A Framework for the Delivery and Evaluation of Personalised Multilingual Information Retrieval

A thesis submitted to the

University of Dublin, Trinity College

for the degree of

Doctor of Philosophy

Mohammed Rami ElHussein Ghorab

Knowledge and Data Engineering Group (KDEG)

School of Computer Science and Statistics

Trinity College, Dublin,

ghorabm@tcd.ie

2014

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other University, and that, unless otherwise stated, it is entirely my own work.

_____

Mohammed Rami ElHussein Ghorab

Date

Permission to lend or copy

I agree to deposit this thesis in the University's open access institutional repository or allow the Library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

_____

Mohammed Rami ElHussein Ghorab

Date

Dedication

*To my daughter **Laila**,*

who is as old as this PhD!

Acknowledgements

First and foremost, all thanks and praise are due to God, who has granted me this great success in my PhD, and has bestowed upon me all the necessary strength, health, wits, patience, and perseverance to complete it.

I would like to express my utmost gratitude to my supervisor **Prof. Vincent Wade** for all his guidance, support, patience, feedback, encouragement, and golden nuggets of advice that he gave me all throughout the years of the PhD. I had several friends who did PhDs during the same period I was doing mine and we always talked about our supervisors; every time we talked I came to realise more and more how lucky I am to be supervised by Vinny. No words are enough to show how grateful I am to have had him as my supervisor.

I would also like to express my sincerest gratitude to **Dr. Alexander O'Connor** whom I simply believe this thesis would not have been accomplished without. I would also like to express my gratitude to **Dr. Séamus Lawless** for all the invaluable discussions and feedback throughout the PhD. Again, no words are enough to express how fortunate I am that Alex and Shay assisted in supervising my PhD and reviewing my publications. I don't think I could have done it without their constant feedback, insightful comments, extremely helpful suggestions, and above all, their sincere encouragement. On the personal level, they have both been very friendly and very supportive and I have always appreciated this, and always will. I can never forget the very first day for me at the University when Shay was the first person for me to meet, where he greeted me at the door of the O'Reilly building and gave me a warm welcome into KDEG.

I would like to thank all my colleagues in KDEG and SCSS who have directly and indirectly supported me during the years of the PhD, especially: Kevin Koidl, Killian Levacher, Dominic Jones, Luca Longo, Ben Steichen, and Hilary McDonalds. I would also like to thank Dr. Tim Brailsford (University of Nottingham) for helping me in starting this PhD at Trinity College.

Very special thanks to all my Egyptian friends in Dublin who have been a major part of this journey. I can't imagine how life would have been here if it wasn't for their friendship and support. I don't think I would have been able to make it through. I don't want to fall into the pitfall of forgetting names, but I have to mention the following very special people: Ahmed Shosha, Waleed Abo-Hamad, Walid Magdy, Ahmed Selim, Amr Arisha, Mohammed Ragab, Sammy Bedair, Amr Mahfouz, Haymen Shams, Wael Rashwan, Ahmed Hosny, Muhammad Ebraheem, Saeed Samy, and Mostafa Bayomi.
If there is one friend worthy of a section of his own in this acknowledgement it would be **Mohamed Samir**. Words of gratitude and thanks can never do him his rights. He is, has always been, and forever will be my best friend, my brother, and my soul mate. He has been with me, in one way or another, throughout all the stages of my life ever since we met in the undergrad years, providing support, sincere advice, help, and encouragement. He has always been there when I needed him. So, THANK YOU SAMIR.

I would like to thank my best friends: Mohammed Hany, Ahmed Magdy, Abd ElRahman ElShazly, Ahmed Fouad, and Amr Reda. I would also like to thank my aunts: Soad and Azza, my uncle: Mohsen, and my dear cousins: Montasir, Zyad, Motaz, Hesham, and Hazem. Very very special thanks to Mohamed Sabri and Essam Eliwa who have been an inspiration and a very important part of both my PhD and MSc journeys.

I would like to thank my former boss at the Information Technology Institute (ITI-Egypt) Dr. Mohamed Salem for his encouragement and support throughout my career. I would also like to thank Bahaa Abd El-Aleem, Amr ElShafey, and Heba Saleh. I would also like to thank my teachers, peers, managers, and colleagues at ITI for the fruitful years during which I studied and worked at ITI. Special thanks to all members of the Java Department (JETS) at ITI, and even specially to Ahmed Mazen and Yassmin Sameh who supported me through the PhD.

I would like to thank my housemates in "Synnott House" in Dublin for all the good and happy times we've spent together which certainly helped me get through the hard times of the PhD, especially: Roshan Koonjul, Kishan Koonjul, Giuseppina Amato, Julien Sterck, and Sandeea Kodai. Very special thanks to Cezary Filipek for being such good

friend and housemate, for his support, for the many fruitful and stimulating conversations we had together, and above all for making this wonderful idea of a multi-cultural house come true in Synnott House.

And I've saved the best for last: My utmost gratefulness and sincerest thanks to my mother, my father, my wife, my brother, and my sister for all their love, devotion, care, encouragement, help, and moral support throughout the PhD journey. It's hard to put it in words, but I can say that I was only able to achieve this and make it that far because they believed in me. I hope I have made you proud.

Abstract

The amount of content provided in different languages on the Web is growing every day. The best answer to a user's query may not necessarily be available in his/her own language, but may reside in the diverse, multilingual corpora of the Web. Furthermore, as Internet penetration increases around the world, the number of multilingual users who seek and interact with information on the Web is also increasing. Personalised Information Retrieval (PIR) aims to help users in satisfying their information needs in a more accurate and less time-consuming manner. The user's search can be personalised by keeping track of his/her personal information and interests, and using this information for query adaptation and result-list adaptation. However, current search personalisation approaches do not pay adequate attention to the effect of multilinguality (of both the users and the content). This has a significant impact on the way PIR services should be delivered and evaluated. The study reported in this thesis argues that users' searches are influenced by language. For example, a multilingual user, whose native language is not English, may prefer to use his/her native language when seeking certain types of content on the Web (e.g. news), yet choose to use English when seeking other types of content (e.g. technical content). Furthermore, in multilingual search, the user may choose to click on documents originating from certain languages depending on the type of information sought. The study reported in this thesis shows that taking multilinguality into consideration significantly affects PIR. The study therefore introduces the notion of *Personalised Multilingual Information Retrieval (PMIR)* and proposes a novel framework for the delivery and evaluation of PMIR services. This entailed designing, implementing, and evaluating a set of algorithms for multilingual user modelling, multilingual query adaptation, and multilingual result-list adaptation. Furthermore, this entailed designing and implementing a framework that enables evaluating the compartmentalisation and the combination of PMIR elements. The evaluation shows the success of the multilingual approach to search personalisation and highlights the benefits of the PMIR framework. The methodology undertaken for this study involved: theoretical investigation, an industry case study, user studies, and empirical evaluation. The PMIR framework and the personalisation approaches proposed in this study contribute to the areas of Personalisation and Information Retrieval as they advance research concerning how to model Web users, how to retrieve information that adequately satisfies their information needs, and how to make this information accessible to them.

