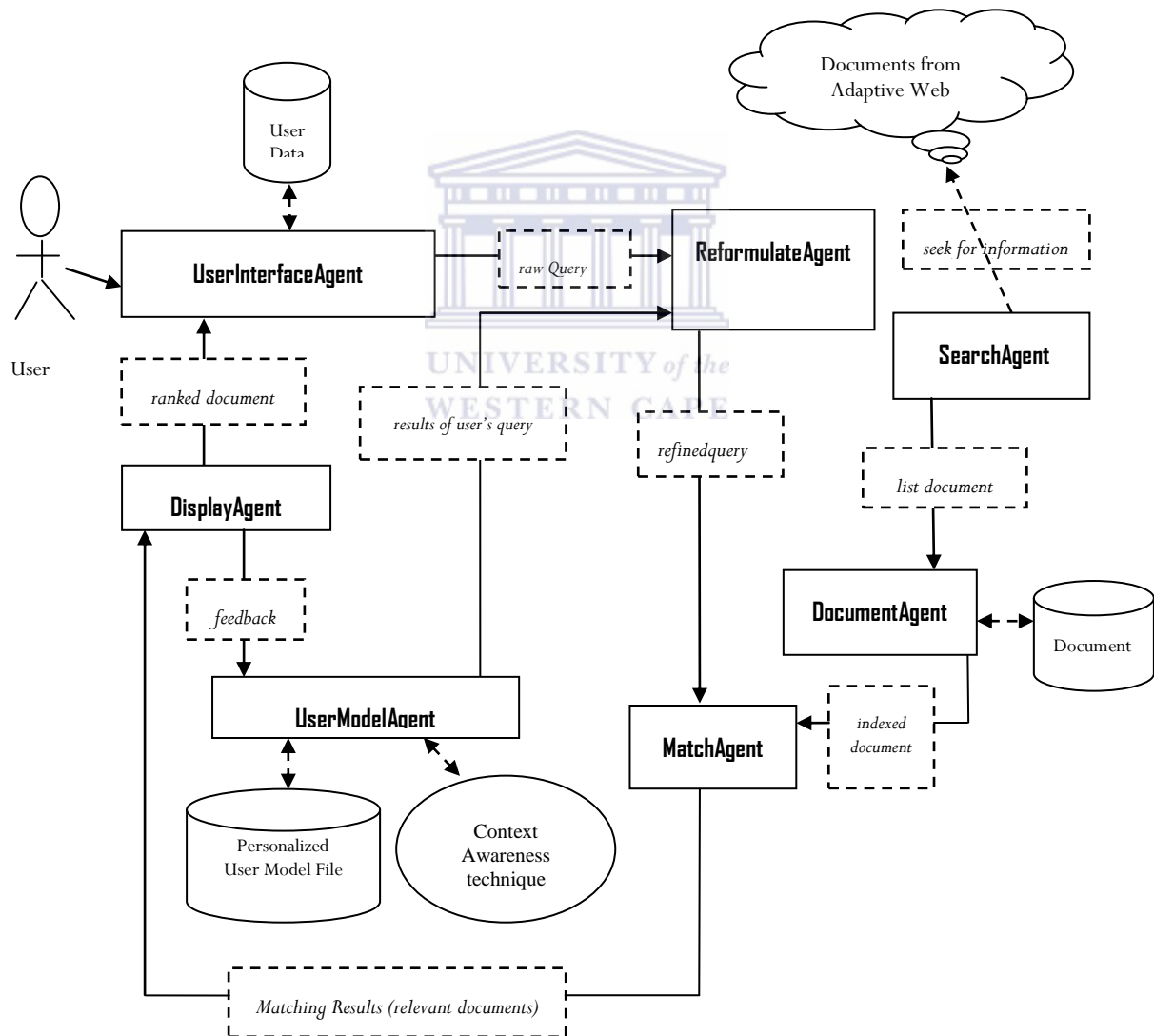


Figure 5-5: Overall Context-Aware IR Agent System Architecture

Context-aware user model agent is a user-feedback that modifies the representation of user-needs and employs context awareness as an enabling technology of adapting the information access process to user's information needs. The task of this agent is to guide the user in the query formulation process and to store and manage the user's interest in the form of a user profile that conforms to users' expectations. This agent is a major element of the system architecture and is composed of DROPT technique, relevance feedback and a context awareness component. The overall dynamic among the agents interaction diagram is shown in Figure 5-5.

5.4 A Context personalized information retrieval model

Context-aware IR optimization requires an adaptation of the processed information (retrieved documents) with respect to the individual users. It depends on the user's personal context whether a user blog article is worth reading with respect to the user's expectations and abilities. Web Perl Programming on the search engines; however lack the processing and IR method that are required to express such dependencies on the user profile. It's not possible for a user to rank all the retrieved documents from search engines as relevant, or that another user finds all retrieved documents below the average fitness score as irrelevant. We are thus looking for a workflow for the context management to enable how users can judge context changes for personalized retrieval based on the user profile. One fundamental problem of most current IR system is that they provide uniform access and retrieval of IR results to all users specially based on the query terms users entered to the system.

To address these issues we propose a personalized IR model based on document preferences as search context to rank individual users results (documents preferences) effectively and efficiently and the behaviours that individual user has engaged in during the matching tasks. The idea of context personalization is to predict relevant ranked documents according to relevance weights. This demonstrates a search context from search engine by observing and analysing user behaviours (i.e. keyword matching based querying frequency). The workflow of the design and evaluation of this proposed context-aware personalized IR model is shown in Figure 5-6. We generate two user predictive models about document ranking: 1) a predictive user model of the relevance of document content; 2) a predictive user model of ranking for currently retrieved documents. We believe this model (Table 5-1) can enhance individual user's retrieval performance greatly. The predictive user models generated data

analysis by individual users knowledge domain, while interacting with the search engine in which ranking of retrieved document has been controlled independently. By analysing the statistical associations between measures of user behaviours and their judgments of document relevance, we create a predictive user model of document relevance by assigning a numerical weight to each retrieved document and ranking of retrieved document, we can get a predictive user model of current search context (relevant or irrelevant). Ranking of retrieved documents could influence user's context because a user indicates documents that are relevant and otherwise according to relevance weights. The problem at hand is thus to find IR mechanism that allows for personalized context-aware IR. The DROPT technique is employed to enable context-aware IR as illustrated in Figure 5-6.

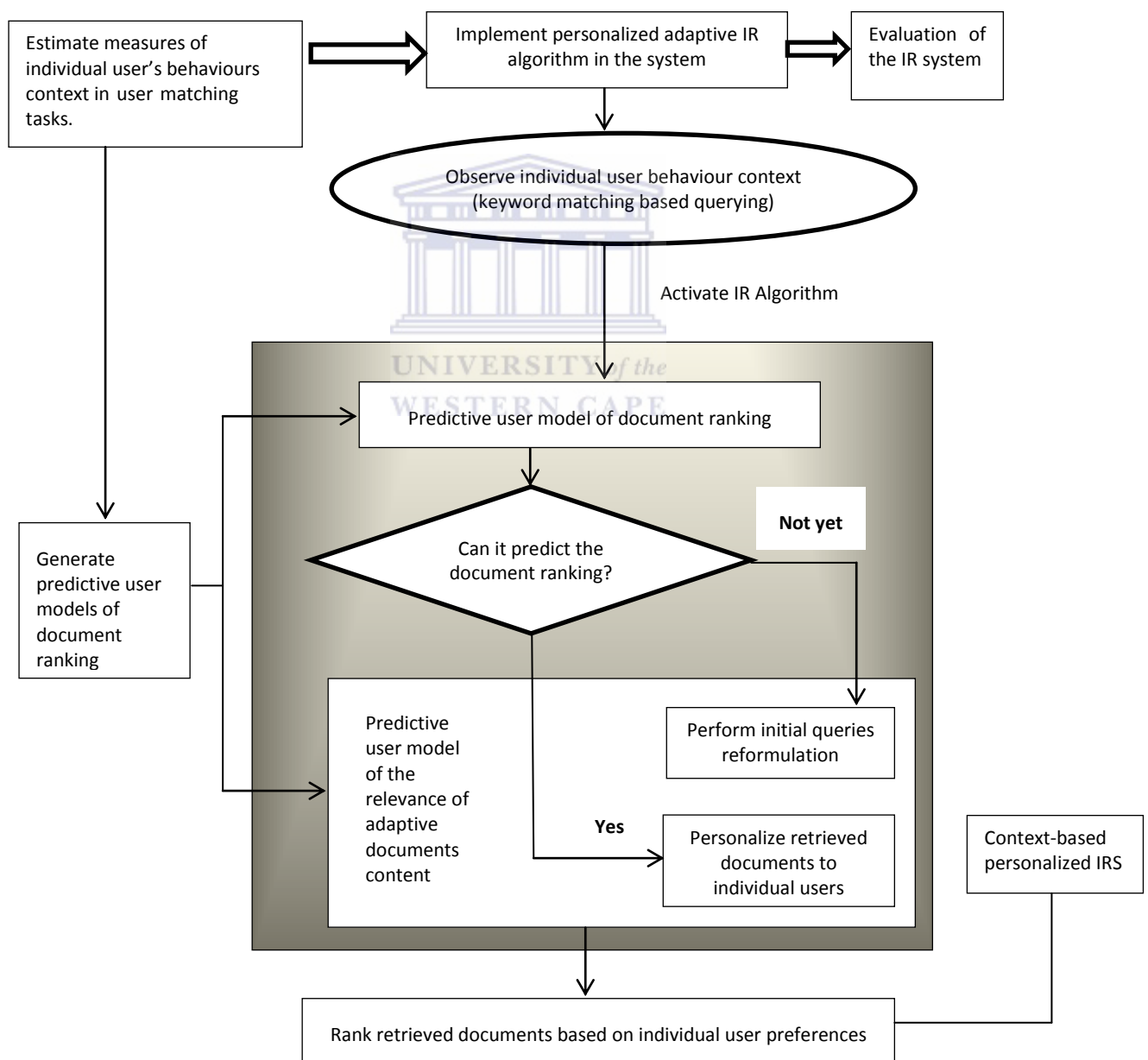


Figure 5-6: The generation of Context Personalized Information Retrieval

The purpose of predicting document ranking for IR system in this thesis is to personalize retrieved documents to individual users during their search context, rather than after they finish the entire document ranking tasks. So, the measures of user behaviour context, which can be immediately noticed is based on calculating the weight of keywords in the document index vectors, calculated as a function of the frequency of a keyword across a document (*keyword matching based querying results*) should be the main sources to predict ranking of retrieved documents according to relevance weights. The work reported in [Li and Belkin 2008] identified task type in human information behaviour as contextual factors to influence the way users search for information. We apply context-awareness in this thesis as a technique to reformulate original user's queries in order to improve the predicted relevance of retrieved documents. Also by reformulating a query we could not only increase the number of relevant documents but also rank the candidate documents.

Table 5-1: Predictive Document Ranking Model (PDRM) Table for User Model Preference

Can model predict document's relevance?	Document Content Context	Description of document ranking model
Yes	Relevant	Predicted to personalize current retrieved documents for ranking tasks.
Not Yet	Irrelevant	Predicted to perform initial queries reformulation but ignored if found to be irrelevant later.

Before the current retrieved document is predicted from individual users' behaviours context, the predictive user model of document relevance is calculated as measures of individual user search (i.e. frequency of keyword matching based querying) in their domain of knowledge; once the retrieved document is predicted from the model, and then the system can activate predictive model of document relevance for ranking task. This demonstrates how the predicted relevance documents can be used to assist users reformulate their initial queries to better understand users' current information needs by user preferences. To personalize search results means to explicitly make use of the user preferences to tailor search results.

5.5 Proposed system design

The major objective of personalisation is to predict and adapt the potentially relevant ranked documents to meet individual user's information requests, and so, collecting a richer collection of context information (document preferences) about individual users in their domain of knowledge may lead to improved personalization. We can achieve this pairwise comparison of documents by preference relevance feedback; a user indicates documents that are relevant and otherwise from the designated document context. Acquiring search context will assist IR systems provide personalized search results to individual users. Context includes the following aspects of the user's current situation such as location, knowledge, user preferred search context, work task etc. Ranking the retrieved documents user model can make the documents appears in the order as the user interest is matched. We have selected the web PPL for the implementation discussed in Section 5.8.

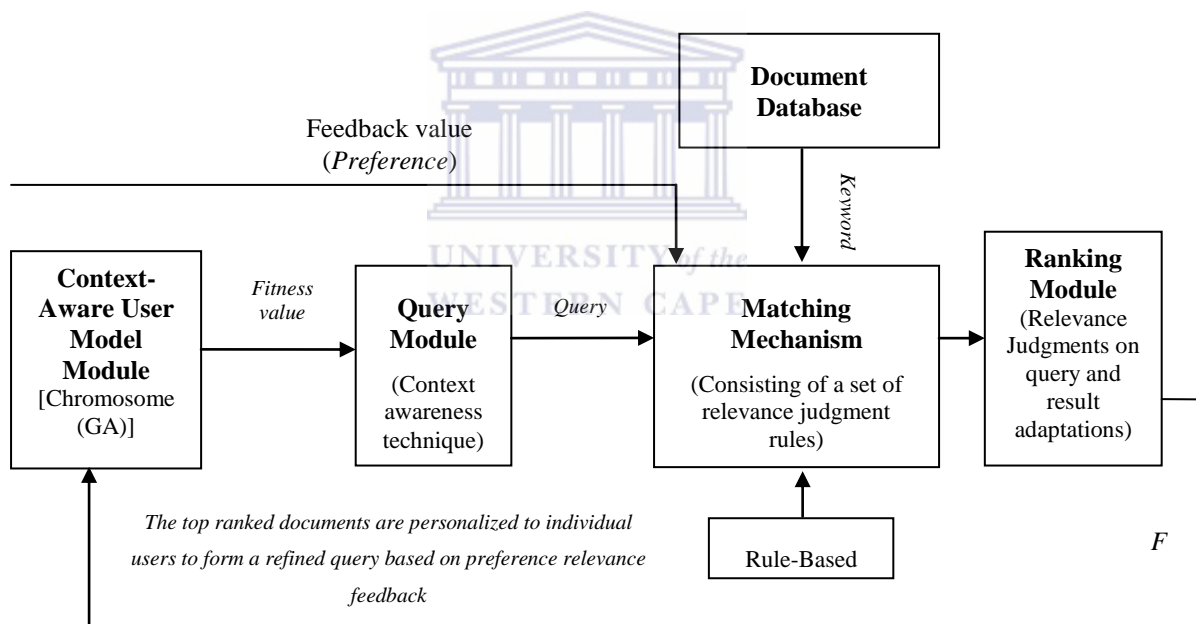


Figure 5-7: Context Aware Personalized System

The goal of personalization is to improve search performance by each individual user, to help them accomplish their search tasks. A personalized IR system should first correctly predict users' information need, document preferences and search context, and then take such information to provide personalized search results to individual users. Therefore, we evaluated the retrieval performances of our predictive models in this study. Aiming to clarify and achieve personalization in this thesis, we use GA with best keywords that best matches the user's interest and to improve the potential effectiveness of the system. The design of the

personalized context-aware IR system is illustrated in Figure 5-7, and its components are described in the following:

A. The Context-Aware User-Model Unit: User modelling for IR is done via GA to evolve and adapt query vectors that are representative models of the user information needs [Goldberg, 1989]. The input to the system is a set of ad-hoc keywords (i.e. query terms). With GA, user model represents theoretical knowledge about the user needs, encoded in a chromosome. The chromosomes are expressed as the ad-hoc keyword with their numerical weight calculated as a function of the frequency of keyword across a document. This best set of keywords is applied in IRS for obtaining the relevant search results. In this study, GA is studied to improve the effectiveness of the IRS components. The vital role for GA in context-aware adaptive system is to find optimal set of documents that best matches the user's interest. This is done by reformulating queries that can adequately identify relevant documents and reject irrelevant documents. Conversely, each of the retrieved documents is given an assessment, interactively by the system user. These two measures are then combined through a context aware adaptive system to derive result adaptation using result scoring to judge the relevance of the document in the document database of competing information needs models. This is a ranking approach termed DROPT. In the application of a GA to IR, one has to provide an evaluation or fitness function for each problem to be solved. Its choice is vital for the GA to function well. The fitness function must be suited to the problem at hand, since the efficiency of the GA will, to a great degree, be determined by how faithfully the fitness function characterizes the function being optimized. In our GA, the definition of our fitness function consists of the rank of appearance of the relevant documents in feedback and the query terms of relevant documents in feedback based on numerical weight. The formal definition of our fitness function is described below. For any weight $w = (w_1, w_2, \dots, w_n)$ in the current document collection \mathbf{N} , its fitness function is calculated by:

$$F = 1 - \frac{n}{N} \quad (5-1)$$

Where n is the number of times the ad-hoc keywords are appearing in the whole document, w is the numerical weight of each document, while N is the total number of documents present in the document collection.