Table of Contents

List of Tables

List of Figures

Glossary

IR	Information Retrieval
MIR	Multilingual Information Retrieval *(also used interchangeably: multilingual search)*
PIR	Personalised Information Retrieval *(also used interchangeably: personalised search)*
PMIR	Personalised Multilingual Information Retrieval
AH	Adaptive Hypermedia
IF	Information Filtering
QA	Query Adaptation *(a general term for: query expansion/modification/augmentation)*
SeQA	Selective Query Adaptation
PRF	Pseudo-Relevance Feedback
RA	Result Adaptation *(a general term for: re-ranking/merging result lists)*
UM	User Model *(a user profile)*
FragUM	Fragmented User Model
EcombUM	Early-Combined User Model
LcombUM	Late-Combined User Model
RRmerge	Round Robin Merging
SCmerge	Score-based Merging
SimQ	Query Similarity
SimD	Document Similarity
SimT	Total Similarity
MT	Machine Translation
TEL	The European Library
GUI	Graphical User Interface
HCI	Human-Computer Interaction
TFPN	True/False - Positive/Negative
MVC	Mode-View-Controller

Chapter 1: Introduction

This chapter presents the motivation for this Personalised Multilingual Information Retrieval (PMIR) study and states the research question raised in this thesis. Furthermore, the chapter presents the objectives of the study and the research approach taken to meet these objectives. It also identifies the contributions and the deliverables of this thesis. Finally, the chapter provides an overview of the structure of the thesis.

1.1 Motivation

Today's Web is becoming increasingly multilingual[1]. Nearly half of the content available on the Web is provided in languages other than English, such as Russian (6%), Spanish (5%), Chinese (4%), Japanese (4%), and Arabic (3%). The best answer to a user's query may not necessarily be available in his/her own language, but may reside in the diverse, multilingual corpora of the Web. While English remains the dominant language in terms of the amount of content on the Web, several other languages are witnessing a significant increase in Web content, such as Chinese, Spanish, German, French, and Russian[2]. Moreover, with the rising development in countries such as Brazil, Russia, India, and China (a.k.a. the BRIC countries) (Wainer et al., 2009, Jain, 2007), more information is being published on the Web that is not originally authored in English. Therefore, as the Web community is growing to a situation where multilinguality is becoming an important aspect of users' daily interactions with information, solutions are needed to assist users in overcoming the barrier between the languages that the users can comprehend and the languages in which relevant information is available.

Web content can be generally divided into two categories: Professional Content (i.e. content that is authored by service providers) and User-Generated Content (i.e. content that is authored by service users). Professional content (e.g. enterprise content) is produced in a variety of languages by many service providers and enterprises with the aim of reaching a wide user or customer base. Moreover, with the advent of Web 2.0 technologies, users are encouraged to publish their own content and interact with other users' content as opposed to the passive viewing of content that is solely published by service providers. As a result of this, the Web is witnessing a huge increase in User-Generated Content (UGC) (Obrist et al., 2008) which is naturally authored in a plethora of languages.

[1] http://www.internetworldstats.com/
[2] http://en.wikipedia.org/wiki/Languages_used_on_the_Internet

Localisation has become a core part of the business process of many enterprises. Localisation aims to adapt information, in terms of content and presentation, to culture, locale, and linguistic environment (van Genabith, 2009). *Personalisation* can be thought of as a continuation along the same trajectory as Localisation; Localisation focuses on adaptation for a region or identified population, whereas personalisation takes it to the extreme of an individual person (O'Connor et al., 2009). Personalisation makes use of a range of user and usage information to tailor services and content according to the needs of the user with the aim of facilitating the process of finding, accessing, and comprehending information (Steichen et al., 2011, Vallet et al., 2010, Teevan et al., 2009, Gauch et al., 2007, Brusilovsky and Millán, 2007).

Personalised systems have been demonstrated in several areas in the literature, such as Web search (Vallet et al., 2010, Stamou and Ntoulas, 2009, Teevan et al., 2009, Agichtein et al., 2006a), eLearning (Brusilovsky and Millán, 2007, Conlan et al., 2003, De Bra et al., 2003), and news dissemination (Katakis et al., 2009, Billsus and Pazzani, 2007). A key component in personalised systems is the user model (Kobsa, 2007a, Gauch et al., 2007). User models are used to represent a variety of information about the user, such as: the user's prior knowledge, interests (likes and dislikes), personal preferences, and demographic data (e.g. location, language, age, etc.). Moreover, user models may also store information about the user's preferred modality, content delivery mechanism, and the context surrounding the use of the system. The information in the user models is then employed to adapt both the type of content presented to the user and the way in which content is presented to the user, with the aim of increasing user satisfaction when interacting with the system.

Multilinguality (with respect to both content and users) and **Personalisation** are the two key foundational aspects of the study reported in this thesis. Specifically, this study falls within two research areas: (1) *Multilingual Information Retrieval*, with a focus on multilingual search on the Web; and (2) *Personalised Information Retrieval*, including personalised search and personalised access to online content (e.g. customer support content, digital library archives, or open Web content).

The current situation with information consumption is that more and more multilingual users are interacting with content on the Web. Therefore content and service providers are under more pressure to find ways to accurately satisfy the information needs of those multilingual users. With respect to search systems, one of the ways of accurately satisfying a user's query is by employing personalisation. With multilinguality becoming an important dimension of the information seeking/consumption process, personalised search, in turn, has to be extended into

the multilingual dimension. Therefore, this study is concerned with: *personalised multilingual search* and *multilingual personalised search*. The former notion specifically entails studying approaches to personalising a multilingual search service and the latter notion generally entails studying how multilinguality affects search personalisation, including the studying of how to model multilingual users and cater for their multilingual search interests.

The remaining part of this section presents a brief account of the research conducted in the area of Multilingual Information Retrieval and in the area of Personalised Information Retrieval, and then highlights the challenges associated with the fusion of the two areas and how this thesis addresses them.

Multilingual Information Retrieval (MIR) and Cross-Language Information Retrieval (CLIR) are subfields of Information Retrieval (IR) that are concerned with retrieving documents from document collections that are not limited to the query's language (Peters et al., 2012, Oard, 2010, Nie, 2010). The terms CLIR and MIR are used interchangeably in some parts of the literature. However, this thesis distinguishes between them using the following distinction from the literature: in CLIR, documents are retrieved from one target language to satisfy a query in a source language, and the result list is often presented without translation. On the other hand, MIR involves the retrieval of documents from one or more languages, including the source language. Furthermore, MIR may involve translating the results to the language of the source query and merging the retrieved result lists into one list (Tsai et al., 2008, Si and Callan, 2005, Chen and Gey, 2004). Translation plays a crucial role in MIR and CLIR, where either the source query is translated into the target language (at runtime), or the documents are translated into designated languages *a priori* (offline) (McCarley, 1999, Oard, 1998). The query translation approach has gained wider recognition in the literature (Hefny et al., 2011, Chen and Gey, 2004, Gao et al., 2007), and therefore it is the approach used in this study.

Many studies have investigated improvements in retrieval effectiveness in MIR and CLIR by developing techniques that enhance query disambiguation and query translation. For example, in (Gao et al., 2007), the authors proposed an algorithm for cross-lingual query suggestion based on multilingual search logs (query logs). The authors in (Ambati and Uppuluri, 2006) developed multilingual search systems using bilingual dictionaries and information from monolingual search logs. In (Cao et al., 2007) the authors suggest a Markov Model that combines query translation and expansion in one process.

Although adaptation in the abovementioned studies is performed on multilingual search, it is not performed at the level of the individual user. In other words, these studies adapt

multilingual search on a macro level (based on collective information from search logs), rather than personalise the search on an individualised level. The research reported in this thesis aims to develop adaptation algorithms that cater for individual user needs in multilingual search and aims to construct user models that represent attributes and interests of multilingual users.

Personalised Information Retrieval (PIR) has gained significant attention in the literature (Steichen et al., 2011, Teevan et al., 2010, Mulwa et al., 2011, Micarelli et al., 2007, Agichtein et al., 2006a). Providing a personalised service to Web search users helps them in satisfying their information needs (Vallet et al., 2010, Speretta and Gauch, 2005, Teevan et al., 2005). Textual Information Retrieval systems have become wide-spread across the Web community, being used in search engines (Agosti and Melucci, 2001, Kobayashi and Takeda, 2000), digital libraries (Agosti, 2011, Chowdhury and Chowdhury, 1999), or local search facilities provided on numerous websites (i.e. site-specific search). A typical search process would involve users submitting queries, often in the form of a set of terms, to a retrieval system and receiving a ranked list of results in return. A natural characteristic of traditional IR systems is that if different users submit the same query, the system would yield the same list of results, regardless of the user. PIR systems, on the other hand, include the user in the equation (Micarelli et al., 2007, Brusilovsky and Tasso, 2004, Silvestri, 2010). In other words, a PIR system does not retrieve documents[1] that are relevant to the query alone, but ones that are also relevant to the user; thus, different users may actually receive different results for the same query. This can be done by keeping track of the user's personal information and interests and then using this information to adapt the query (e.g. query expansion) or adapt the results (e.g. result re-ranking).

A key component of PIR systems is the *user model* (Hu and Chan, 2008, Gauch et al., 2007, Sugiyama et al., 2004, Brusilovsky, 2001). A user model keeps track of the user's information such as demographic data and search interests. PIR systems employ various mechanisms to gather user and usage information. For a PIR system to obtain user information, it could either request that users explicitly supply this information or it could implicitly gather this information in an unobtrusive manner from the users' search history. Gathering information from search history entails analysing objects that are exhibited in search logs including queries, clickthrough data and, documents. An important aspect of PIR systems is how each system stores and represents the gathered information. Some systems store this information in an individualised manner (Zhang et al., 2007, Speretta and Gauch, 2005, Pretschner and Gauch,

[1] The terms *document* and *result* are used interchangeably in this thesis to denote any object in the result list retrieved in response to a query

1999, Psarras and Jose, 2006), while other systems maintain an aggregate view of usage information across the cohort of system users (Agichtein et al., 2006b, Smyth and Balfe, 2006). The user models in the aforementioned studies represented the users' search interests in a monolingual fashion. It is not an uncommon case in today's world to have users who are familiar with multiple languages. For example, many internet users from various countries are familiar with English in addition to their native language. Moreover, some countries, such as Switzerland, South Africa, Canada, and USA are naturally multilingual (by law or by population preference). The research reported in this thesis argues that taking the aspect of multilinguality into consideration significantly affects the way user information is gathered, modelled, and employed for the delivery of a personalised service on the Web.

The research reported in this thesis shows that users' searches are influenced by language. For example, a multilingual user, whose native language is not English, may prefer to use his/her native language when seeking certain types of content (e.g. news), yet choose to use English when seeking other types of content (e.g. technical content). Furthermore, in multilingual search, the user (whether a monolingual or a multilingual user) may choose to click on documents originating from certain languages depending on the type of information sought. This behaviour suggests that a user has multiple behavioural personas[1] when seeking information on the Web, dependent on the combination of their language capabilities, and the availability and variety of content in various languages. Therefore, a key element in this research is realising that users may browse documents from multiple languages (whether in the original form or a translated form) and that they may be capable of submitting search queries in multiple languages as well. Thus, the personalisation approach in this research is focused on the fact that both users and content can be multilingual.

In summary, this study argues that there is a need to investigate Personalised Multilingual Information Retrieval in order to define the elements and the workflow of the PMIR process and to evaluate approaches to multilingual search personalisation. In particular, there is a need to examine the effect of introducing multilinguality to the search personalisation process and to investigate how to evaluate that effect.

[1] The term *persona* refers to the mode of behaviour of a user when interacting with a service or system. Thus, *multiple user personas* refers to the multiple representations which identify different facets of the same user.

1.2 Research Question

The main research question posed in this study is: ***What are the key considerations for evaluating the effect of a multilingual approach to search personalisation?***

This research question encompasses the following challenges:
Challenge #1: *What are the key components of the search personalisation process and how can the process accommodate a personalised multilingual search service?*
Challenge #2: *Are there certain behavioural patterns or differences that can be observed for users in multilingual search?*
Challenge #3: *Can the use of user models that encompass the aspect of multilinguality improve retrieval effectiveness in PMIR?*
Challenge #4: *How should query adaptation and result adaptation algorithms be extended in order to incorporate the aspect of multilinguality?*
Challenge #5: *What is the users' perception of using a system that offers personalised search across multiple languages?*

In light of the identified research challenges, this study aims to improve personalised search through a better understanding of how users seek and interact-with content within the context of multilinguality. Therefore, the scope of the study includes: the investigation of how users behave when searching for information, the investigation of user modelling and personalisation approaches, and the investigation of users' perception of multilingual search services. However, investigating how to improve the quality of the content or the quality of translation is outside of the scope of this thesis. Furthermore, studying IR models is also outside of the scope of the thesis.