B. Query Unit: We apply context awareness in this study as a technique to this unit to reformulate queries in order to improve the predicted relevance of retrieved document. The query module processes the user query to find the more relevant documents. Consequently,

the matching mechanism retrieves the document which matches the query according to the rules. It searches in the database in which the query terms are stored.

C. Document Database Unit: The document database is a repository of documents from subject areas of experts that are sent to the user for relevance feedback. Each document held by the knowledge database has an associated index, which is a set of keywords that identifies the document. On request, document indices are sent to the Search Agent to be compared against the queries. Document Database stores the best keyword which is generated by Genetic Algorithm.

D. Matching Unit: The comparison of the query against the document representations is called the query process. The matching process results in a list of potentially relevant documents. Users will browse this document list in search of the information they need. The search agent is used to retrieve information in response to an incoming query and return the best matching document according to the rules.

E. The Rule-Based Unit: A user must specify some information, considered as context pertaining to the query. This context (preferences) provides a high-level description of the users information need and eventually control the search strategy used by the system. In this study, we focus on modelling the information using rules that best matches user’s interest to judge the relevance of competing information need models. Such rule states, among a set of conditions, a particular **YES** or **NO** together with a weight. The rules are shown in Table 5-2.

Table 5-2: Relevance Judgments Model (RJM) Table for User Model Judgments

		Matching	
		Y	N
Feedback	Threshold score exceeded?		
	P	HR	HI
	E	HR	HI
	G	HR	HI
	F	MR	MI
	B	LR	LI
	H	LR	LI

Matching values: Yes (Y), No (N)

Feedback values: Perfect (P), Excellent (E), Good (G), Fair (F), Bad (B), Harmful (H)

Relevance Judgment values: Highly Ranked (HR), Moderately Ranked (MR), Lowly Ranked (LR), Lowly Ignored (LI), Moderately Ignored (MI) and Highly Ignored (HI).

Each of the cells in Table 5-2 represent *IF* < *CONDITION*> *THEN* < *ACTION*> *Statement*. Users can express conditions regarding the values of a preference. For example, the first cell in the Table 5-2 above is a statement IF < Matching = Y; Feedback = P > THEN < Judgment = HR >, where Y represents matching condition value "YES", P represents feedback value "Perfect" and "HR" represents relevance judgment value "Highly Ranked" respectively. These judgment rules rely on obtaining information from a domain of expert by scoring each of the retrieved documents. Users provide a judgment of the documents over a scale of [0...100], and the matching is calculated over a scale [0.0...1.0] with feedback values belong to [0.0...1.0] and relevance judgment (output values) were performed on a non-binary manner, where documents were judged on a six-level scale: Highly Ranked (HR), Moderately Ranked (MR), Lowly Ranked (LR), Highly Ignored (HI), Moderately Ignored (MI), or Lowly Ignored (LI).

F. The Ranking Unit: The objective of ranked retrieval is to put the most relevant documents in the top of the ranked list, reducing the time the user has to invest in scanning through the entire documents. This unit ranks the document according to the relevance of the user query. Relevance judgments were performed on query-documents, where documents were judged on a six-level scale. The top ranked documents in the retrieval list are used to form a refined query. The output obtained is the set of best keywords and they represent the possible solutions to the IR problem. The relevance judgment can be carried out as a mean value of judgment after all the ranked documents have been assessed.

5.5.1 Preference relevance feedback of user judgments on documents

The notion of user preference has been discussed in the literature of IR, although its relevance has perhaps not been fully explored. Based on [Yao, 1995] investigation, the concept of user preference is adopted for the measurement of the relevance of documents in this present research study. A user preference relation has been applied in this research to provide a suitable means for “*pairwise comparison of documents*”. Given any two documents $d, d^l \in D$, where D denotes a finite set of documents. We assume that a user is able to decide if one document is more or less relevant than another based on the relevance weight of the

document. Our goal is to establish a basis for the representation of user judgments on the relevance of documents within the normalization interval scale $v \in \{0,1\}$. The user preference relation can be defined by binary relation \succ on D as follows:

$$d > d^l \text{ iff the user prefers } d \text{ to } d^l$$

This expresses user's degree of interest. In this study, however, a rule-based context-aware personalized system associates a set of inputs (conditions) with a set of rules to obtain an output (judgments). The facet level of document judgment was proposed in [Liu et al. 2010], and it includes two values: segment and document. Segment level tasks require locating specific information within a page, while document level tasks only require users to judge if a page is relevant in general but do not necessarily require locating specific information.

The design of the preference values for keyword matching based querying is shown in Figure 5-8. In this regard, the preference relevance feedback of user judgments on documents help users conduct searches iteratively and reformulate search queries to reflect a user's interest.

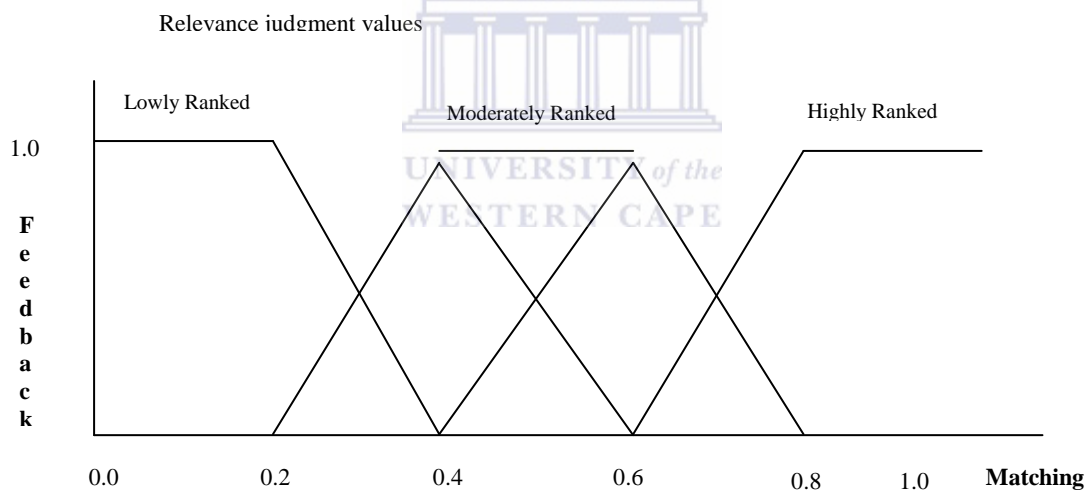


Figure 5-8: Design of Preference value

The design of preference value system is used to adjust the fitness of individual user information need models. It is a rule-based that uses the matching between a search context and a retrieved document, and the user feedback to derive the required fitness modification for the search context. The underlying philosophy of the rules is to rank those documents that the user judges to be relevant to his or her needs, and ignore those user judges to be irrelevant. Therefore, if the user judges a document to be relevant then the fitness of the search results used for retrieval of the document should be highly ranked, and especially more so if the matching measure is low. Conversely, if the user feedback is not relevant but,

the matching value between query and document is high then the fitness of the search context should be low. Users provide an assessment of the documents over a scale of [0...100], and the matching is calculated over a scale of [0.0...1.0]. The preference rules are shown in Table 5-2.

5.6 Sequential document ranking personalization

Document ranking personalization is achieved by incorporating user models via query reformulation and document retrieval as the main task of our proposed system, during which occurs query processing, involving implicit and preferential explicit relevance feedback; later, the search results are categorized according to domain types employed for proper presentation to individual user. Finally, the personalized retrieved document to individual user is based on his/her preferences. This task is illustrated in Figure 5-9 below.

When the user submits an initial query (1) to the system, it passes along a pre-processing phase, resulting in an index terms. Then these terms are taken (a) by the reformulation phase which analyses similar terms based on the Participants Knowledge Domains (PKDs) (2); the analysed terms are used to reformulate the query.

The reformulated query is submitted to the relevance feedback phase (b) in order to perform implicit relevance feedback. To achieve this, the ad-hoc retrieval context information (3) is matched (c) to topical contextual information relevance (4), resulting (d) in a set of contextual relevant documents (5). Then, these documents are analysed using TF-IDF weighting measures to define which terms will reformulate the query.

After the query has been reformulated, it is matched (e) to contextual index entries from the document repository (5), obtaining a list of relevant documents. After that, the documents are ranked (f), via our new DROPT technique. This ranked list is then forwarded (6) to the domain types classification component, which organize the search results according to user preferences of PKDs concepts (7). Then the information relevance sorts its inner results in ranking order according to relevance weights.

After that, the categorized search results according to domain types are presented (8) to the system application. Hence, the context personalized user model (9); personalizes retrieved documents to individual user based on his/her preferences. Once the user selects a concept in the presented document relevance list, the original query is reformulated, taking all documents clustered into the concept as preferential explicit relevance feedback evidences.

To achieve this, all query processing's phases and their dependencies are matched, ranked, then categorized and presented according to relevance weights.

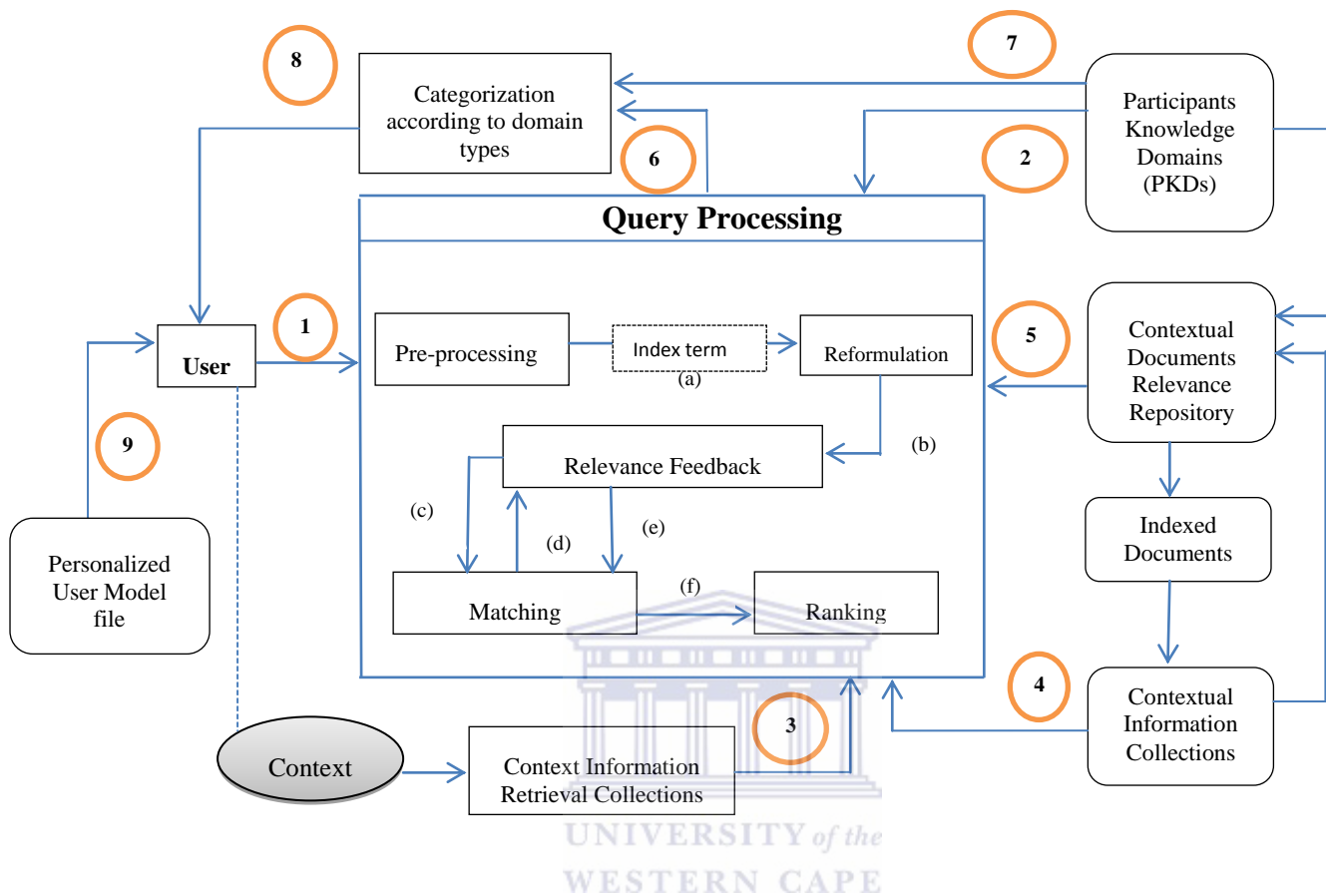


Figure 5-9: Document Ranking Personalization Flow

This iterative user behaviour search concepts repeats each time the user reformulates a concept and only finishes when he explicitly notifies the system application that his information need is satisfied, by choosing to see the ranking of personalized search results.

We apply context awareness in this thesis to reformulate queries in order to improve the predicted relevance of retrieved documents. The process employs user models comprises of categorical terms to represent domain actions and an IR model for personalization, and to index the documents, in order to predict potential relevant documents during ad-hoc retrieval of search results.

5.7 Experimental design

The experiment was designed to study a new user's behaviour source i.e. ranking of retrieved documents that can influence the information retrieval process. Though considering user

searching actions (i.e. clicking on a document in a search result, printing a document, moving a document into a folder, etc.) as sources for implicit relevance of documents, the techniques presented in this thesis is different because it considers document ranking. From that view, the techniques is interesting and innovative as it emphasizes that the IR process is not just about matching between documents and queries but relationships among matching, user actions and user preferences in ranked documents of retrieved results.

The experiment was designed and piloted using systems that allows interactive information retrieval (IIR) experiments that log users 'in different browsers interactive search behavior. The system has a search engine where tables are created for experimental generated data from searching tasks. The systems were used to determine the frequency of keyword matching-based querying results to monitor the progress of the experiment. They performs several information-related tasks activities such as searching, filtering, matching, displaying, and learning information needs over time. This is concerned with the reuse of the existing standards, approaches, and agent technology framework components, and how to incorporate them into the design of the IR system.

During the search, the participant interactions with the search engine were logged via the system log in menu. In each search task, the participants were asked to obtain the frequency of keyword matching based querying across a document; that were relevant to meet their information requests. The behavioural measures we examine are the frequencies of the user issued query (i.e. frequency of keyword matching based querying) while interacting with the IR system.

5.8 Implementation

The first phase is the preprocessing phase for the given query. A query is represented as a set as follows $Q = \{t_1, t_2, t_3, \dots t_n\}$, where Q: a set of user query terms, t = a term of the query

For each domain, our system selects similar queries that were used in the past by the user. The system takes the queries that have similarity values greater than average relevance weight $(\bar{\omega})$. The $\bar{\omega}$ value is decided through experiments.

The second phase is the searching phase for the given query. For each query that was used in the past, the system selects similar documents that were used in the past by the user. The

system takes the documents that have relevance weight values greater than average relevance weight ($\overline{\omega}$).

The third phase is the personalization of search results for each domain knowledge employed. The personalized predictive ranking model identifies retrieved documents to individual user from the domains according to his/her preferences.

The results are shown in Table 5-3. This proposal has been tested with 100 documents; 20 search tasks for each of the employed domain of participants.