1.3 Research Objectives and Approach

The study reported in this thesis aims to fulfil the following objectives:

1. To gain insight into users' search behaviour in light of multilinguality.
2. To investigate user modelling approaches that account for the aspect of multilinguality.
3. To establish a framework for evaluating the compartmentalisation and the combination of PMIR elements.
4. To improve retrieval effectiveness in MIR by means of personalised query adaptation and result adaptation algorithms.

The methodology adopted to address these objectives involved the following:

- Theoretical investigation: review, analysis, and classification of existing approaches in the literature.
- A case study: demonstrating the need for and the usefulness of MIR in a realistic scenario.
- User studies: analysis of patterns and differences in users' search behaviour.
- Empirical evaluation: quantitative and qualitative evaluation of PMIR. This is achieved by the use of proven evaluation methodologies, from the fields of Information Retrieval (IR) and Adaptive Hypermedia (AH), applied in a technical framework.

The following points represent a step-by-step approach to address the objectives:

1. **Investigate the state of the art in PIR and PMIR:** this theoretical investigation involves the identification of points of strength and weakness in existing personalisation approaches. It serves as a basis for understanding the key components of the search personalisation process**.**
2. **Evaluate the usefulness of MIR in realistic scenarios:** this involves evaluating the effectiveness of multilingual search in a realistic customer support scenario (industry case study), which is an important step prior to the investigation of personalised multilingual search.
3. **Analyse users' search behaviour:** this entails carrying out an analysis that contributes towards understanding how the presence of multilinguality, whether on the user's side or the content's side, affects the process of seeking and gaining access to information. Thus, the analysis provides guidelines regarding how user models should be represented in order to be well-suited for PMIR.
4. **Evaluate the effect of adapting search results based on the attribute of language:** this comprises an initial evaluation to explore the potential of performing search adaptation based on the search language (i.e. query language) and the language of the content.
5. **Establish a framework that defines the components and workflow of the PMIR process:** this entails defining of the elements of PMIR and the inter-communication between these elements in order to develop a comprehensive framework for the delivery and evaluation of PMIR services.
6. **Investigate various user model representations:** this comprises the investigation of alternative approaches to representing the user's attributes and search interests. This involves investigating different techniques and structures for constructing multilingual user models and maintaining the user's multilingual search interests.

7. **Develop search personalisation algorithms that cater for multilinguality:** this involves developing algorithms for query adaptation and result adaptation in PMIR.
8. **Evaluate the multilingual approach to search personalisation:** this involves quantitative evaluation of the effectiveness of the proposed adaptation algorithms which operate in conjunction with the proposed user models. This also involves qualitative evaluation of the usability and perceived usefulness of PMIR with respect to the users.

1.4 Contribution and Deliverables

This study introduces and evaluates a novel approach to multilingual search personalisation and shows that factoring multilinguality into the search personalisation process introduces significant changes to the components and workflow of the process. It also affects the way user models, query adaptation algorithms, and result adaptation algorithms are designed, implemented, and evaluated.

The findings of this study contribute to the areas of Personalisation, User Modelling, Information Retrieval, and Digital Libraries. Specifically, the deliverables of the study are as follows:

- **PMIR Framework:** the main deliverable is a clearly defined process and workflow for PMIR, and a system for the delivery and evaluation of PMIR services.
- **PMIR Approaches:** the second deliverable takes the form of a set of approaches for multilingual search personalisation. These approaches comprise multilingual user models and adaptation algorithms that operate in conjunction with these models.

Parts of the research reported in this thesis have been published in the following international conferences, workshops, and journals of repute:

- User Modeling and User-Adapted Interaction Journal (UMAI).
- Cross-Language Evaluation Forum (LogCLEF)[1].
- Conference on Theory and Practice of Digital Libraries (TPDL).
- Conference on User Modeling, Adaptation and Personalization (UMAP).
- Conference on Research and Development in Information Retrieval (SIGIR) [Doctoral Consortium].
- Workshop on Personalised Multilingual Hypertext Retrieval (PMHR).
- Workshop on Semantic Media Adaptation and Personalisation (SMAP).

[1] CLEF is now known as: Conference and Labs of the Evaluation Forum.

1.5 Thesis Overview

This chapter presented the motivation for this research and stated the research question and challenges addressed in this study. The chapter also presented a set of objectives that this study aims to meet in order to address the challenges. Finally, the contribution and deliverables of the study were presented.

A state-of-the-art survey of the literature is presented in Chapter Two. The survey features a critical review and classification of the stages and components of Personalised Information Retrieval (PIR); thus, it addresses the first challenge stated in Section 1.2. The review covers a variety of personalised experimental systems and commercial systems in the fields of monolingual IR, multilingual IR, and a number of closely related fields. The survey presents novel classifications of PIR systems. These classifications provide new insights on how information gathering and modelling approaches affect the design of PIR services with respect to the scope of personalisation addressed. The survey also provides a discussion of general issues related to the research areas of IR, MIR, Personalisation, User Modelling, and AH. The chapter ends by summarising the findings of the survey, drawing conclusions about the current state of the art in PIR, and highlighting the open research directions that are to be addressed in this thesis.

Chapter Three presents the design of the research conducted in this study. The chapter builds on the lessons learnt from the state-of-the-art survey and outlines the design of the theoretical approach proposed to meet the objectives of the thesis. The chapter discusses the elements of the solution which involves designing user models that reflect the aspect of multilinguality and search personalisation algorithms that cater for multilinguality. The chapter then proposes a framework for the delivery and evaluation of PMIR which harnesses all the elements of the solution. This involves a discussion of the PMIR process and workflow, and a discussion of design considerations for the framework.

In Chapter Four, an implementation of the PMIR framework and the personalisation algorithms proposed in this thesis is discussed. The discussion of the framework involves explaining implementation details concerning its various components and highlighting how the design requirements stated in Chapter Three guided the implementation process. The discussion of the algorithms involves the pseudo-code and the details of user-model construction, query adaptation, and result adaptation algorithms.