Table 5-3: Data Generated by DBD: MySQL, LWP and CAM::PDF

Doc_id	Title	Ad-hoc Keywords
1.	New Directions in Cognitive Information Retrieval	Information Retrieval
2..	Medium access control with mobility-adaptive mechanism for wireless sensor networks	Medium access control
3.	Agent Technology and eHealth	eHealth
4.	Swarm Intelligent Routing Solution for Wireless Sensor Networks	Swarm intelligent
5.	Personalized Web Search by Using Learned User Profile in Re-ranking.	User profile
6.	An online energy efficient routing protocol with traffic load prospects in wireless sensor networks	Traffic load
7	Cluster-Tree based data gathering in wireless sensor networks	Data gathering
8.	Ant colony optimization: Introduction and recent trends	Ant colony optimization
9.	Keyword based context aware selection of natural language query pattern	User Profile
10.	Implicit relevance feedback for context aware information in UbiLearning environment	Relevance feedback
11.	Energy efficient clustering algorithm for wireless sensor networks	Clustering algorithm
12.	Privacy-aware autonomous agents for pervasive healthcare	autonomous agents
13.	On the use of passive clustering in wireless video sensor networks	Passive clustering
14.	Wireless telemedicine and m-health: technologies, applications and research issues	Wireless telemedicine
15.	Intelligent agents: theory and practice	Intelligent agents
16.	Patient monitoring using personal area networks of wireless intelligent sensors	Intelligent sensors
17.	Context-aware retrieval: Exploring a New Environment for information retrieval and information filtering	Information filtering,

18.	Evaluating IRS performance based on user performance	User performance
19.	Wireless sensor networks for home health care	Health care
20.	A heuristic-based methodology for semantic augmentation of user queries	Semantic
21.	Workflow scheduling algorithms for grid computing	Workflow scheduling
22.	Secure group communication in grid environment	Grid environment
23.	Towards Novel And Efficient Security Architecture For Role-Based Access Control In Grid Computing	Efficient security
24.	A Scalable Authorization Approach for Grid System Environments	Authorization
25.	High-performance scientific computing for the masses: developing secure grid portals for scientific workflows	Grid portals
26.	Secure and efficient cryptosystem for smart grid using Homomorphic encryption	Homomorphic encryption
27.	Applicability analysis of grid security mechanisms on cloud networking	Cloud networking
28.	MetaData for efficient, secure and extensible access to data in a medical grid	Medical grid
29.	Integrating Trust into Grid Resource Management Systems	Trust
30.	Manual job submission architecture that considered workload balance among computing resources in the grid interoperation	Interoperation
31.	Data Mining and Visualization of Large Databases	Data Mining
32.	Improving Web search ranking by incorporating user behavior information	Web search
33.	Learning user interaction models for predicting Web search result preferences	User Interaction
34.	Human Information Interaction: An Ecological Approach to Information Behaviour	Information behaviour
35.	Incremental relevance feedback for information filtering	Information filtering
36.	Challenges in information retrieval and language modelling	Language modelling
37.	Design and implementation of a semantic search engine for Portuguese	Semantic search engine
38.	Personalized Access to Contextual Integration by using an Assistant for Query Reformulation	Query reformulation
39.	Context-based Hybrid Methods for User Query Expansion	User query expansion
40.	Personalized access to information by query reformulation based on the state of the current task and user profile	Personalized access
41.	Applications of Software Agent Technology in the Health Care Domain	Software agent
42.	Using Data Mining Predictive Models to Classify Credit Card Applicant	Predictive models
43.	A cognitive perspective on search engine technology and the WWW	Cognitive perspective

44	Developing Multi-Agent Systems with JADE, Wiley Series in Agent Technology	Multi-Agent system
45	Survey of clustering data mining techniques	Clustering data mining
46	Inverted Base File General Metric Space Indexing for Quality Aware Similarity Search in Information Retrieval	Inverted base
47	A Survey of Automatic Query Expansion in Information Retrieval	Automatic query expansion
48	WebMate: a personal agent for browsing and searching	Personal agent
49	User Model for Adaptive Information Retrieval on the Web: Towards an Interoperable and Semantic Model	Adaptive information retrieval
50	Human-computer interaction with mobile devices and services	Mobile devices
51	A contextual evaluation protocol for a session-based personalized search	Contextual evaluation
52	A rule-based approach to content delivery adaptation in web information systems	Rule-based
53	Understanding and Using Context. Personal Ubiquitous Computing	Personal ubiquitous computing
54	Interactive query expansion: a user-based evaluation in a relevance feedback environment	User-based evaluation
55	Hierarchic document clustering using Ward's method.	Hierarchic document clustering
56	Crowdsourcing Document Relevance Assessment with Mechanical Turk	Document relevance
57	'The Effectiveness of Web Search Engines for Retrieving Relevant Ecommerce Links'	Web search engines
58	IR evaluation methods for retrieving highly relevant documents	Evaluation methods
59	Document Ranking and the Vector Space Model	Document ranking
60	Anatomy and empirical evaluation of an adaptive Web-based information filtering system	Adaptive web
61	Learning user interaction models for predicting web search result preferences.	Learning user interaction
62	How does search behavior change as search becomes more difficult?	Search behaviour
63	Using query contexts in information retrieval	Query context
64	Agglomerative clustering of a search engine query log	Agglomerative clustering
65	A user centered experiment and logging framework for interactive information retrieval.	Interactive information retrieval
66	The IIR evaluation model: A framework for evaluation of interactive information retrieval systems.	Evaluation model
67	Task complexity affects information seeking and use	Information seeking

68	Identifying User Goals from Web Search Results.	User goals
69	Modeling user navigation behaviours in a hypermedia based learning system: An individual differences approach.	User navigation behaviour
70	Summarizing local context to personalize global web search.	Local context
71	Personalized query expansion for the web.	Personalized query expansion
72	Dynamic Assessment of Information Acquisition Effort during Interactive Search.	Interactive search
73	Usefulness as the criterion for evaluation of interactive information retrieval	Usefulness
74	Issues of context in information retrieval (IR): an introduction to the special issue.	Information retrieval
75	A large-scale evaluation and analysis of personalized search strategies.	Search strategies
76	Evaluating implicit measures to improve web search	Implicit measures
77	Learning users' interests by unobtrusively observing their normal behavior.	Learning users interest
78	Beyond Dwell Time: Estimating Document Relevance from Cursor Movements and other Post-click Searcher Behavior.	Document relevance
79	Relevant term suggestion in interactive web search based on contextual information in query session logs	Contextual information
80	Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search.	Implicit feedback
81	A field study characterizing web-based information-seeking tasks.	Information seeking
82	Display time as implicit feedback: Understanding task effects.	Task effects
83	The effects of topic familiarity on information search behavior.	Topic familiarity
84	A comparison of query and term suggestion features for interactive searching	Interactive searching
85	Implicit feedback for inferring user preference: A bibliography.	Inferring user preference
86	Applying collaborative filtering to UseNet news.	Collaborative filtering
87	Automatic identification of user goals in Web search.	Autonomous identification
88	Evaluating and optimizing autonomous text classification systems	Autonomous text classification
89	A faceted approach to conceptualizing tasks in information seeking	Faceted approach
90	Analysis of Query Reformulation Types on Different Search Tasks.	Search tasks
91	Helping identify when users find useful documents: Examination of query reformulation intervals.	Useful documents

92	Analysis and Evaluation of Query Reformulations in Different Task Types	Task types
93	Personalizing information retrieval for multi-session tasks: The roles of task stage and task type.	Multi-session tasks
94	Personalized web search by mapping user queries to categories.	Mapping user queries
95	Information-seeking strategies of novices using a full-text electronic encyclopaedia.	Information seeking strategies
96	Information Filtering Based on User Behavior Analysis and Best MatchText Retrieval.	User behaviour analysis
97	Query Chains: Learning to rank from implicit feedback.	Implicit feedback
98	Relevance feedback in information retrieval.	Relevance feedback
99	Modeling Information Content Using Observable Behavior	Modelling information
100	Study of the usefulness of known and new implicit indicators and their optimal combination for accurate inference of users interests.	Implicit indicators

From Table 5-3, the required information is extracted in terms of *Doc_id*, *keywords* and *weight (tf)*; calculated as a function of the frequency of keyword across a document, hence stored separately for convenience as shown in Tables 5-4, 5-5, 5-6, 5-7 and 5-8 respectively. These ad-hoc keywords represent domain of knowledge of the system users' participants in the area of different subjects in Computer Science at UWC, Ice Box Research Laboratory.

5.8.1 Results

In order to generate the prediction user context, we used the DROPT algorithm to calculate the relevance weights for retrieved documents. For ranking, we combined ranking of all participants.

The average relevance weights of individual users were obtained ($\bar{w} = 0.866$ for Domain 1, $\bar{w} = 0.912$ for Domain 2, $\bar{w} = 0.899$ for Domain 3, $\bar{w} = 0.846$ for Domain 4, and $\bar{w} = 0.845$ for Domain 5. The overall average relevance weight, $\bar{w} = 0.874$ was obtained for the 5 Domains of participants combined. Thus for Domain 1, any document whose value was higher than **0.866** would be predicted for ranking as a "relevant" document, and marked 'X'; and any document with a lower value would be predicted but ignored if found to be "irrelevant" later (Table 5-4). Also, for Domain 2, any document whose value was higher than **0.912** would be predicted for ranking as a 'relevant' document and marked 'X'; and any document with a lower value would be predicted but ignored if found to be "irrelevant" later (Table 5-5). For Domain 3, any document whose value was higher than **0.899** would be predicted for ranking as a "relevant" document, and marked 'X'; and any document with a

lower value would be predicted but ignored if found to be "irrelevant" later (Table 5-6). For Domain 4, any document whose value was higher than **0.846** would be predicted for ranking as a "relevant" document, and marked 'X'; and any document with a lower value would be predicted but ignored if found to be "irrelevant" later (Table 5-7). Lastly, for Domain 5, any document whose value was higher than **0.845** would be predicted for ranking as a "relevant" document, and marked 'X'; and any document with a lower value would be predicted but ignored if found to be "irrelevant" later (Table 5-8). We generated five prediction models; each from domain of participants with different generated data from the user behaviour attributes measure when the matching tasks were considered during interaction mode. This shows that any document whose value was higher than **0.876** would be predicted for ranking performance results at known "relevant" document, and marked 'X'; and any document with a lower value would be predicted but ignored if found to be irrelevant later (Table 6-3) for analysis on ranking performance results.

The goal is to appropriately predict "relevant documents" for ranking performance results based on user preference. Therefore, we measured precision and recall of relevant documents, marked 'X' as explained comprehensively in the next Chapter. The context-based IR system and algorithm developed demonstrates promising results attributes of the user behaviour. The detailed statistical analysis of the generated data is discussed in Section 6.6 of the next Chapter.

Table 5-4: Derived Information for Ranking Prediction from Domain of participant 1

Information					
Doc #	Keywords	Weight (tf)	Relevant	Fitness score	Avg. fitness score
9.	Query pattern	5	X	0.95	≥ 0.866
10.	Relevance Feedback	5	X	0.95	≥ 0.866
17.	Information filtering	6	X	0.94	≥ 0.866
5.	User Profile	8	X	0.92	≥ 0.866
3.	e-Health	8	X	0.92	≥ 0.866
18.	User Preference	9	X	0.91	≥ 0.866
15.	Intelligent agents	10	X	0.90	≥ 0.866
12.	Autonomous agents	13	X	0.87	≥ 0.866
20.	Semantic	18		0.82	
1.	Information Retrieval	19		0.81	
31	Data Mining	10	X	0.90	≥ 0.866
32	Web search	17		0.83	
33	User Interaction	7	X	0.93	≥ 0.866
34	Information behaviour	14		0.86	
35	Information filtering	28		0.72	
36	Language modelling	23		0.77	
37	Semantic search engine	25		0.75	
38	Query reformulation	12	X	0.88	≥ 0.866
39	User query expansion	9	X	0.91	≥ 0.866
40	Personalized access	22		0.78	
Average fitness score →					0.866

The values displayed in Table 5-4 shows the results of the search system for documents retrieved from a search engine back end prototype.

Table 5-5: Derived Information for Ranking Prediction from Domain of participant 2

Information					
Doc #	Keywords	Weight (tf)	Relevant	Fitness score	Avg. fitness score
4.	Swarm intelligent	2	X	0.98	≥ 0.912
7.	Data gathering	2	X	0.98	≥ 0.912
6.	Traffic load	3	X	0.97	≥ 0.912
2.	Medium access control	3	X	0.97	≥ 0.912
13.	Passive clustering	3	X	0.97	≥ 0.912
16.	Intelligent sensors	3	X	0.97	≥ 0.912
14.	Wireless telemedicine	4	X	0.96	≥ 0.912
11.	Clustering algorithm	4	X	0.96	≥ 0.912
8.	Ant colony optimization	5	X	0.95	≥ 0.912
19.	Health care	16		0.84	
41	Software agent	15		0.85	
42	Predictive models	5	X	0.95	≥ 0.912
43	Cognitive perspective	7	X	0.93	≥ 0.912
44	Multi-Agent system	8	X	0.92	≥ 0.912
45	Clustering data mining	18		0.82	
46	Inverted base	12		0.88	
47	Automatic query expansion	10		0.90	
48	Personal agent	19		0.81	
49	Adaptive information retrieval	24		0.76	
50	Mobile devices	14		0.86	
Average fitness score →					0.912

The values displayed in Table 5-5 shows the results of the search system for documents retrieved from a search engine back end prototype.

Table 5-6: Derived Information for Ranking Prediction from Domain of participant 3

Information					
Doc #	Keywords	Weight (tf)	Relevant	Fitness score	Avg. fitness score
21.	Workflow scheduling	13		0.87	
22.	Grid environment	2	X	0.98	≥ 0.899
23.	Efficient security	4	X	0.96	≥ 0.899
24.	Authorization	2	X	0.98	≥ 0.899
25.	Grid portals	4	X	0.96	≥ 0.899
26.	Homomorphic Encryption	14		0.86	
27.	Cloud networking	2	X	0.98	≥ 0.899
28.	Medical grid	2	X	0.98	≥ 0.899
29.	Trust	2	X	0.98	≥ 0.899
30.	Interpolation	8	X	0.92	≥ 0.899
51	Contextual evaluation	5	X	0.95	≥ 0.899
52	Rule-based	16		0.84	
53	Personal ubiquitous computing	22		0.78	
54	User-based evaluation	18		0.82	
55	Hierarchic document clustering	10		0.90	
56	Document relevance	24		0.76	
57	Web search engines	3	X	0.97	≥ 0.899
58	Evaluation methods	22		0.78	
59	Document ranking	21		0.79	
60	Adaptive web	8	X	0.92	≥ 0.899
Average fitness score →					0.899

The values displayed in Table 5-6 shows the results of the search system for documents retrieved from a search engine back end prototype.

Table 5-7: Derived Information for Ranking Prediction from Domain of participant 4

Information					
Doc #	Keywords	Weight (tf)	Relevant	Fitness score	Avg. fitness score
61.	Learning user interaction	18	X	0.82	
62.	Search behaviour	7	X	0.93	≥ 0.846
63.	Query context	14	X	0.86	≥ 0.846
64.	Agglomerative clustering	22		0.78	
65.	Interactive information retrieval	3	X	0.97	≥ 0.846
66.	Evaluation model	24		0.76	
67.	Information seeking	9	X	0.91	≥ 0.846
68.	User goals	3	X	0.91	≥ 0.846
69.	User navigation behaviour	22		0.78	
70.	Local context	10	X	0.90	≥ 0.846
71.	Personalized query expansion	6	X	0.94	≥ 0.846
72.	Interactive search	26		0.74	
73.	Usefulness	12	X	0.88	≥ 0.846
74.	Information retrieval	28		0.72	
75.	Search strategies	17		0.83	
76.	Implicit measures	25		0.75	
77.	Learning users interest	13	X	0.87	≥ 0.846
78.	Document relevance	24		0.76	
79.	Contextual information	19		0.81	
80.	Implicit feedback	11	X	0.89	≥ 0.846
Average fitness score \longrightarrow					0.846

The values displayed in Table 5-7 shows the results of the search system for documents retrieved from a search engine back end prototype.

Table 5-8: Derived Information for Ranking Prediction from Domain of participant 5

Information					
Doc #	Keywords	Weight (tf)	Relevant	Fitness score	Avg. fitness score
81.	Information seeking	28		0.72	
82.	Task effects	12	X	0.88	≥ 0.845
83.	Topic familiarity	11	X	0.89	≥ 0.845
84.	Interactive searching	7	X	0.93	≥ 0.845
85.	Inferring user preference	5	X	0.95	≥ 0.845
86.	Collaborative filtering	14		0.86	≥ 0.845
87.	Autonomous identification	19		0.81	
88.	Autonomous text classification	22		0.78	
89.	Faceted approach	8	X	0.92	≥ 0.845
90.	Search tasks	28		0.72	
91.	Useful documents	20		0.80	
92.	Task types	26		0.74	
93.	Multi-session tasks	25		0.75	
94.	Mapping user queries	8		0.92	
95.	Information seeking strategies	13	X	0.87	≥ 0.845
96.	User behaviour analysis	6	X	0.94	≥ 0.845
97.	Implicit feedback	13	X	0.87	≥ 0.845
98.	Relevance feedback	14	X	0.86	≥ 0.845
99.	Modelling information	10	X	0.90	≥ 0.845
100.	Implicit indicators	21		0.79	
Average fitness score →					0.845

The values displayed in Table 5-8 shows the results of the search system for documents retrieved from a search engine back end prototype.