Chapter Five presents the evaluation carried out for this study. The chapter discusses the setup and the outcomes of the experiments. The chapter first presents the results of the industry case study which shows the usefulness of MIR in realistic customer support scenarios. It then discusses the findings of the investigation of users' search behaviour which involved analysing a dataset of multilingual search logs. Following from the preliminary investigation, the chapter presents the outcome of an exploratory experiment that demonstrates the efficacy of incorporating the attribute of language (user's language and content's language) in the process of personalising search results. The chapter then discusses a set of experiments for quantitative evaluation of the effectiveness of a number of proposed PMIR algorithms applied in conjunction with various user model representations. The experiments compare the improvements achieved by the various personalisation approaches with respect to users coming from different linguistic backgrounds. Finally, the chapter reports qualitative evaluation of the usability of the PMIR system and the users' perception of the multilingual search results presented to them (in terms of relevance and quality of translation).

This thesis is concluded in Chapter Six which provides a summary of the key contributions of this study, a discussion of how the objectives are met and how the research question is answered, and a discussion of future work that may be carried forward from the thesis.

Each chapter, and the sections thereof, address one or more of the research challenges stated in Section 1.2. Figure 1 shows a layout of the challenges and where they are addressed in the thesis.

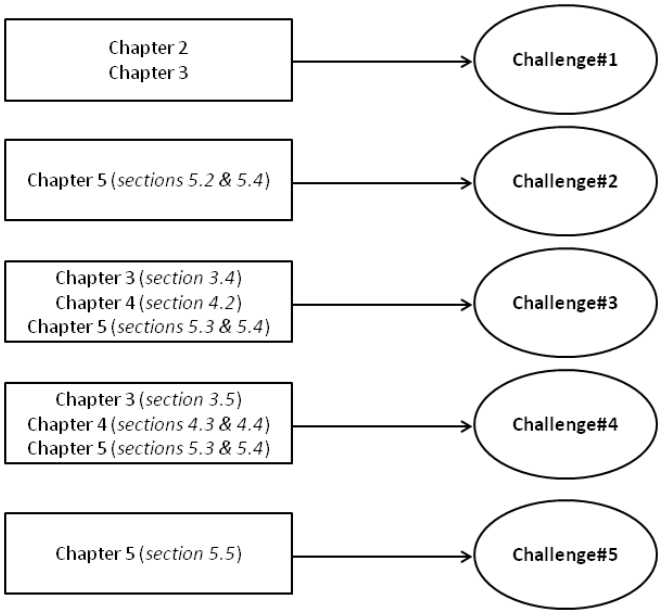

Figure 1: addressing the research challenges in the thesis

Chapter 2: Background and State-of-the-Art

This chapter provides a state-of-the-art survey and classification of PIR literature. The review covers a variety of personalised experimental systems and commercial systems in the fields of monolingual IR, multilingual IR, and a number of closely related fields. The chapter also provides a discussion of general issues related to the research areas of IR, MIR, Personalisation, User Modelling, and AH. Finally, the chapter presents a summary of the findings of the survey and highlights the open research directions which this thesis aims to address.

2.1 Introduction

With the enormous increase in the amount of information on the Web, there is a growing need for systems that offer personalised services to Web users, where information is adapted to the user's needs in terms of content and presentation (Brusilovsky, 2001, Jameson, 2008). Modelling user and usage information, whether on an individual user scope or community scope, is an essential process in personalised systems. Significant research is being carried out concerning how to gather, represent, and make use of such information for providing personalised services on the Web (Gauch et al., 2007, Micarelli et al., 2007, Brusilovsky and Tasso, 2004, Brajnik et al., 1987).

PIR assists users in satisfying their information needs (Micarelli et al., 2007, Steichen et al., 2011, Agichtein et al., 2006a, Vallet et al., 2010, Teevan et al., 2010). For example, assume a certain user is interested in critical reviews of works of literature (e.g. novels or plays) and submits the query "A Tale of Two Cities" to a search engine. The retrieval system will then attempt to retrieve all documents that are relevant to the query terms from the document collection. This will return diverse documents as results for this search, such as: text excerpts from the body of the novel, information about the film that was created based on the novel, websites that offer to sell the novel or the film, critical reviews of the novel, information about the author Charles Dickens, and perhaps a number of irrelevant documents or documents that are related to another article or object that shares the same name. Therefore, users who are specifically interested in critical reviews or analysis of the literature will have to respond by either one of two actions: either they will have to sift through the many results that are not relevant to their information needs in order to find the ones that are relevant to them, or they will have to reformulate the query in order to specify their intent (e.g. submit a new query: "A Tale of Two Cities Analytic Review"). Now if the system "knows" that a particular user is interested in reviews of works of art, then it can adapt the result list with respect to this inferred

interest. Results that represent analytic reviews about the novel would therefore be moved to the top of the ranked list where they would be more easily accessible to the user. This process is referred to as result re-ranking, or more generally as *result adaptation*. Furthermore, the system can adapt the original query itself, for example by automatically adding some terms to it, such as "critique", "criticism", "analysis", "analytic", or "review", so that more specific results could be retrieved in the first place. This process is known in the field of IR by many terms: query expansion, query augmentation, query modification, or more generally *query adaptation*.

PIR systems generally undertake three stages in order to provide their personalised service (Gauch et al., 2007). The first stage is *information gathering*, where a set of mechanisms are put in place to collect information about the users. The second stage is *information representation*, where user modelling approaches and data structures are used to represent the information that was gathered about the user. The third stage is the *implementation and execution of personalisation*, where re-composition, adaptation, and recommendation algorithms are employed to adapt the queries or the results to the users, based on their models.

This survey features a review and critical analysis of the three PIR stages. Furthermore, the survey classifies existing PIR systems according to the various approaches exhibited in each stage. The survey also provides a review of the different methods used to evaluate PIR systems.

The objectives of carrying out this critical review of the literature are:

1. To investigate points of strength and weakness in current approaches to PIR so as to guide the research activity of this study.
2. To highlight open challenges in the field of PIR and why it makes sense to carry out research that addresses those challenges.
3. To gain an insight into the whole PIR process and classify the approaches exhibited in PIR stages in order to formalise a framework for implementing, delivering, and evaluating PMIR services in an adequate manner.

A number of papers in the literature have carried out state-of-the-art surveys of some aspects of PIR systems (Micarelli et al., 2007, Gauch et al., 2007, Kelly and Teevan, 2003). The survey presented in this chapter extends existing surveys as follows:

- A novel classification of PIR systems is presented in this chapter. Systems are categorised with respect to the scope on which personalisation is performed, namely as: *individualised personalisation*, *community-based personalisation*, or *aggregate-level personalisation systems*. Individualised personalisation is when the system's adaptive decisions are taken according to the information about each individual user as exhibited

in his/her user model (Stamou and Ntoulas, 2009, Speretta and Gauch, 2005). Community-based personalisation takes a step further from individualised personalisation where the system's adaptivity is performed in a collaborative manner (Teevan et al., 2009, Sugiyama et al., 2004). This involves systems in which a model is also constructed on a per-user basis, but where sharing of information between models can take place. Aggregate-level personalisation refers to the notion of a system that does not explicitly make use of a user model to represent each individual user; in which case personalisation is guided by aggregate usage data as exhibited, for example, in search logs (Agichtein et al., 2006b, Sun et al., 2005). This may be considered as a special case (a wider scope) of the community-based type, but the difference is that no user model exists *per se*. An in-depth discussion of this classification is provided in Section 2.4.2.2.

- This survey comprises a review of both monolingual and multilingual PIR systems, as opposed to other surveys that only covered monolingual systems.
- A novel classification of user models, according to their underlying data structure and the nature of their content, is presented in Section 2.3.2.2.
- An extensive discussion of query adaptation and result adaptation techniques is provided in Section 2.4. Furthermore, a novel classification of query adaptation techniques is presented in Section 2.4.2.3 where the techniques are divided into *user-focused* vs. *non-user-focused* (i.e. personalised techniques that involve user information in the process and non-personalised techniques that only involve information from the queries and the document collection) and *implicit* vs. *explicit* (i.e. techniques that do not require user intervention and ones that require a specific user action).
- This survey features a dedicated section for reviewing evaluation approaches in PIR systems (Section 2.5), where the most important quantitative and qualitative evaluation techniques from the fields of Information Retrieval and Adaptive Hypermedia are discussed.

The rest of this chapter is organised as follows. Section 2.2 discusses the information gathering stage of PIR systems where various techniques and sources are used to acquire the necessary information on which personalisation is based. Section 2.3 discusses the information representation stage where different data structures are used to maintain user and usage information in PIR systems. Section 2.4 discusses the personalisation implementation and execution stage where a variety of techniques are used for search personalisation. Section 2.5 discusses the different evaluation approaches and metrics used to evaluate PIR systems in terms

of effectiveness and usability. Finally, Section 2.6 presents the summary and conclusion of the state-of-the-art survey.

2.2 Information Gathering

2.2.1 Overview

This section of the survey is concerned with the first stage of personalisation, which is information gathering. A discussion is provided regarding the different sources and types of information on which personalisation can be based, and also regarding the different approaches to obtaining this information. The importance of discussing the information gathering stage stems from the idea that the nature of information available for a personalised system determines the way that the system can implement personalisation at later stages. The analysis is carried out over three criteria: the information gathering approach, the type of information, and the source of information. An overview of these three criteria is as follows:

- **Information gathering approach:** the first criterion is the approach to gathering the information. Information can be gathered in an implicit manner where it is obtained without any extra effort from the user or in an explicit manner where the users have to explicitly supply information to the system.
- **Type of information:** the second criterion is the type of information gathered about the users and their usage behaviour when interacting with the system. User information is information collected about users themselves, such as their personal information, demographic information, or search interests. Usage information, on the other hand, is information recorded about the users' interactions with systems on the Web; for example, in the scope of Web search, this includes submitted queries, browsed pages, annotated content, bookmarked pages, and tags. User information is traced back to a certain user, whereas usage information may be aggregated across many users.
- **Source of information:** the third criterion considered in this part of the analysis is the information source. Usage information can be gathered at the server-side or at the client-side. In addition, this criterion is also concerned with where the information is maintained, and highlights related privacy concerns.

The following section provides a detailed review and analysis of different approaches exhibited in PIR systems. The analysis focuses on the information gathering stage of the surveyed systems, guided by the three criteria outlined above.

2.2.2 Review

2.2.2.1 Information Gathering Approach

Information can be gathered in an implicit or an explicit manner (Gauch et al., 2007). In the implicit method, information is gathered unobtrusively, without any additional effort from the user. This is typically the case when a system keeps track of the user's search history in terms of submitted queries and clicked results (Stamou and Ntoulas, 2009, Gao et al., 2007, Smyth and Balfe, 2006, Speretta and Gauch, 2005). This also includes processing any stored user documents or items (e.g. emails, calendar items, etc.) (Chirita et al., 2007, Agichtein et al., 2006a, Teevan et al., 2005), or harvesting information from the user's interactions with social applications (e.g. social networks, social tagging applications, blogs, etc.) (Vallet et al., 2010, Carman et al., 2008). The implicit method attempts to automatically infer user's interests or context of use from the processed logs or user items.

In the explicit method, the users themselves have to supply information to the system, whether positive or negative. This can take the form of a user providing the system with an initial specification of interests or "non-interests" (Micarelli and Sciarrone, 2004), providing positive or negative relevance feedback about retrieved documents (Chen and Sycara, 1998, Asnicar and Tasso, 1997, Harman, 1992b), or scrutinising (inspecting and modifying) the information that the system has learnt about the user so far (Psarras and Jose, 2006, Micarelli and Sciarrone, 2004, Pitkow et al., 2002). Concerns regarding the explicit method are that users may not wish to exert the extra time or effort to supply the information to the system and that users may sometimes input inconsistent or incorrect information (Budzik and Hammond, 2000, Carroll and Rosson, 1987). Some systems, such as the Outride system presented in (Pitkow et al., 2002), gather user and usage information in a mixed approach of implicit and explicit methods.

A good example of systems that depend on the explicit approach for gathering various information is the WIFS system (Micarelli and Sciarrone, 2004), which is a PIR system that operates on the domain of computer science literature. In WIFS, the system initially learns the user's interests through an interview form that is used when the user first registers with the system. The form allows the user to explicitly specify his/her degree of interest in different computer science topics on a scale from -10 to +10 (i.e. very irrelevant to very relevant). Moreover, the user can provide explicit relevance feedback about viewed documents on the same scale. Upon the user's feedback, the terms in the rated document are processed by the system which affects the user model by the alteration of interest weights, the removal of interests, or the insertion of new interests in the user model. Finally, the user model can be

scrutinised where the user is allowed to inspect and modify the inferred interests and their weights.