5.8.2 Discussion

Our results on the indexed ad-hoc keywords represent domain of the system user's five participants in an in-lab experimental setting. The results demonstrate that combining individual system user's behavioural measures can improve ranking prediction accuracy (according to relevance weights), for documents ranking tasks, and however that individual users ranking performed much better than combining document rankings of the systems. This accomplishes personalization of retrieved documents for individual users as the focus of this thesis. The retrieval effectiveness is measured using well known metrics *Precision* and *Recall*, at known relevant documents. Also ranking performance results is discussed in detailed between the relevance judgment values during performance evaluation in the next Chapter.

5.9 Chapter Summary

In this Chapter the development process consist of requirement analysis, requirement determination, context-aware agent, system architecture and agent level design stages. Each stage is provided with suitable modelling tools: Use Cases for requirements analysis, sequence diagrams in system design and context-aware agents to pro-actively act on behalf of users. The incorporation of context-aware IR model for personalized retrieval of documents is discussed in adaptive Web application environment. In addition, we presented an overview of the prototype implementation of the proposed system. Firstly, search engine back end prototype was developed for dynamic process environment. The information search process demonstrated an interactive process between information source and information system users', and in particular current users' interactive behaviours present IR systems generated data to understand the user search context. We believe the methods and results of this study will provide us a better comprehension of how user behaviours can assist us to acquire search context and to personalize search results using a predictive user document ranking model. The Discussion and comparison of seven context-aware agent's interaction was carried out in a given environment and context, from which clarification was illustrated from the adaptive Web IR application environment. Context awareness was employed as a technique to reformulate queries to satisfy the functionalities of the proposed system. The proposed system and developed algorithm are evaluated in the next chapter.

Chapter 6

System Evaluation

6.1 Evaluation methodology

The efficacy of the IRL technique is determined in terms of two performance measures: Ranking and retrieval performance. *Ranking performance* involves inferring a scoring function to carry out query reformulations with the same created in diverse contexts, in order to identify the DROPT parameters. To conduct this evaluation the query is reformulated in different contexts within the domain of the system users. For each experiment, different DROPT parameters are selected. We examine the effectiveness of personalized search system using manually selected information needs as a testbed for comparing retrieval system and algorithms developed in this study. Individual user information such as queries submitted, results returned (title), document identity, weight of the document and URL selected from results returned is collected. To evaluate the effectiveness of the system; user's feedback is evaluated by requiring explicit judgments by an Online Interactive Reinforcement Learning Retrieval Prototype (OIRLRP): a context aware personalized search system. The effectiveness of the performance measures is evaluated in terms of precision and recall of the system. Each query was designed to retrieve "top n documents", which were judged by system users participant in each of the subject areas for 5 successive retrievals. These query terms represent the domain of knowledge of the system user.

To show that the learned retrieval function improves retrieval, an interactive experiment was conducted by five different system users participants. The evaluation experiments were conducted with a collection of 20 queries (each query represents a user profile) and 100 representative documents. This experiment allows the system users to test the retrieval effectiveness of the documents retrieved. Tables 5-4, 5-5, 5-6, 5-7 and 5-8 were provided for ranking performance results (evaluation) of the retrieved documents from the domain of participants.

The other documents in the collection were selected from other areas which can have overlapping contextual preferences, for example, contextual information” contains the world, contextual”, which can apply to a contextual preferences as well. A summary of the document databases is given in Table 6-1.

Table 6-1: Documents Collection

Subjects areas	Number of relevant documents in the collection
Information retrieval	30
Search engine	18
Context-aware information retrieval	22
Agent/Multi-agent	20
Knowledge Management	13
Cluster Analysis	15
Grid Computing	12
Machine Learning	17
Wireless Sensor Networks	15
Evolutionary Algorithms	18
Others	20
Total	200

6.2 Evaluation metrics

Ranking algorithm evaluate relevance over accepted IR metrics, namely *Precision at n* ($P(n)$), *Recall at n* ($R(n)$), and *Mean Average Precision* (MAP). Each metric focuses on a different aspect of system performance, as we describe below.

Precision at n: $P(n)$ measures the fraction of documents ranked in the top n results that are labelled as relevant. In our setting, we require a relevant document to be labelled “Perfect”, ‘Excellent’ ‘Good’ or ‘Fair’. The position of relevant documents within the top n is irrelevant, and hence this metric measure overall user satisfaction with the top n results.

MAP: an information retrieval performance measure that combines precision and recall and rewards relevant documents ranked higher in the list of retrieved documents. It is computed as the average of the precision values for each relevant document in the ranked results.

Recall at n: $R(n)$ which measures the fraction of retrieved relevant documents within the top n documents over the total number of relevant documents in the document collection.

6.3 Ranking performance results

In order to measure the performance of the DROPT technique search, each query produced a document based on the matching conditions and the retrieval was repeated for 20 query reformulations from the domain of system user experts. The underlying philosophy of the relevance judgment rules is to rank those documents, which exceeded the overall weighted fitness score that the system user judges to be relevant to his/her information needs, and ignore those documents the system users judges to be irrelevant (less preferred). Participants provided a judgment of the documents over a scale of [0...100] and the matching value is calculated over a scale of [0.0... 1.0]. Figure 6-1 shows a ranked list that help the user fill their information needs. Table 6-2 shows the MAP results and Table 6-3 shows the precision results at known relevant documents for ranking performance from the domain of participants.

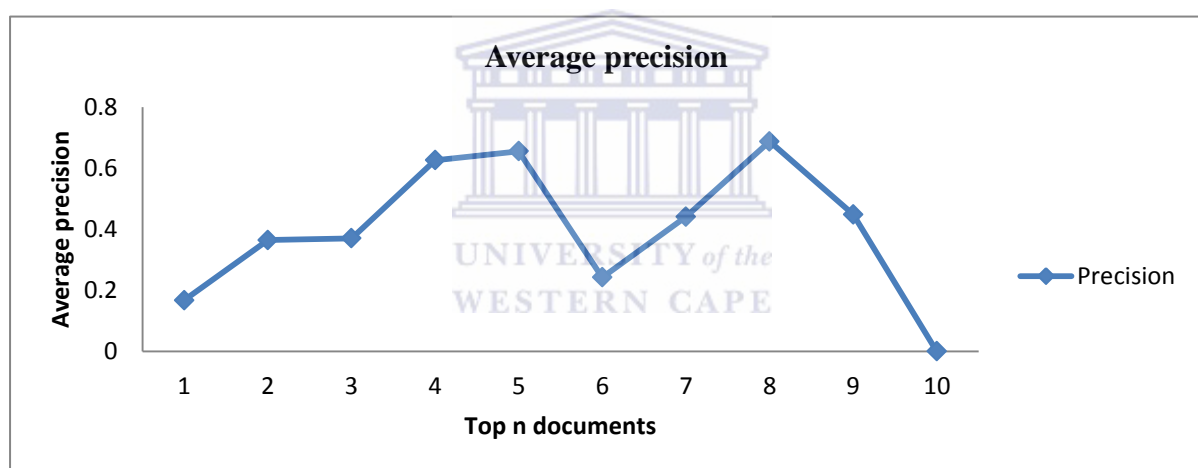


Figure 6-1: Average precision Graph for ranking performance results

Table 6-2: Mean average precision results for ranking performance from 5 domain of experts

Generations	1	2	3	4	5	6	7	8	9	10
MAP	0.167	0.364	0.370	0.626	0.655	0.242	0.441	0.687	0.448	0.000

As indicated in Table 6-3, scores that falls below overall weighted fitness values (0.876) for the ranking parameter do not show significant ranking improvement. This is because at low ranking scores below this value, irrelevant documents are rejected by the system user participants. From the user interaction mode, domain knowledge, topic familiarity and search knowledge was at the peak when distribution of relevant documents in each of the employed domains for the participant varies. This demonstrate when domain knowledge was at peak,

participant integrated diverse concepts in their searches but made fewer changes to their searches. It was discovered when domain was low, participant did more search, selected less efficient concepts in the search and made errors in the query reformulation. Also the domain search behaviour generated more query terms from participant 1, 4, and 5 compared to 2 and 3. This is because a term that is important to one participant is sometimes not important to others. It was noticed that the effect of user information search behaviour with search topics increased as participant reading time decreased, while search efficiency increased. Thus user knowledge about a topic increases as participant go through phases of searching. The difference in distributions shows how individual users search results by acquired context information during ad-hoc retrieval to predict potential relevant documents. This adapts and explore new domain for potentially relevant documents. When the environment of the adaptive system changes the highest ranked documents of interest automatically adjust to the new environment. The best ranking performance of the system is given by medium values between (0.857-0.909) of the precision values. As shown in Figure 6-2 the system is more stable for ranking parameter value of 0.909 from domain of participant 3 and, the number of ranked relevant documents in the search result is also noticeably higher than for the other ranking parameter values from domain of participants 1, 2, 4, and 5. Also considering Figure 6-2, which shows the total ranked relevant documents retrieved in the 56 search processes, the ranking performance of 0.95 has the highest number of ranked documents retrieved from domain of the five participants.

Table 6-3: Precision results for ranking performance at known relevant documents

Document #	Queries	Relevant	Tf	Precision	Fitness score
1	Information retrieval		19	0.000	0.68
2	Medium access control	X	3	0.500	0.95
3	E-health	X	8	0.500	0.87
4	Swarm intelligent	X	2	0.570	0.97
5	User profile	X	8	0.667	0.87
6	Traffic load	X	3	0.667	0.95
7	Data gathering	X	2	0.750	0.97
8	Ant colony optimization	X	5	0.800	0.92
9	Query pattern	X	5	0.750	0.92
10	Relevance feedback	X	5	0.800	0.92
11	Clustering algorithm	X	4	0.833	0.93
12	Autonomous agent		13	0.833	0.78

13	Passive algorithm	X	3	0.857	0.95
14	Wireless telemedicine	X	4	0.875	0.93
15	Intelligent agents	X	10	0.857	0.83
16	Intelligent sensors	X	3	0.889	0.95
17	Information filtering	X	6	0.875	0.90
18	User preference	X	9	0.889	0.85
19	Health care		16	0.000	0.73
20	Semantic		18	0.000	0.70
21	Workflow scheduling		13	0.000	0.78
22	Grid environment	X	2	0.500	0.96
23	Efficient security	X	4	0.667	0.93
24	Authorization	X	2	0.750	0.96
25	Grid portals	X	4	0.800	0.93
26	Homomorphic Encryption		14	0.000	0.77
27	Cloud networking	X	2	0.857	0.96
28	Medical grid	X	2	0.875	0.96
29	Trust	X	2	0.889	0.96
30	Interoperation	X	8	0.900	0.87
31	Data Mining	X	10	0.909	0.83
32	Web search		17	0.000	0.71
33	User Interaction	X	7	0.923	0.88
34	Information behaviour		14	0.000	0.77
35	Information filtering		28	0.000	0.53
36	Language modelling		23	0.000	0.62
37	Semantic search engine		25	0.000	0.58
38	Query reformulation	X	12	0.944	0.80
39	User query expansion	X	9	0.947	0.85
40	Personalized access		22	0.000	0.63
41	Software agent		15	0.000	0.75
42	Predictive models	X	5	0.917	0.92
43	Cognitive perspective	X	7	0.923	0.88
44	Multi-Agent system	X	8	0.929	0.87
45	Clustering data mining		18	0.000	0.70
46	Inverted base		12	0.000	0.80
47	Automatic query expansion	X	10	0.941	0.84
48	Personal agent		19	0.000	0.63
49	Adaptive information retrieval		24	0.000	0.60

50	Mobile devices		14	0.000	0.76
51	Contextual evaluation	X	5	0.909	0.92
52	Rule-based		16	0.000	0.73
53	Personal ubiquitous computing		22	0.000	0.63
54	User-based evaluation		18	0.000	0.70
55	Hierarchic document clustering		10	0.000	0.83
56	Document relevance		24	0.000	0.60
57	Web search engines	X	3	0.941	0.93
58	Evaluation methods		22	0.000	0.63
59	Document ranking		21	0.000	0.65
60	Adaptive web	X	8	0.950	0.87
61	Learning user interaction	X	18	0.000	0.82
62	Search behaviour	X	7	0.500	0.93
63	Query context	X	14	0.332	0.86
64	Agglomerative clustering		22	0.000	0.78
65	Interactive information retrieval	X	3	0.400	0.97
66	Evaluation model		24	0.000	0.76
67	Information seeking	X	9	0.500	0.91
68	User goals	X	3	0.444	0.91
69	User navigation behaviour		22	0.000	0.78
70	Local context	X	10	0.450	0.90
71	Personalized query expansion	X	6	0.363	0.94
72	Interactive search		26	0.000	0.74
73	Usefulness	X	12	0.385	0.88
74	Information retrieval		28	0.000	0.72
75	Search strategies		17	0.000	0.83
76	Implicit measures		25	0.000	0.75
77	Learning users interest	X	13	0.470	0.87
78	Document relevance		24	0.000	0.76
79	Contextual information		19	0.000	0.81
80	Implicit feedback	X	11	0.500	0.89
81	Information seeking		28	0.000	0.72
82	Task effects	X	12	0.500	0.88
83	Topic familiarity	X	11	0.333	0.89
84	Interactive searching	X	7	0.250	0.93
85	Inferring user preference	X	5	0.200	0.95

86	Collaborative filtering		14	0.000	0.86
87	Autonomous identification		19	0.000	0.81
88	Autonomous classification text		22	0.000	0.78
89	Faceted approach	X	8	0.444	0.92
90	Search tasks		28	0.000	0.72
91	Useful documents		20	0.000	0.80
92	Task types		26	0.000	0.74
93	Multi-session tasks		25	0.000	0.75
94	Mapping user queries		8	0.000	0.92
95	Information seeking strategies	X	13	0.600	0.87
96	User behaviour analysis	X	6	0.563	0.94
97	Implicit feedback	X	13	0.523	0.87
98	Relevance feedback	X	14	0.500	0.86
99	Modelling information	X	10	0.474	0.90
100	implicit indicators		21	0.000	0.79
Average				0.529	0.845

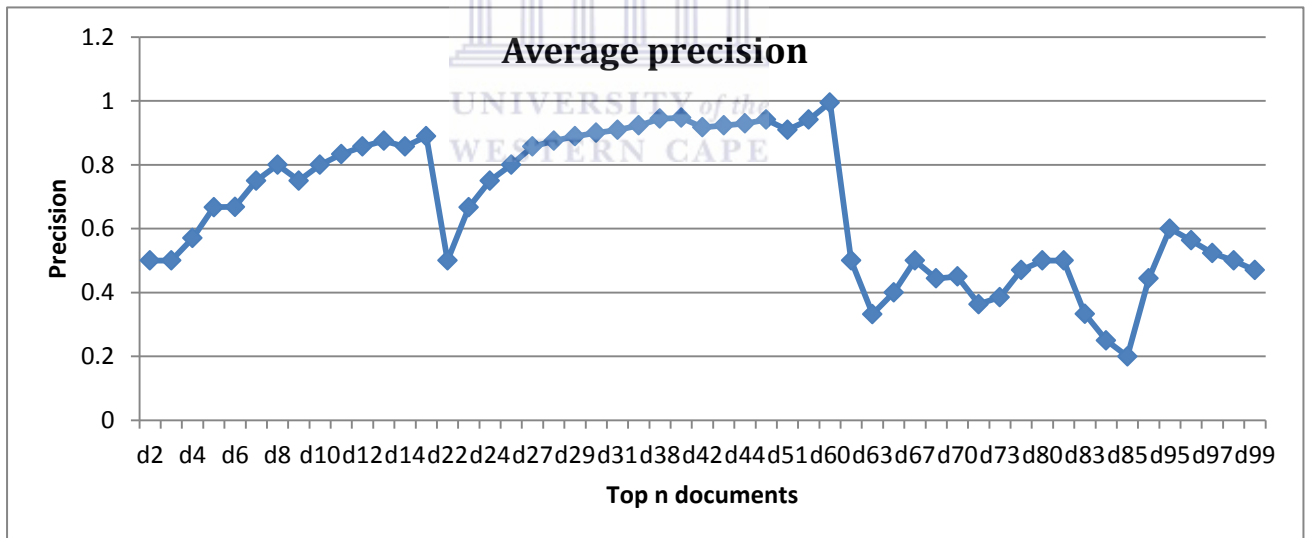


Figure 6-2: Precision Graph for ranking performance results at known relevant documents

6.4 Retrieval results

In this research, a comparison was made between the retrieval performance of traditional relevance feedback and an IRL method based on a DROPT technique, which is combination

of human interactive relevance feedback and context awareness. In this study, we apply context awareness as a technique to reformulate the queries in order to improve the predicted relevance of the retrieved documents. Both methods were tested under the same experimental conditions. We propose measurements namely; preference relevance feedback that ranks a matching value with a feedback value. The technique is interesting and innovative as it emphasizes that the IR process also involves relationships among matching, user actions and user preferences in ranked documents of retrieval results. The standard methods for calculating precision and recall are based on a binary measure of relevance; while in the proposed system ranked items are calculated using scoring approach to calculate the overall weighted fitness score based on equation (6-1). Table 6-4 shows the weighting of the user relevance feedback.

The fitness function of the chromosome (document) used is calculated by:

$$F = 1 - \frac{n}{N} \quad (6-1)$$

Where n is the number of times the (query terms) ad-hoc keywords are appearing in the whole document while N is the total number of documents present in the document collection (corpus).

Table 6-4: Feedback weight values

Relevant Judgment	ω
Perfect	1.0
Excellent	0.8
Good	0.6
Fair	0.4
Bad	0.2
Harmful	0.0

Retrieval effectiveness was demonstrated through a recall-precision graph. For the purpose of comparison, recall and precision graphs were constructed for the two different information retrieval methods, using *cut-off* of 15. A cut-off is a rank that defines the minimal retrieval set. Tables 6-5, 6-6, 6-7, 6-8 and 6-9 show the recall and precision result for 20 generations for IER and RF methods from the domain of experts.

Table 6-5: Precision and recall values for ranking performance at known relevant document

	Domain of Participant 1: Queries				
Document #	Queries	Relevant	Recall	Precision	Fitness score
1	Information retrieval		0.000	0.000	0.68
3	E-health	X	0.000	0.500	0.87
5	User profile	X	0.000	0.667	0.87
9	Query pattern	X	0.316	0.750	0.92
10	Relevance feedback	X	0.368	0.800	0.92
12	Autonomous agent	X	0.000	0.833	0.78
15	Intelligent agents	X	0.000	0.857	0.83
17	Information filtering	X	0.632	0.875	0.90
18	User preference	X	0.000	0.889	0.85
20	Semantic		0.000	0.000	0.70
31	Data Mining	X	0.850	0.909	0.83
32	Web search		0.000	0.000	0.71
33	User Interaction	X	0.912	0.923	0.88
34	Information behaviour		0.000	0.000	0.77
35	Information filtering		0.000	0.000	0.53
36	Language modelling		0.000	0.000	0.62
37	Semantic search engine		0.000	0.000	0.58
38	Query reformulation	X	0.567	0.944	0.80
39	User query expansion	X	0.654	0.947	0.85
40	Personalized access		0.000	0.000	0.63

Table 6-6: Precision and recall values for ranking performance at known relevant documents

Domain of Participant 2: Queries					
Document #	Queries	Relevant	Recall	Precision	Fitness score
2	Medium access control	X	0.053	0.500	0.95
4	Swarm intelligent	X	0.111	0.570	0.97
6	Traffic load	X	0.158	0.667	0.95
7	Data gathering	X	0.211	0.750	0.97
8	Ant colony optimization	X	0.263	0.800	0.92
11	Clustering algorithm	X	0.421	0.833	0.93
13	Passive clustering	X	0.474	0.857	0.95
14	Wireless telemedicine	X	0.526	0.875	0.93
16	Intelligent sensors	X	0.579	0.889	0.95
19	Health care		0.000	0.000	0.73
41	Software agent		0.000	0.000	0.75
42	Predictive models	X	0.778	0.917	0.92
43	Cognitive perspective	X	0.782	0.923	0.88
44	Multi-Agent system	X	0.729	0.929	0.87
45	Clustering data mining		0.000	0.000	0.70
46	Inverted base		0.000	0.000	0.80
47	Automatic query expansion	X	0.785	0.941	0.84
48	Personal agent		0.000	0.000	0.63
49	Adaptive information retrieval		0.000	0.000	0.60
50	Mobile devices		0.000	0.000	0.76

Table 6-7: Precision & recall values for ranking performance at known relevant documents

Domain of Participant 3: Queries					
Document #	Queries	Relevant	Recall	Precision	Fitness score
21	Workflow scheduling		0.000	0.000	0.78
22	Grid environment	X	0.684	0.500	0.96
23	Efficient security	X	0.737	0.667	0.93
24	Authorization	X	0.789	0.750	0.96
25	Grid portals	X	0.842	0.800	0.93
26	Homomorphic Encryption		0.000	0.000	0.77
27	Cloud networking	X	0.895	0.857	0.96
28	Medical grid	X	0.947	0.875	0.96
29	Trust	X	1.000	0.889	0.96
30	Interoperation	X	0.675	0.900	0.87
51	Contextual evaluation	X	0.595	0.909	0.92
52	Rule-based		0.000	0.000	0.73
53	Personal ubiquitous computing		0.000	0.000	0.63
54	User-based evaluation		0.000	0.000	0.70
55	Hierarchic document clustering		0.000	0.000	0.83
56	Document relevance		0.000	0.000	0.60
57	Web search engines	X	0.745	0.941	0.93
58	Evaluation methods		0.000	0.000	0.63
59	Document ranking		0.000	0.000	0.65
60	Adaptive web	X	0.824	0.950	0.87

Table 6-8: Precision & recall values for ranking performance at known relevant documents

Domain of Participant 4: Queries					
Document #	Queries	Relevant	Recall	Precision	Fitness score
61	Learning user interaction		0.000	0.000	0.82
62	Search behaviour	X	0.604	0.500	0.93
63	Query context	X	0.490	0.332	0.86
64	Agglomerative clustering		0.000	0.000	0.78
65	Interactive information retrieval	X	0.789	0.400	0.97
66	Evaluation model		0.000	0.000	0.76
67	Information seeking	X	0.845	0.500	0.91
68	User goals	X	0.477	0.444	0.91
69	User navigation behaviour		0.000	0.000	0.78
70	Local context	X	0.756	0.450	0.90
71	Personalized query expansion	X	0.513	0.363	0.94
72	Interactive search		0.000	0.000	0.74
73	Usefulness	X	0.000	0.385	0.88
74	Information retrieval		0.000	0.000	0.72
75	Search strategies		0.000	0.000	0.83
76	Implicit measures		0.000	0.000	0.75
77	Learning users interest	X	0.475	0.470	0.87
78	Document relevance		0.000	0.000	0.76
79	Contextual information		0.000	0.000	0.81
80	Implicit feedback	X	0.645	0.500	0.89

Table 6-9: Precision & recall values for ranking performance at known relevant documents

Domain of Participant 5: Queries					
Document #	Queries	Relevant	Recall	Precision	Fitness score
81	Information seeking		0.000	0.000	0.72
82	Task effects	X	0.841	0.500	0.88
83	Topic familiarity	X	0.647	0.333	0.89
84	Interactive searching	X	0.892	0.250	0.93
85	Inferring user preference	X	0.724	0.200	0.95
86	Collaborative filtering		0.000	0.000	0.86
87	Autonomous identification		0.000	0.000	0.81
88	Autonomous text classification		0.000	0.000	0.78
89	Faceted approach	X	0.430	0.444	0.92
90	Search tasks		0.000	0.000	0.72
91	Useful documents		0.000	0.000	0.80
92	Task types		0.000	0.000	0.74
93	Multi-session tasks		0.000	0.000	0.75
94	Mapping user queries		0.000	0.000	0.92
95	Information seeking strategies	X	0.456	0.600	0.87
96	User behaviour analysis	X	0.612	0.563	0.94
97	Implicit feedback	X	0.407	0.523	0.87
98	Relevance feedback	X	0.417	0.500	0.86
99	Modelling information	X	0.734	0.474	0.90
100	Implicit indicators		0.000	0.000	0.79

The values displayed in figure 6-3 shows the 100 search results of the system for documents retrieved. Documents are sorted and were set in ascending order of Retrieval Status Values (RSV). Hence, any document whose relevance weight was higher than Average Fitness Weight (AFW) **0.874** as shown in Figure 6-3 would be predicted as a "relevant" document and ranked accordingly; and any document with a lower value would be predicted as an "irrelevant" document. In this respect, 56 documents are ranked and given to users to meet their information needs. Conversely, 44 retrieved documents (fall below AFW) are rejected by the users (not displayed) as shown in figure 6-3.

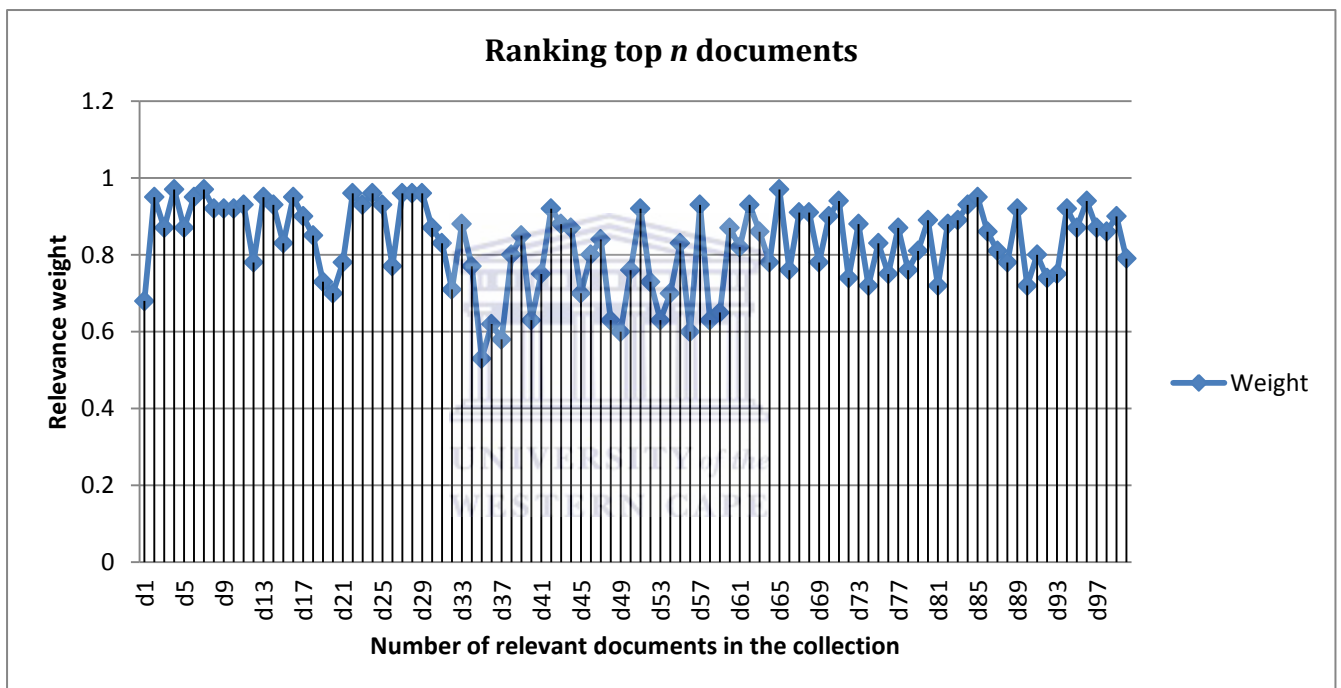


Figure 6-3: Ranking performance Graph results at the known relevant documents

6.5 Experimental results of DROPT technique

For comparison of algorithms, we have used "Precision at position n" (P@n) metrics [Jarvelin & Kekalainen, 2000]. Precision at n measures the relevancy of the top n results of the ranking list with respect to a given query (equation 6-2).

$$P@n = \frac{\text{No of relevant documents in top } n \text{ results}}{n} \quad (6-2)$$

P@n can only handle cases with binary judgment "relevant" or "irrelevant" with respect to a given query at rank n. To compute P@n, 100 queries were judged in these 6 levels by users.

For the evaluation of our algorithm we conducted the following tests. The test process involves using the 100 queries provided by the system users. The measure (P@n) is used for evaluation. We compute them for each query and then take the average dimension (n) for all queries. Figure 6-4 shows comparison of the DROPT algorithm with other algorithms in the P@n measure. As the figure shows, our adaptive algorithm outperforms the others. DROPT technique achieves a 45.6% in P@n compared to BM25 which is the best one of the other. The figure compares the precision for these 20 queries set between the TF-IDF, BM25 and DROPT. The technique is interesting and innovative as it emphasizes that the IR process also involves relationships among matching, user actions and user preferences in ranked documents of retrieval results. It shows that the precision value of the proposed ranking technique is comparatively higher for all the query sets. The drop in iterations between 13 to 14 shows that documents retrieved is irrelevant and later relevant documents were retrieved. The number of top n results showed to users will depicts the relevancy degree of the retrieved documents with respect to a given query with rank n (judged by the system users).

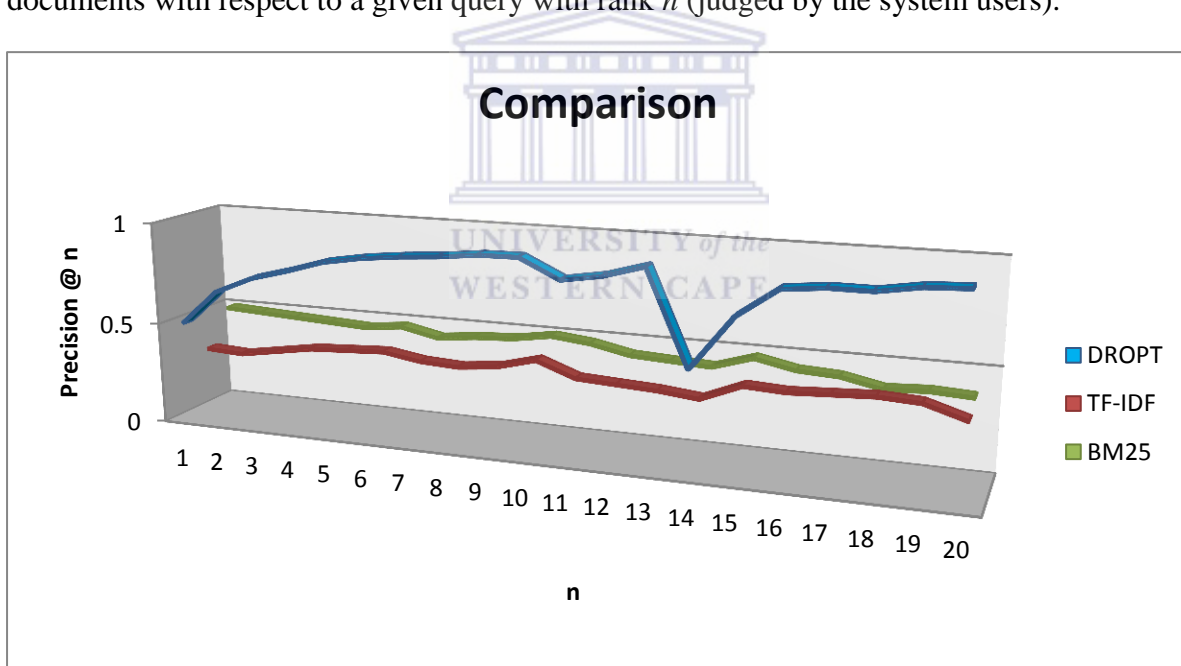


Figure 6-4: Comparison of DROPT with BM25 and TF-IDF in the P@n measure.

6.5 Personalizing search results

Personalization is the process of presenting the right information to a specific user at the right moment with the aim to improve search accuracy by matching user's interests. This research presents a novel DROPT measure for IR as an approach for applying subjective relevance judgments of documents returned by an IR system, as a mean to derive and adapt user

information needs models that can be used to improve IR effectiveness. The idea of context personalization proposed, responds to the fact that user preferences are multiple, changing, heterogeneous, and even contradictory and should be understood in context with the user goals in mind. As a result, user profile is represented using intelligent representation involving contextual attributes. In this approach, we can collect and analyze user information preferences and use it to construct a user's contextual profiles dynamically. Implicitly, the context associated with a contextual preference query is the current context, that is, the context surrounding the user at the time of the submission of the query. The current context should correspond to a single context state, where each of the values of the context parameter takes a specific value from domain of experts. Besides, information can be collected from the user explicitly for example, by asking for feedback such as preferences using relevance judgment. Documents are ranked based on their score, where higher scores are considered to be more relevant to the user after comparing the query of the document to the user's profile. Figure 6-5 illustrates this scenario.