It is necessary, at this point in the chapter, to clarify the notion of what is deemed an implicit method and what is deemed an explicit method. To a certain extent an implicit method partially entails an action on the user's behalf, such as clicking on a result's link (in the case of learning from clickthrough history). However, this type of gathering user information is deemed as implicit because it does not require that users perform any *additional* activities other than the ones they would normally carry out during a search session. Likewise, an explicit method partially entails some form of automatic processing either before or after the user's action. For example, if the user is asked to provide explicit relevance feedback to the system by marking one of the results as relevant, then the next step would be that the system automatically processes the document to extract its keywords and append them to the user's interests. Nevertheless, this would still be deemed as an explicit method because it involved some extra activity by the user that is specifically carried out for obtaining information that would assist in the personalisation process.

2.2.2.2 Type of Information

The type of information gathered by a system, whether user or usage information, influences how a system can personalise its service. User information is usually in the form of personal or demographic information such as the user's name, age, language, or country. User information may also include the user's job title, job description, or competency. Usage information exists in many forms, including queries that the user submitted to the search system, clicked results and their snippets (titles and summaries of documents), full browsing activity[1], and dwell time[2] on clicked documents. User and usage information also include information that can be obtained from external resources (i.e. from resources other than the search system itself), such as the user's emails, calendar items, and stored desktop documents on the user's machine.

A number of systems in the literature only keep track of clickthrough behaviour, which comprises submitted queries and clicked documents (Smyth and Balfe, 2006, Stamou and Ntoulas, 2009, Cui et al., 2003, Qiu and Cho, 2006). Other systems extend this information by also logging the text from the snippets (titles and summaries) of clicked results (Yin et al., 2009, Psarras and Jose, 2006, Ruvini, 2003, Shen et al., 2005). Snippets are regarded by several

[1] Browsing activity comprises URLs clicked from the result list and any pages followed afterwards, along with other browsing-related information

[2] Dwell time is the estimated time that the user has spent viewing a document

studies as query-focused summaries of documents and are therefore used to extract interest terms that are relevant to the context of the query. For instance, the MiSearch system (Speretta and Gauch, 2005) maintains snippet information with the aim of comparing the effectiveness of a user model where terms are obtained only from submitted queries to one where terms are obtained from snippets of clicked documents.

The majority of PIR systems maintain monolingual search logs, and relatively few operate on a multilingual level. An example of multilingual PIR systems is the cross-language search system described in (Gao et al., 2007) where the authors extend the logged information by keeping track of queries submitted in different languages. They are motivated by the idea that in the same period of time, many users from different language backgrounds will share similar information needs. Thus, similar queries in different languages will exist in the logs, which can be used for personalised cross-language search. This kind of personalisation, however, is performed on the aggregate-level (macro level) of the cohort of user queries and not in an individualised manner (i.e. not per user).

A number of systems in the literature gather a richer set of information about usage behaviour. For example, in the OBIWAN system (Pretschner and Gauch, 1999) cached Web pages on the user's machine and their estimated dwell time are analysed in order to determine the user's interests. Another example is (Teevan et al., 2005) where the authors gather information about the user's queries, visited Web pages, emails, calendar items, and stored desktop documents. These studies are motivated by the notion that a richer set of user information contributes towards a more complete view of the user's context, which will improve personalisation.

2.2.2.3 Source of Information

The amount of information available for PIR systems varies depending on the sources or repositories from which information is obtained. Moreover, where the gathered or processed information is maintained also has an effect on the personalisation process in terms of when and where the information is available to be employed for personalisation. Furthermore, privacy concerns are raised concerning where the information will be stored and how it will be used (Kobsa, 2007b).

Usage information can be obtained from the server-side where the user's interactions with the system are logged. Several research studies (Yin et al., 2009, Qiu and Cho, 2006, Psarras and Jose, 2006, Speretta and Gauch, 2005, Liu et al., 2004) and numerous live systems on the Web,

such as *Google*[1], *Bing*[2], *Yahoo*[3], *Facebook*[4], and *del.icio.us*[5], maintain and process the history of users' interactions with the system at the server-side. One drawback of this approach, however, is that it may sometimes raise privacy concerns for the users. Nevertheless, this approach is used by the majority of commercial systems on the Web and these systems have managed to attract a large user base.

A number of studies in the literature, while maintaining information at the server-side, took into consideration the privacy aspect. For example, in the I-SPY system (Smyth and Balfe, 2006), the authors argue that no user identification or personal details should be logged among the data at the server in order to preserve the anonymity of the user. This is believed to provide a certain comforting degree of privacy to the users of the system. The authors call this kind of personalisation anonymous personalisation. However, the problem with this anonymous approach is that it limits the possibilities of individualised personalisation, as it has to be performed at the aggregate level of behaviour of the search users.

Usage information can also be gathered at the client-side. The advantage of gathering information at the client-side, compared to server-side logging, is that it allows for a richer set of information to be collected about user interactions and behaviour. For example, the exploration of information at the client-side gives opportunity for analysing the full browsing activity of the user which extends to pages that the user viewed after abandoning the search interface. This is done by accessing the browser's cache or by using software tools that are installed on the client's machine (e.g. browser plug-ins). Examples of such systems are (Stamou and Ntoulas, 2009, Chirita et al., 2007, Teevan et al., 2005, Shen et al., 2005, Pretschner and Gauch, 1999).

Another advantage of systems that maintain information at the client-side is that they offer a certain degree of privacy to their users by guaranteeing that user information will not be submitted to a remote server. However, some client-side systems lack this advantage as they submit the collected information to the server for further processing. Examples of such systems are presented in (Agichtein et al., 2006a, Sugiyama et al., 2004, Stefani and Strapparava, 1999).

[1] http://www.google.com
[2] http://www.bing.com
[3] http://www.yahoo.com
[4] http://www.facebook.com
[5] http://www.delicious.com

2.2.3 Summary and Discussion of Information Gathering

The previous section reviewed existing approaches to information gathering in PIR systems. Table 1 offers a summarised view of these approaches along with publication examples.