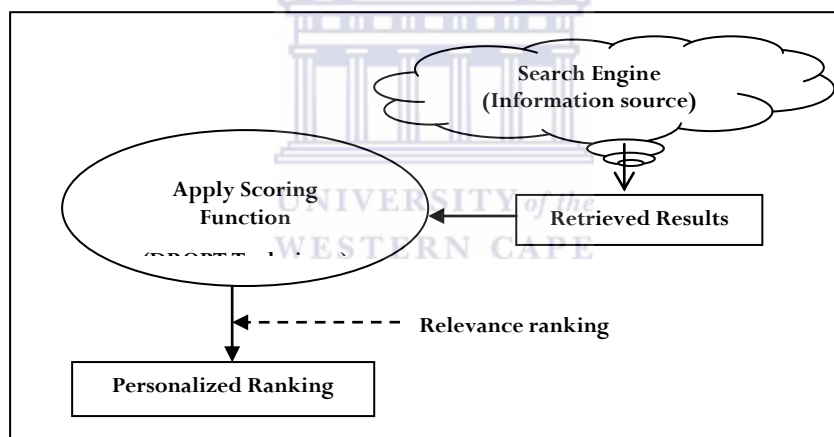


Figure 6-5: Personalized Search Results

The user then provides feedback on the relevance of the documents which the system uses to tune the user profile for adaptation. For these reasons, the subjective relevance is a cognitive user-centered task, which means two system users presenting the same query to an IR system may give different relevance judgment on the retrieved documents. It is helpful to note that a classic system user may have multiple and overlapping preferences. In learning the matching mechanism, when the environment of the adaptive search system changes so that only the highest ranked documents are of interest, then our ranking-driven DROPT approach is able to automatically adjust to the new environment.

6.6 Statistical analysis

Significance test interpretation was carried out in this research study with the purpose to measure the effectiveness of IR using interactive reinforcement learning (user's feedback and context-awareness) in comparison to relevance feedback. The test was established to reject the null hypothesis, H_0 that there is difference between the group means of Domain of system user participants 1, 2, 3, 4 and 5. Rejecting H_0 infers accepting the alternative hypothesis; H_1 with at least one of the means is different from others in retrieval efficacy in order to improve the system performance. The means and the standard deviations of the “*Online Interactive Reinforcement Learning Retrieval Prototype (IRLRP)*” keyword matching based querying experiments discussed in the previous Chapter are executed in the following.

Definitions:

Let M_{SB} depicts variance between the five domains considered in this research study.

Let M_{SW} depicts variance within the five domains considered in this research study.

In order to evaluate both the means and standard deviations of the keyword matching based querying experiments discussed in the previous Chapter, we construct hypothesis test based on the values obtained across all issued queries after 100 generations (20 search tasks from each participant domain) using Analysis of Variance (ANOVA).

$H_0: \mu = \mu 1 = \mu 2 = \mu 3, = \mu 4, \mu 5$ where 1, 2, 3, 4 and 5 are domains considered in this study.

H_1 : At least one of the means is different from the others.

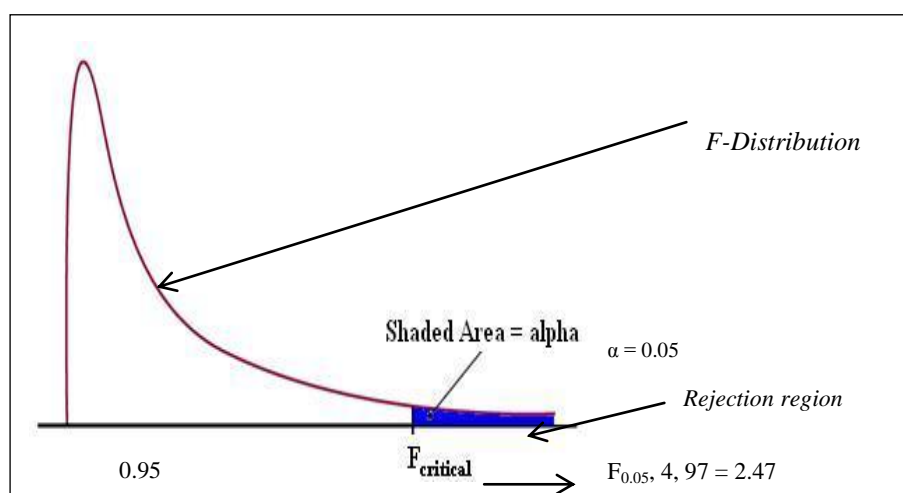


Figure 6-6: Showing values of 2.47 at $F_{0.05, 4, 97}$

It is noted that there are presently the value of $K = 5$ domains, that is, Domains 1, 2, 3, 4, and 5. Therefore, $DOF_N = K-1 = 5-1 = 4$. The sum total of data for all the five domains depicted as $N = n_1 + n_2 + n_3 + n_4 + n_5 = 20 + 20 + 20 + 20 + 20 = 100$.

Using the $DOF_D = N-K = 100-3 = 97$ and $\alpha = 0.05$ (the least significant value). The critical value if $F_{0.05, 4, 97} = 2.47$ (determined using F-Distribution table).

We need to find: $\bar{x} = \text{mean of mean} = \sum x/N$

$$M_{SB} = \sum ni(\bar{x} - \bar{x})^2 / K - 1 \text{ and } M_{SW} = \sum(ni - 1)Si^2 / N - K$$

Table 6-10: The values of occurrences of generated keywords from domains of participant 1, 2, 3, 4, 5

Parameters Determined	Ad-hoc Keywords/Query Terms					Occurrences of matched keywords				
	Domain 1	Domain 2	Domain 3	Domain 4	Domain 5	Domain 1	Domain 2	Domain 3	Domain 4	Domain 5
Query pattern	Swarm intelligent	Workflow scheduling	Learning user interaction	Information seeking		5	2	13	18	28
Relevance feedback	Data gathering	Grid environment	Search behaviour	Task effects		5	2	2	7	12
Information filtering	Traffic load	Efficient security	Query context	Topic familiarity		6	3	4	14	11
User profile	Medium access control	Authorization	Agglomerative clustering	Interactive searching		8	3	2	22	7
e-Health	Passive clustering	Grid portals	Interactive information retrieval	Inferring user preference		8	3	4	3	5
User preferences	Intelligent sensors	Homomorphic encryption	Evaluation model	Collaborative filtering		9	3	14	24	14
Intelligent agents	Wireless telemedicine	Cloud networking	Information seeking	Autonomous identification		10	4	2	9	19
Autonomous agents	Clustering algorithm	Medical grid	User goals	Autonomous text classification		13	4	2	3	22
Semantic	Ant colony optimization	Trust	User navigation behaviour	Faceted approach		18	5	2	22	8
Information retrieval	Health care	Interoperation	Local context	Search tasks		19	16	8	10	28
Data Mining	Software agent	Contextual evaluation	Personalized query expansion	Useful documents		10	15	5	6	20
Web search	Predictive models	Rule-based	Interactive search	Task types		17	5	16	26	26
User Interaction	Cognitive perspective	Personal ubiquitous computing	Usefulness	Multi-session tasks		7	7	22	12	25
Information behaviour	Multi-Agent system	User-based evaluation	Information retrieval	Mapping user queries		14	8	18	28	8
Information filtering	Clustering data mining	Hierarchic document clustering	Search strategies	Information seeking strategies		28	18	10	17	13
Language modelling	Inverted base	Document relevance	Implicit measures	User behaviour analysis		23	12	24	25	6
Semantic search engine	Automatic query expansion	Web search engines	Learning users interest	Implicit feedback		25	10	3	13	13
Query reformulation	Personal agent	Evaluation methods	Document relevance	Relevance feedback		12	19	22	24	14
User query expansion	Adaptive information retrieval	Document ranking	Contextual information	Modelling information		9	24	21	19	10

	Personalized access	Mobile devices	Adaptive web	Implicit feedback	Implicit indicators	22	14	8	11	21	
$\sum x$	→					268	177	202	303	310	
\bar{x}	→					13.4	8.85	10.1	15.15	15.5	
S^2	→					46.94	42.73	57.45	59.92	60.45	
n	→					20	20	20	20	20	N=100

The mean of mean denoted as $\bar{\bar{x}}$ was determined as follows:

$$\bar{\bar{x}} = \sum x/N = 268+177+202+303+310 = 1260/100 = 12.6$$

The mean for each of the domains are evaluated as follows:

$$\bar{x}_{\text{Domain 1}} = \sum x/N = 268/20 = 13.4$$

$$\bar{x}_{\text{Domain 2}} = \sum x/N = 177/20 = 8.85$$

$$\bar{x}_{\text{Domain 3}} = \sum x/N = 202/20 = 10.1$$

$$\bar{x}_{\text{Domain 4}} = \sum x/N = 303/20 = 15.15$$

$$\bar{x}_{\text{Domain 5}} = \sum x/N = 310/20 = 15.5$$

Also the variance for each of the domains is evaluated as follows:

$$S^2_{\text{Domain 1}} = \sum (x - \bar{x})^2 / N = 228.9/20 = 22.89$$

$$S^2_{\text{Domain 2}} = \sum (x - \bar{x})^2 / N = 154.5/20 = 15.45$$

$$S^2_{\text{Domain 3}} = \sum (x - \bar{x})^2 / N = 200.01/20 = 20.01$$

$$S^2_{\text{Domain 4}} = \sum (x - \bar{x})^2 / N = 487.56/20 = 24.48$$

$$S^2_{\text{Domain 5}} = \sum (x - \bar{x})^2 / N = 596.79/20 = 29.84$$

$$\text{Mean of mean } \bar{\bar{x}} = \sum x/N = (268+177+202+303+310)/100 = 12.6$$

Also from Table 6-10 shown, $M_{\text{SB}} = \sum ni(\bar{x} - \bar{\bar{x}})^2 / K - 1$ could be determined as follows:

$$M_{\text{SB}} = \sum ni(\bar{x}_{\text{Domain1}} - \bar{\bar{x}})^2 + \sum ni(\bar{x}_{\text{Domain2}} - \bar{\bar{x}})^2 + \sum ni(\bar{x}_{\text{Domain3}} - \bar{\bar{x}})^2 +$$

$$\sum ni(\bar{x}_{\text{Domain4}} - \bar{\bar{x}})^2 + \sum ni(\bar{x}_{\text{Domain5}} - \bar{\bar{x}})^2 / K - 1$$

$$M_{\text{SB}} = 20(13.4-12.6)^2 + 20(8.85-12.6)^2 + 20(10.1-12.6)^2 + 20(15.15-12.6)^2 + 20(15.5-12.6)^2 / 5-1 = 717.1/4 = 179.275$$

$$\text{Also, } M_{\text{SW}} = \sum (ni - 1)Si^2 / N - K$$

$$M_{SW} = (20-1) S^2_{\text{Domain 1}} + (20-1) S^2_{\text{Domain 2}} + (20-1) S^2_{\text{Domain 3}} + (20-1) S^2_{\text{Domain 4}} + (20-1) S^2_{\text{Domain 5}} / 100 - 3 = 19(46.94) + 19(42.73) + 19(57.45) + 19(59.9) + 19(60.2) / 97 = 5077.18 / 97 = 52.34$$

Therefore, the test statistics is $F = M_{SB} / M_{SW} = 179.275 / 52.34 = 3.42$

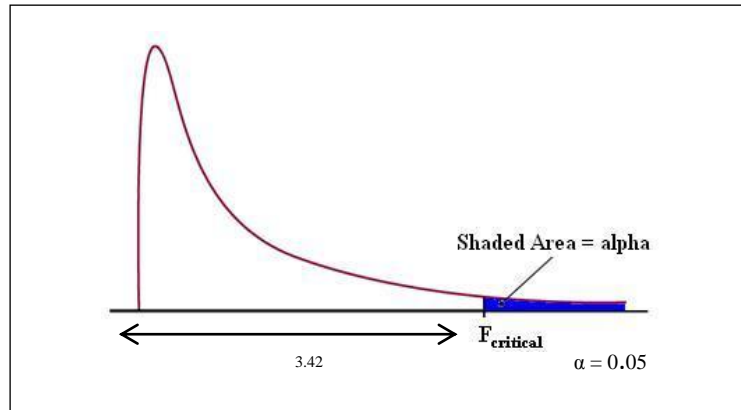


Figure 6-7: Showing F-Distribution table for 3.42

Conclusion:

Since F-statistical table falls to the left of F-distribution ($3.42 > 2.47$) under the acceptance region. Therefore we may conclude at a 5% level of significance test that there is a significant difference in the means of at least one group of Domains 1, 2, 3, 4 and 5. This is because the values of ad-hoc keywords matched against documents that were searched independently across each of the domains of system user's participants and the corresponding values of occurrences of issued query were obtained. The interpretation of this statistical result demonstrates the improvement of information retrieval efficacy through the attributes from the user behaviour actions while interacting with the IR system.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Context-awareness in IR is an exciting and challenging area of human-computer interaction. The basic idea is to give computers understanding in order to make them recognize the situations in which users interact with information systems and the services they provide. Using adaptive IR system, situations can be detected and classified as contexts. Once the proposed system has recognized in which context an interaction takes place, this information can be used to change and adapt the behaviour of IR applications and systems. The input side of the human-computer interaction (HCI) looks at information that individual users generate in order to interact with the real world and thus provides context-awareness in HCI.

Developing context-aware IR systems is very interesting and challenging. One has to keep in mind that users learn how to interact with the system, and that they adapt their behaviour. It is important that users understand the varying and adaptive behaviour of the IR application and connect it to the current situations they are in. So, it is also crucial to develop understandable context-aware IR system that adapts to the users' expectations. In line with this, well-designed context-awareness is a great and powerful way to make user-friendly and enjoyable IR applications.

Delivering the right information to the user is fundamental in IR system. Many traditional IR models assume term independence and view a document as information overload; however getting the right information requires a deep understanding of the content of the document and relationships that exist between terms in the documents, and extracting terms from the documents. In order to address this challenge, employing an efficient and effective text retrieval technique, which retrieves the most relevant documents and rank them at the top of the list, to improve system performance and retrieval effectiveness becomes critical. This can be achieved by applying context-aware clustering algorithms to extract terms from the

document. Context-aware clustering is suitable for applications in which the context is an important factor and the number of clusters is not known prior and such application is user profiling and, more specifically, the mining of user context can be effectively used for document clustering in the context of IR. Conversely, for a technique to be effective, it should offer a ranking mechanism involving user relevance judgment (feedback) about retrieved documents. Focusing on a document retrieval application, we proposed personalized context-aware IR model to retrieved documents to individual users. This in turn satisfying individual users' information needs. Ranking the retrieved document user model makes the documents appears in the order as the user interest is matched.

The overall goal of this research was to develop algorithms that optimize the ranking of documents. The goal of ranking functions is to match documents to user queries and place them in an order of their predicted relevance. The goal was to build a system capable of acquiring context information to individual users through the relevance documents during their search activities. Two research questions were developed to address the research goal, particularly how relevant information can be ranked with regards to context of information seeker. This was achieved by generating predictive document ranking models for IR.

The objectives of the research were accomplished by analysing results from a controlled user in-lab experiment. Participants were asked to search for twenty sessions that varied by document titles, and all of their interactions with the computer were logged on the client side. During the search, participants were asked to determine the occurrence of the keyword matching based querying that were relevant for helping them to accomplish the assigned search task and these generating behaviors were considered as explicit judgments of document relevance. In this study, we generated ranking models of document relevance on the basis of users' search interactive behaviors.

The research defines a user behaviour source (ranking of retrieved documents) that can influence the information retrieval process. Though considering user searching actions (i.e. clicking on a document in a search result, printing a document, moving a document into a folder, etc.) as sources for implicit relevance of documents, the techniques presented in this thesis is different because it considers document ranking. From that view, the techniques is interesting and innovative as it emphasizes that the IR process is not just about matching between documents and queries but relationships among matching, user actions and user preferences in ranked documents of retrieved results, could be indicators of document relevance.

User interactive behavior measures on relationships among matching help understand how users interact on the clicked documents in response to a given query, and they are indicative of document relevance. Also, user interactive behaviours measures during user actions help describe what the user does between issuing one query and the next. User interactive behaviours about user preferences help understand how to acquire search results. This in turn could improve the information retrieval effectiveness. The functional requirements analysis for development of personalized IR system was discussed based on the knowledge representation and the document ranking technique. The personalized search results means to explicitly make use of the user context to tailor search results.

Our results demonstrate a significant effect of document ranking on predictive ranking models according to document relevance. Document ranking not only affected the user interactive behaviour as predictors of document relevance, it also affected the relevance weights for each of the user interactive behaviours to improve IR effectiveness. In addition, when document information is available, the ranking model gives better prediction of document relevance. Therefore, we can conclude that it is important for personalized IR systems to detect the context in which a search is conducted, especially the document ranking, and then to apply the user model to personalize search results to individual users. Also document ranking influenced how users interacted with search systems during search. Previous studies have shown that document ranking could influence users' search interactive behaviors on the search level, e.g. the amount of effort to accomplish the ranking and the search techniques employed.

In order to satisfy the functional requirements of IR system, a context-aware IR system was proposed, which is able to personalize to individual user preferences, and explore new domains for potentially relevant document. This demonstrates how context awareness was employed to improve the predicted relevance of the retrieved documents according to information relevance through user's feedback that cannot be explored through a traditional IR process. This thesis has presented an approach for improving document retrieval efficacy by combining context-aware clustering and context-aware to make exciting IR applications. The thesis made contribution to the field of IR, by combining user preference relevance feedback, evolutionary algorithms, context awareness, and user information needs models which can be derived by contextual matching and feedback values that optimize ranking of retrieved documents. The idea of context personalization proposed, responds to the fact that

user preferences are multiple, changing, heterogeneous, and even contradictory and should be understood in context with the user goals and tasks in mind.

A document ranking algorithm was proposed to be integrated into the developed context-aware IR system that would provide a limited number of ranked documents in response to a given query. This improves the ranking mechanism for the search results in an attempt to adapt the retrieval environment of the users and amount of relevant context-aware information to each user's information needs, and self-learning that can automatically adjust its search structure to a user's query behaviour. The ranking technique presented in this thesis is different compared to other ranking mechanisms proposed in other TF-IDF approaches where there is no any position for the user; directly or indirectly produced promising results. Experimental results show that the precision value of the proposed ranking technique is comparatively higher for the query sets over the use of traditional relevance feedback alone. A DROPT technique has been evaluated to reflect how individual user judges the context changes in IR results ranking.

The system performance was evaluated to determine personalization to five diverse user profiles. For each domain of participant profile, it was illustrated that the relevancy of the top n results of the ranking list at known relevant documents for retrieval precision was achieved from the participants interaction with the information system. Besides, the retrieval precision for ranking performance results was superior to that achieved by the traditional relevance feedback. These results can be ascribed to the user-model ranking technique namely: user feedback, context-awareness, and reinforcement interactive learning.

In this design, user involvement is essential for providing the preference relevance feedback only at the ranking stage and user behaviours during interactions with search engine back end, and very essential to developing context-aware IR system that adapts to the individual users' expectations. In our research work we have used evolutionary algorithm (GA) in IR to find optimal set of documents that best matches the user's interest, and improve retrieval effectiveness. This is done by reformulating queries that adequately identified relevant documents and reject irrelevant documents based on individual user's feedback.

The thesis introduced a number of concepts in the context of IR ranking performance optimization. Predictive user model of document ranking were presented to personalize retrieved documents to individual users during their search context, rather than after they finish the entire ranking tasks. Also user-models were represented by documents in GA,

expressed in terms of indexed keywords and corresponding relevance weights for ranking tasks using categorical terms from participant domain. We designed a context-aware IR system, where we combine our entire solution-optimized designs into a single design to convey semantics information. So, context of the original keywords is determined which remove the drawbacks of the so-called keyword barrier of many retrieval models by selecting the most suitable semantics analysis according to the recognized context.

7.2 Future Work

Diverse issues are identified that could be explored as directions for future research in this thesis. There are three interesting directions for future research regarding document indexing in search system. The first is the issue of Web community that has moved to a situation where global multilinguality is becoming an ever more significant of the individual users' daily interaction with information on the Web [Ghorab 2010]. Yet, research in the area of personalized multilingual information retrieval (PMIR) is still in an early stage. Research in this area should enable users to achieve maximum benefit of information on the Web, beyond the barriers of language and country. The consideration may have a profound effect on the way personalized systems gather, model, and exploit individual user information for the delivery of a service that not only adapts to the user's knowledge and interests, but also to the user's cultural and linguistic background.

Results ranking and presentation is the second issue that have been explored in the literature, some of which were well studied in the context of PIR, while others may still require more attention and comparative evaluation regarding how they can be integrated with PIR. For example, a characteristic of the result diversification technique is that it aims at displaying diverse results in the first set of results presented to the user [Santos *et al.* 2010; Minack *et al.* 2009]. This notion can be considered as opposed to personalisation techniques, where the aim is to display many results from the topic that is inferred to be of relevance to individual user. To this end, there may be scope for investigating how these two complementary techniques can be brought together under one roof. There may be even more room for research on search results' presentation techniques that move away from traditional ranked list, where not only the "list" of results is adapted, but also the "content" of the results is re-structured and tailored to meet the user's knowledge and needs [Levacher *et al.* 2011].

The DROPT algorithm evaluated in this research has shown the approach to be promising in retrieval systems. The algorithm has some features like scalability and adaptability. It is

scalable in that we can add new algorithm easily and also adaptable in that it adapts itself with user information needs. There are many directions for future work relating to this approach in IR. Firstly, adding link-based ranking algorithms for comparison such as PageRank, HITS, and DistanceRank etc. Also ranking some fine grained features such as TF and IDF that the proposed algorithms are composed from, using the mentioned approach.



References

- AbdulRaham, R. A. and AbdulAziz, R. A. (2012): Data Mining and Visualization of Large Databases. *International Journal of Computer Science and Security (IJCSS)*, Vol. 6, Issue 5, pp. 295-314.
- Adomavicius G, Tuzhilin, A. (2011): Context-aware recommender systems. In: Ricci F, Rokach L, Shapira B, Kantor P (eds) *Recommender systems handbook*. Springer, Berlin, pp. 217–256
- Agichtein, E, Brill, E., Dumais, S. (2006a): Improving Web search ranking by incorporating user behavior information. In: *29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006)*, August 6-11, Seattle, Washington, USA, pp. 19–26.
- Agosto, D. E. (2012): Human Information Interaction: An Ecological Approach to Information Behaviour. *Journal of the American Society for Information Science and Technology*, Vol. 64, Issue 1, pp. 213-214.
- Allan, J. (2002): Challenges in information retrieval and language modelling. Report of a workshop held at the Centre for Intelligent Information Retrieval, University of Massachusetts, Amherst, September 2002.
- Anand, S.S., and Mobasher, B. (2007): Introduction to intelligent techniques for web personalization. *ACM Transactions on Internet Technology*, Vol.7, no 4, pp. 18.
- Asfari, Q., Doan, B-L., Bourda, Y., and Sanonnet, J-P. (2011): Personalized Access to Contextual Integration by using an Assistant for Query Reformulation. *International Journal on Advance in Intelligent Systems*, Vol. 4, No. 3 and 4, pp. 128-146.
- Asfari, Q., Doan, B-L., Bourda, Y., and Sanonnet, J-P. (2010): Context-based Hybrid Methods for User Query Expansion. *SEMANPRO 2010*. In *Proceedings of the fourth International Conference on Advanced in Semantic Processing*, pp. 69-74.
- Asfari, O., Doan, B. L., Bourda, Y. Sansonnet, J. P. (2009): Personalized access to information by query reformulation based on the state of the current task and user profile, In: *The Third International Conference on Advances in Semantic Processing*, Malta.
- Awad, M. M. (2012): A new geometric model for clustering high-resolution satellite images. *International Journal of Remote Sensing*, Vol. 33, Issue 18, pp. 5819-5838.
- Baeza-Yates, R, & Ribeiro-Neto, B. (1999): *Modern Information Retrieval*. ACM Press/Addison-Wesley.
- Baeza-Yates, R., and Ribeiro-Neto, B. (2011): *Modern Information Retrieval: The Concepts and Technology Behind Search*, 2nd edn. Addison-Wesley, Reading.

Baltrunas, L., Ludwig, B., Peer, S., and Ricci, F. (2012): Context Relevance Assessment and Exploitation in Mobile Recommender Systems. *Pervasive Ubiquitous Computing*, Vol. 16, pp. 507-526.

Bauer M, Heiber T, Kortuem, G, Segall Z. (1998): A collaborative wearable system with remote sensing. *Proceedings of the 2nd International Symposium on Wearable Computers (ISWC'98)*, 1998 Oct 19–20; Pittsburgh, PA. Washington: IEEE Computer Society; p:10–17.

Belhajjame, K., Paton, n. w., Fernandez, A. A., Hedeler, C., and Embury, S. M. (2011): User Feedback as a First Class Citizen in Information Integration Systems. In *Proceedings of 5th Biennial Conference on Innovative Data System Research*, pp. 175-183.

Bhatia, MPS., and Kumar, A. (2008a): The context-driven generation of web search. *Proceedings of Conference on Information Science Technology and Management (CISTM'08)*, pp. 281-287.

Belew, R. K. (2000): *Finding out about - A cognitive perspective on search engine technology and the WWW*. Cambridge University Press, Cambridge

Bellazzi, R and Zupan, B. (2008): Predictive data mining in clinical medicine: Current issues and guidelines. *International Journal of Medical Informatics*, Vol. 77, Issue2, pp. 81-97.

Berkhin, P. (2001): Survey of clustering data mining techniques. [Online]. Available: http://www.accrue.com/products/rp_cluster_review.pdf<http://citeseer.nj.nec.com/berkhin02survey.html>

Bordogna, G and Pasi, G. (2012): A quality driven Hierarchical Data Divisive Soft Clustering for Information Retrieval. *Journal of Elsevier, Knowledge-Based Systems*, Vol. 26, pp. 9-19.

Bordogna, G and Pasi, G. (2011): Soft Clustering for Information Retrieval applications. *Journal of Data Mining and Knowledge Discovery*, Vol. 1, Issue 2, pp. 138-146.

Borlund, P. (2003): "The Concept of Relevance in IR." *Journal of the American Society for Information Science and Technology* Vol. 54, No. 10, pp. 913-925.

Bouramoul, A., Kholadi, M. K., and Doan, B.L. (2011): "Using Context to Improve the Evaluation of Information Retrieval Systems", *International Journal of Database Management Systems (IJDMS)*, Vol.3, No.2, pp. 22-39.

Bouramoul, A., Kholadi, M. K., and Doan, B.L. (2010): "PRESY : A Context based query reformulation tool for information retrieval on the Web," In *JCS : Journal of Computer Science*, Vol 6, Issue 4, pp. 470-477, 2010., ISSN 1549-3636, New York, USA. April 2010.

Budanitsky, A. (1999): *Lexical semantic relatedness and its application in natural language processing*. Technical Report CSRG-390, University of Toronto, Canada.

Campos, P. G., Cantador, I, and Diez, F. (2013): Exploiting Time Context in Collaborative Filtering: An Item Splitting Approach. In *Proceedings of the 2nd Workshop on Context Awareness in Retrieval and Recommendation*, February, 2013, Rome, Italy, pp. 3-6.

Carpineto, C. and Romano, G. (2012): A Survey of Automatic Query Expansion in Information Retrieval. *ACM Computing Surveys*, Vol. 44, No. 1, Article 1, 50 pages.

- Carreras, A. M. and Botia, A. J. (2013): Building and Evaluating Context-Aware Collaborative Working Environment. *Journal of Information Science*, Vol. 235, pp. 235-241.
- Chaira, T. (2011): A novel intuitionistic fuzzy c-means clustering algorithm and its application to medical images. Published in: *Applied Soft Computing*, Elsevier, Vol. 11, Issue 2, pp. 1711-1717.
- Chellatamilan, T., and Suresh, R. M. (2013): Concept Based Query Expansion and Cluster Based Feature Selection for Information Retrieval. *Life Science Journal*, Vol. 10, Issue 7, pp. 661-667.
- Chen, T. Finin, and A. Joshi (2003): An Intelligent Broker for Context-Aware Systems. *Adjunct Proc. of UbiComp2003*, pp: 183–184.
- Chevalier, M., Julien, C., Soule-Dupuy, C. (2013): User Model for Adaptive Information Retrieval on the Web: Towards an Interoperable and Semantic Model. *International Journal of Adaptive, Resilient and Autonomic Systems*, Vol. 3, Issue 3, pp. 1-19.
- Cleverdon, C. (1997): The Cranfield Tests on Index Language Devices. In: Sparck-Jones, Karen; Willett, Peter (Eds.): *Readings in Information Retrieval*. Morgan Kaufman. pp. 47-59.
- Croft, W. B. (1997): Clustering large files of documents using the single-link method. *JASIS*, vol. 11, pp. 341-344.
- Danica, D., Milan, A., Hamish, C., Kalina, C. (2013): Improving Habitability of Natural Language Interfaces for Querying Ontologies with Feedback and Clarification Dialogues. *Journal of Web Semantics*, pp: 1-13.
- Dey A.K. (1998): Context-aware computing: The CyberDesk project. *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments; 1998 March 23–25; Stanford. Menlo Park, CA: AAAI Press, p:51–54.*
- Dey, A. K. (2001): Understanding and Using Context. *Personal Ubiquitous Computing*, vol. 5, no. 1, pp:4–7.
- Dinh, D and Tamine, L. (2012): Towards a Context Sensitive Approach to Searching Information based on Domain Specific Knowledge Source. *Web Semantics: Science Services and Agents on the WWW*, Vol. 12, pp. 41-52.
- Dumais, S., E. Cutrell, J. J. Cadiz, G. Jancke, R. Sarin and D. Robbins (2003). Stuff I've seen: a system for personal information retrieval and re-use. *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, Toronto, Canada, pp. 72-79.
- Dumitrescu, A and Santini, S. (2009): Think locally, search globally; context based information retrieval (2009). In *International Conference on Semantic Computing*, pages 396–401, Los Alamitos, CA, USA, IEEE Computer Society.
- Emmanouilidis, C, Koutsiamanis, R-A., Tasidou, A. (2013): Taxonomy of architecture, context-awareness, technologies and applications. *Journal of Network and Computer Applications*, Vol. 36, pp. 103-125.
- Ercan, G. and Cicekli, L. (2012): Keyphrase extraction through query performance prediction. *Journal of Information Science*, Vol. 38, No. 5, pp. 476-488.

- Everitt, B. Landau, S. and Leese, M. (2001): Cluster Analysis. London: Arnold.
- Fei, W, Changshui, Z, and Tao, L. (2007): Regularized clustering for documents. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '07). ACM, New York, NY, USA, pp: 95-102.
- Feldman, R and Sanger, J. (2007): The text mining handbook; advanced approaches for analysing unstructured data, Cambridge University Press.
- Fellbaum, C. et al. (1998): WordNet: An electronic lexical database. The MIT press.
- Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A (2007): User profiles for personalized information access. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web, 1 edn, pp. 54–89. Springer, Berlin
- Ghorab, M.R., Leveling, J., Zhou, D., Jones, G.J.F., Wade, V. (2010): Identifying common user behaviour in multilingual search logs. In: Peters, C., Di Nunzio, G., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A., Personalised Information Retrieval Roda, G. (eds.) Lecture Notes in Computer Science (6241/2010), Multilingual Information Access Evaluation I. Text Retrieval Experiments, pp. 518–528. Springer, New York
- Gladun, A., Roqushina, J., Valencia, G. R., and Bejar, R. M. (2013): Semantic-driven modelling of user preferences for information retrieval in the biomedical domain. Journal of Informatics for Health and Social Care. Vol. 38, No. 2, pp. 150-170
- Glushko, R. J. and Nomorosa, K. J. (2013): Substituting Information for Interaction: A Framework for Personalization in Service Encounters and Service Systems. Journal of Service Research, Vol. 16, No. 1, pp. 21-38.
- Goker, A., and Myrhaug, H. (2008): Evaluation of a mobile information system in context. Information Process Management, Vol. 44, no. 1, pp. 39-65.
- Goldberg, D. E. (1989): Genetic Algorithms in search optimization and machine learning. Addison-Wesley, Harlow, England. ISBN: 0-201-15767-5. pp: 412.
- Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G., Halatsis, C. (2007): Creating Ontology for the User Profile: Method and Applications. pp. 407–412. Research Challenges in Information Science (RCIS2007), Ouarzazate
- Guha, S, Rastogi, R, and Shim, K. (1998): CURE: An efficient clustering algorithm for large databases, in Proceedings. ACM SIGMOD International Conference Management of Data, pp: 73–84.
- Guha, S, Rastogi, R, and Shim, K. (2000): ROCK: A robust clustering algorithm for categorical attributes, Information System, vol. 25, no. 5, pp: 345–366.
- Guha, R. McCool, R, Miller, E. (2003) ‘Semantic search’, WWW ‘03: Proceedings of the Twelfth International Conference on World Wide Web, May, Budapest, Hungary, pp: 19-28.
- Gupta, P., Awasthi, Rastogi, R. (2013): Impact of Word Sense Ambiguity for English Language in the Web Information Retrieval. International Journal of Engineering Research and Technology, Vol. 2, Issue 1, pp. 1-5.
- Hall, L, Özyurt, I, and Bezdek, J. (1999): Clustering with a genetically optimized approach, IEEE Trans. Evolutionary Computation, vol. 3, no. 2, pp: 103–112.

- Hansen, P and Jaumard, B. (1997): "Cluster analysis and mathematical programming," *Mathematics Program*, vol. 79, pp: 191–215.
- He, Q. (1999): A review of clustering algorithms as applied to IR, Univ. Illinois at Urbana-Champaign, Tech. Rep. UIUCLIS-1999/6+IRG.
- Hearst, M. A. (2011): Natural Search User Interfaces. *Communication of the ACM*, Vol. 54, No. 11, pp: 60-67. DOI: 10.1145/2018396.2018414.
- Henricksen, K, Indulska, J, and Rakotonirainy, A. (2002): Modelling context information in pervasive computing systems. In *Proc. 1st Intl Conf. on Pervasive Computing*, LNCS 2414, pp: 167–180.
- Hirst, G. and St-Onge, D. (1998): Lexical chains as representations of context for the detection and correction of malapropisms. In Christiane Fellbaum (Ed.), *WordNet: An electronic lexical database*, Chapter 13, pp. 305-332, The MIT Press, Cambridge, MA.
- Ho, C., Lin, M. H., and Chen, H-M. (2012): Web Users' behaviour pattern of tourism information search: From online to offline. Published In *Elsevier*, Vol. 33, Issue 6, pp. 1468-1482.
- Islam, M., Liu, C., and Zhou, R. (2013): A framework for query refinement with user feedback. *The Journal of Systems and Software*, Vol. 86, Issue 6, pp. 1580-1595.
- Jain, A. K and Dubes, R. C. (1988): *Algorithms for Clustering Data*. Prentice-Hall, Inc.
- Jain, A, Murty, M, and Flynn, P. (1999): Data clustering: A review, *ACM Computer Survey*. vol. 31, no. 3, pp: 264–323.
- Jara, A. J., Lopez, P., Fernandez, D., Cashlo, J. F., Zamora, M. A., and Skarmeta, F. (2013): Mobile digcovery: discovering and interacting with the world through the Internet of things. *Journal of Personal and Ubiquitous Computing*, Published Springer-Verlag London.
- Jarvelin, K and Kekalainen, J (2000): IR evaluation methods for retrieving highly relevant documents. Published in: Belkin, N. J., Ingwersen, P. and Leong, M. K. (eds). In: *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY: ACM, pp. 41-48.
- Jones, S., Shao, L., Zhang, J, and Liu, Y. (2011): Relevance Feedback for Real World Human Action Retrieval. In: *Intelligent Multimedia Interactivity*, Vol. 33, Issue 4, pp. 446-452.
- Kebler, C., Raubal, M., and Wosniok, C. (2009): Semantic Rules for Context-aware Geographical Information Retrieval. In P. Barnaghi, editor, *4th European Conference on Smart Sensing and Context*, EuroSSC 2009, University of Surrey. Vol. 5741 of LNCS Springer, pp. 77-92
- Kirn, S, Anhalt, C, Krcmar, H, Schweiger, A. (2006): Multiagent Engineering: Theory and Applications in Enterprises, *International Handbook on Information Systems*, Springer, Berlin–Heidelberg, 2006 (Ch. Agent. Hospital—Health Care Applications of Intelligent Agents, pp. 199–220).
- Kolatch, E. (2001): Clustering algorithms for spatial databases: A Survey. [Online]. [Accessed on 2011/05/17] Available: <http://citeseer.nj.nec.com/436443.html>.

- Koorangi, M, Zamanifar, K. (2007): a Distributed Agent Based Web Search using a Genetic Algorithm. *International Journal of Computer Science and Network Security*, vol. 7, no. 1, pp. 65-76.
- Levacher, K., Lawless, S., Wade, V. (2011): A proposal for the evaluation of adaptive content retrieval, modification and delivery. In: *Workshop on Personalised Multilingual Hypertext Retrieval (PMHR 2011)*, pp. 18–25. ACM, Eindhoven
- Lew, M. S., Sebe, N., Djeraba, C., & Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, Vol. 2, No. 1, pp: 1-19.
- Li, W., Ganguly, D., Jones, and G.J.F. (2011): Enhanced Information Retrieval Using Domain-Specific Recommender Models. In: Amati, G., Crestani, F. (eds.) *ICTIR 2011*. LNCS, vol. 6931, pp. 201–212.
- Li, Y. and Belkin, N. J. (2008): A faceted approach to conceptualizing tasks in information seeking. *Information Processing and Management*, Vol. 44, No. 6, pp. 1822-1837.
- Liu, F., Yu, C., Meng, W. (2004): Personalized Web search for improving retrieval effectiveness. *IEEE Trans. Knowl. Data Eng.* **16**, 28–40
- Manning, C.D., Raghavan, P., Schütze, H. (2008): *Introduction to Information Retrieval*. Cambridge University Press, Cambridge
- Lukowicz, P., Pentland, A. S., and Ferscha, A. (2011): From Context Awareness to Socially Aware Computing, *IEEE Pervasive Computing*, Vol. 11, No. 1, pp. 32-41.
- Luo, H. I. Wei, H, and Ren, Y. (2009): Constructing Ensembles by different Clustering Algorithms for Object Categorization. *Intelligent Information Technology Applications, IITA 2009*. Third International Symposium on 21-22 Nov. 2009, Nanchang. Vol. 2, pp. 461–464.
- Maciaszek, L. A. (2007): *Requirements analysis and system design*. Pearson Education, Essex, UK, 3rd edition.
- Marco, D. S., Vidhya, N., Churchill, E. (2013): In *Proceedings of the SIGHI Conference on Human Factors in Computing Systems*, pp. 2487-2496.
- Manning, C.D., Raghavan, P., Schütze, H. (2008): *Introduction to Information Retrieval*. Cambridge University Press, Cambridge
- Maulik, U and Bandyopadhyay, S. (2000): Genetic algorithm-based clustering technique, *Pattern Recognition*, vol. 33, pp: 1455–1465.
- McCarty, J.A. and Hastak, M. (2007): Segmentation Approaches in data mining: A Comparison of RFD, CHAID and Logistic Regression. *Journal of Business Research*, Elsevier, Vol. 60, pp. 656-662.
- Melucci, M. (2005): Context modelling and discovery using vector space bases. *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, Bremen, Germany, pp. 808-815.
- Micarelli, A., Gasparetti, F., Sciarrone, F., Gauch, S. (2007): Personalized search on the WorldWideWeb. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) *The Adaptive Web*, 1 edn, pp. 195–230. Springer, Berlin

- Minack, E., Demartini, G., and Nejd, W. (2009): Current approaches to search result diversification. In: First International Workshop on Living Web: Making Web Diversity a True Asset, Washington DC
- Mollineda, R and Vidal, E. (2000): A relative approach to hierarchical clustering, in Pattern Recognition and Applications, Frontiers in Artificial Intelligence and Applications, M. Torres and A. Sanfeliu, Eds. Amsterdam, The Netherlands: IOS Press, vol. 56, pp: 19–28.
- Morris, J. and Hirst, G. (1991): Lexical cohesion computed by thesaurus relations as an indicator of the structure of text. *Computational Linguistics*, Vol. 17, no. 1, pp. 21–43.
- Mylonas, P., Vallet, D., Castells, P., Fernandez, M., Avrithis, Y. (2008): Personalized information retrieval based on context and ontological knowledge. *Knowledge Engineering Review*, Vol. 23, No. 1, pp. 73-100.
- Nafiz, A. and Fatos-T, Y. (2001): An Overview of Character Recognition Focused on off-line Handwriting. *IEEE Transactions on Systems, Man, and Cybernetics – Applications and Reviews*, Vol. 3, No. 2, pp. 216-233.
- Noh, H. Y., Lee, J. H., Oh, K. S., and Cho, S. B. (2012): Exploiting indoor location and mobile information for context-awareness service. *Journal of Information Processing and Management*, Vol. 14, Issue 1, pp. 1-12.
- Nyongesa, H. O., Maleki-dizaji, S. (2006): User modelling using evolutionary interactive reinforcement learning. *Inf Retrieval*, vol. 9, no. 3, pp. 343-355. DOI: 10.1007/s10791-006-4536-3
- Padmapriya, A and Subitha, N. (2013): Clustering Algorithms for Spatial Data Mining: An Overview. *International Journal of Computer Applications*, Vol. 68, No. 10, pp. 28-33.
- Palomino, M. A., Vincenti, A., and Owen, R. (2013): Optimising web-based information retrieval methods for horizon scanning, *Foresight*, Vol. 15, Issue 3, pp. 159-176.
- Rauber, A, Paralic, J, and E. Pampalk, E. (2000): Empirical evaluation of clustering algorithms, *Journal of Information Organization Science*, vol. 24, no. 2, pp: 195–209.
- Salton, G and Buckley, C. (1988): Term-weighting approaches in automatic text retrieval. *Information Processing and Management* 24, pp: 513–523.
- Saparova, D., Kibaru, F., and Basic, J. (2013): Use of widgets as Information Management tools in online shared spaces. *International Journal of Information Management*, Vol. 33, Issue 2, pp. 401-407.
- Santos, R.L.T., Macdonald, C, Ounis, I. (2010): Exploiting query reformulations for Web search result diversification. In: 19th International Conference on World Wide Web (WWW 2010), pp. 881–890. ACM, Raleigh
- Saracevic, T. (2007): "Relevance: A review of the literature and a framework for thinking on the notion in information science. Part III: Behavior and effects of relevance." *Journal of the American Society for Information Science and Technology* Vol. 58, No.13, pp. 2126-2144.
- Setchi, R, Tang, Q, and Stankov, I. (2011): Semantic-based information retrieval in support of concept design. In: *Proceedings of Advanced Engineering Informatics*, pp.131-146.
- Sease, R. (2008): Metaphor's Role in the Information behaviour of Humans Interacting with Computers. *Journal of Information Technology and Libraries*, Vol. 27, No. 4, pp. 9-16.

- Sebastian, F. (2002): Machine learning in automated text categorization. *ACM Computing Surveys*, Vol. 34, no. 1, pp: 1–47.
- Shen, X., Tan, B., Zhai, C. (2005): Implicit user modelling for personalized search. In: 14th ACM International Conference on Information and Knowledge Management (CIKM 2005), pp. 824–831. ACM, Bremen
- Shivaswamy, P. K. and Joachims, T. (2011): Online Learning with Preference Feedback. In NIPS workshop on Choice Models and Preference Learning.
- Song, S and Li, C. (2006): Improved ROCK for Text Clustering Using Asymmetric Proximity, SOFSEM 2006, LNCS 3831, pp: 501 – 510.
- Song, R., Luo, Z., Nie, J. Y., Yu, Y., and Hon, H. W. (2009): Identification of ambiguous queries in web search. *Information Processing and Management*, Vol. 45, No. 2, pp. 216–229
- Speretta, M., and Gauch, S. (2005): Personalized search based on user search histories. In: IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005), pp. 622–628. Compiegne University of Technology, Compiegne
- Spink, A and Cole, C. (2005): *New Directions in Cognitive Information Retrieval*. Springer-Verlag, Germany.
- Stamou, S., and Ntoulas, A. (2009): Search personalization through query and page topical analysis. *User Model. User-Adapt. Interact.* Vol. 19, pp. 5–33
- Steichen, B., Lawless, S., O’connor, A., Wade, V. (2009): Dynamic hypertext generation for reusing open corpus content. In: 20th ACM Conference on Hypertext and Hypermedia (Hypertext 2009), pp. 119–128. ACM, Torino
- Steichen, B., Ashman, H, and Wade, V. (2012): A Comparative Survey of Personalized Information Retrieval and Adaptive Hypermedia Techniques. *Journal of Information Processing and Management*, Vol. 48, Issue 4, pp. 698-724
- Steinbach, M, Karypis, G, and Kumar, V. (2000): A comparison of document clustering techniques. In: *Proceedings of KDD workshop on Text Mining*. Technical Report #00 – 034.
- Strang T, and Linnhoff-Popien C. (2004): A Context Modeling Survey. In: *First International Workshop on Advanced Context Modeling, Reasoning and Management*, held as part of UBICOMP 2004, Nottingham, UK.
- Sugiyama, K., Hatano, K., Yoshikawa, M. (2004): Adaptive Web search based on user profile constructed without any effort from users. In: 13th International Conference on World Wide Web (WWW 2004), pp.675–684. ACM, New York
- Tamine, L., Boughanem, M., and Daoud, M. (2010): “Evaluation of contextual information retrieval effectiveness: overview of issues and research,” In *Journal of Knowledge and Information Systems*. Volume 24 Issue 1, pp. 1-34. Springer, London, United Kingdom.
- Teevan, J., Dumais, S.T., Horvitz, E. (2005): Personalizing search via automated analysis of interests and activities. In: 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2005), pp. 449–456. ACM, Salvador
- Tseng, L. and Yang, S. (2001): A genetic approach to the automatic clustering problem, *Pattern Recognition*, vol. 34, pp: 415–424.

- Vieira, V., Tedesco, P., Salgado, A.C., and Brézillon, P. (2007): Investigating the specifics of contextual elements management: the cematika approach. *Context*, pp. 493-506.
- Wang, W and Fan, S. (2010): Application of Data Mining Technique in Customer Segmentation of Shipping Enterprises. Published in: Database Technology and Application (DBTA), 2nd International Workshop on 27 -28 Nov. 2010, Wuhan, pp. 1-4.
- Wei, C. Lee, Y, and Hsu, C. (2000): Empirical comparison of fast clustering algorithms for large data sets, in Proceeding. 33rd Hawaii International Conference System Sciences, Maui, HI, 2000, pp: 1–10.
- White, R. W., Ruthven, I., and Jose, J. M. (2005): A study of factors affecting the utility of implicit relevance feedback. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Salvador, Brazil. 35-42.
- White, R. and D. Kelly (2006): A study on the effects of personalization and task information on implicit feedback performance. *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM, Arlington, Virginia, USA, PP. 297-306.
- Witten, I, H, Moffat, A, Bell, T. C (1999): *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann Publishers, San Francisco, CA.
- Xue, W., and Deng, H. (2012): Unstructured queries based on mobile user context. *International Journal of Pervasive Computing and Communications*, Vol. 8, Issue 4, pp. 368-394.
- Yang, Y. (2009): Image segmentation based on fuzzy clustering with neighbourhood information. *Journal of Optical Application*, Vol. 39, No. 1, pp. 135-147.
- Yang, Y. and Huang, S. (2007): Image segmentation by fuzzy c-clustering algorithm with a novel penalty term. *Journal of Computing and Informatics*, Vol. 26, pp. 17-31.
- Yao, Y. Y. (1995): Measuring retrieval effectiveness based on user preference of documents, *Journal of the America Society for Information Science*, vol. 46, no. 2, pp. 133-145.
- Yousri, N. S, Kamel, M. S, and Ismail, M. A. (2008): Finding Arbitrary Shaped Clusters for Character Recognition. Published in: *Proceedings ICIAR '08 Proceedings of the 5th International Conference on Image Analysis and Recognition*, pp. 597-608.
- Yu, J. and Jeon, M. (2010): A Context-Aware Intelligent Recommender System in Ubiquitous Environment. In: *Proceedings of the 10th IASTED International Conference Artificial Intelligence and Application (AIA 2010) February 15-17 2010, Innsbruck, Austria*, pp. 229-234.
- Zhang, T, Ramakrishnan, R, and Livny, M. (1996): BIRCH: An efficient data clustering method for very large databases, in *Proceedings of ACM SIGMOD Conference Management of Data*, pp: 103–114.
- Zhao, X., Anma, F., Ninomiya, T., Okamoto, T. (2008): Personalized Adaptive Content System for Context-Aware Mobile Learning. *International Journal of Computer Science and Network Security*, Vol. 8, No. 8, pp. 153-161.
- Zhou, D., Lawless, S., Wade, V. (2012): Improving search via personalized query expansion using social media. *Inf. Retr.* pp. 1–25. Doi: 10.1007/s10791-012-9191-2