Table 1: summary of information gathering approaches

Information Gathering Approach	Type of Information	Source of Information	Example Publications
Implicit	Queries, clicked documents, or snippets of clicked documents	Server-side	Yin et al. 2009, Smyth and Balfe 2006, Qiu and Cho 2006, Speretta and Gauch 2005, Cui et al. 2003, Ruvini 2003
Implicit	Queries in different languages and clicked documents	Server-side	Gao et al. 2007
Implicit	Queries, clicked documents, or snippets of clicked documents	Client-side	Stamou and Ntoulas 2009, Shen et al. 2005
Implicit	Queries, clicked and cached web pages, dwell time on pages, desktop documents, emails, or calendar items	Client-side	Chirita et al. 2007, Teevan et al. 2005, Pretschner and Gauch 1999
Implicit	Queries, clicked and cached web pages, dwell time on pages, or other usage features	Client-side (information submitted to server)	Agichtein et al. 2006, Sugiyama et al. 2004, Stefani and Strapparava 1999
Implicit	Tags and Bookmarks on online social applications	Server-side	Vallet et al. 2010, Carmel et al. 2009, Xu et al. 2008, Carman et al. 2008
Implicit & Explicit	Queries, clicked documents, and user supplied information (e.g. user can scrutinise model or specify categories)	Server-side and user intervention	Psarras and Jose 2006, Liu et al. 2004, Pitkow et al. 2002
Explicit	User's categorical interests, and user supplied information (e.g. user can provide explicit relevance feedback or scrutinise the model)	Client-side and user intervention	Micarelli and Sciarrone 2004, Chen and Sycara 1998

It should be noted that there is a high tendency in more recent literature towards the use of implicit methods for information gathering. Three reasons may be given for this tendency. The first is that users have shown to be generally reluctant to provide explicit feedback to systems (Gauch et al., 2007, Carroll and Rosson, 1987). In other words, it has been shown that users dislike the idea of having to exert the extra time or effort required to explicitly supply information to a system; they would prefer to see that the system is correctly "guessing" what kind of information they need instead of them having to specify their needs or clarify their intentions to the system explicitly (Budzik and Hammond, 2000). The second reason is that some studies, such as (White et al., 2002), have shown that personalised systems can equally benefit from implicitly gathered information as from explicitly gathered information. The third

reason is that implicit feedback generates masses of data, far more than could be gathered by explicit feedback. All this has encouraged many systems to make use of search history for IR personalisation. **The PMIR framework proposed in this thesis mainly uses the implicit approach to information gathering (i.e. analysing search history) and, in addition, also gathers basic demographic information about the users when they sign-up with the system.**

The controversial decision of whether systems should collect and maintain information at the server-side or at the client side has two dimensions: the functional dimension and the privacy dimension. With respect to the functional dimension, the advantages of client-side monitoring are: (1) the availability of a richer set of information that is beyond the reach of a server-side system; and (2) part of the system's burden of processing information (computing resources) is taken away from the server. However, the drawbacks of client-side systems are: (1) they usually require the installation of a certain application or plug-in at the client's machine, either to monitor or to process data, which some users may reject; (2) logged information is not available or not complete if the user uses the system from multiple machines; and (3) it would not be possible for the system to perform any collaborative or collective processing over all the user models and usage information, which is the kind of processing that many search engines need to do in order to draw conclusions about popular and high quality pages that receive many hits (views). **The PMIR framework proposed in this thesis collects and maintains search logs at the server-side; in addition to the aforementioned advantages of the server-side approach, the nature of the experiments carried out in this study mandates that the usage information be available for analysis for the sake of evaluation and research.**

2.3 Information Representation

2.3.1 Overview

This section is concerned with the second stage of PIR systems, which involves the storage and representation of the information that was gathered in the first stage. In many systems, a user model is constructed in order to represent the user's interests in an individualised manner. However, some personalised retrieval systems maintain an aggregate representation of users' preferences and general usage behaviour. This kind of collective information is used for personalisation across the cohort of aggregated users. In this survey, both kinds of systems are covered, with a more in-depth analysis of the former systems (i.e. the ones involving an individualised user model). Moreover, this section also discusses systems where a thesaurus or

a knowledge source was used to organise the representation of the gathered information. Finally, this section discusses the different mechanisms that are used to update the information maintained by PIR systems.

Gaining an insight into the information representation stage is important because it explores the nature and structure of user models that are a core part of many personalised systems. Furthermore, it gives way to understanding how query and result adaptation are performed, as both are closely dependent on the type of information maintained by the system (details of query and result adaptation will be discussed in Section 2.4).

The analysis presented in this section is carried out over the following three criteria:

- **Existence of an individualised user model and scope of interests:** the first criterion examined in this part of the analysis is concerned with systems which make use of an individualised user model and the scope of user interests maintained by the model (i.e. short-term or long term interest). Hereafter, the term *individualised user model* will be used to refer to the notion of the explicit existence of a user model in a system, regardless of the approach by which the model's information was gathered (be it explicitly or implicitly).
- **Usage information / user model representation:** the second criterion, which can be regarded as the most important criterion of this section, is concerned with how user and usage information is represented. This involves both systems which make use of individualised user models and systems that represent information on an aggregate level.
- **Dynamism of user model and information update scheme:** the third criterion over which systems are discussed in this section is the degree of dynamism of the information stored in the user model and the mechanisms in place for updating this information. The information stored in the user model could be static, such as personal characteristics or demographic user information (which are rather permanent), or dynamic, such as the user's interests (which usually evolve with time).

The literature analysis presented in the following section goes through the information representation stage of the surveyed systems according to the three criteria outlined above.

2.3.2 Review

2.3.2.1 Existence of an Individualised User Model and Scope of Interests

A key component in many PIR systems is the user model which maintains the user's information on an individualised level, especially the terms that represent the user's search interests. These interests could be long-term or short-term interests.

In the context of IR systems, long-term interests are regarded as persistent interests that can be exhibited in the user's search history on the long run. Inferring these interests from past searches can help in enhancing similar future searches (Qiu and Cho, 2006, Speretta and Gauch, 2005, Psarras and Jose, 2006, Liu et al., 2004, Pretschner and Gauch, 1999). This is done by analysing the text of the user's queries and the clicked documents (or their snippets) and extracting key terms from them, for example by selecting the most frequently appearing terms. Interest terms are then used for adapting future queries or their results so that documents that are more relevant to the user are retrieved and displayed to the user at higher ranks. Besides harvesting interest terms from queries and clicked documents, some systems infer the users' long-term interests from their desktop documents, emails, or calendar items (Chirita et al., 2007, Teevan et al., 2005).

Short-term interests are regarded as ephemeral interests that are usually satisfied by a few ad-hoc searches in a relatively shorter period of time (typically, one search session). Short-term interests are usually harvested from submitted queries and retrieved documents within one search session and used to personalise the search immediately within that search session (Ruvini, 2003, Shen et al., 2005).

Some systems perform personalisation based on both long-term and short-term interests (Stamou and Ntoulas, 2009, Sugiyama et al., 2004). A good example is (Sugiyama et al., 2004) where the user's full browsing activity is monitored in a live manner. This enables the system to deduce both short-term and long-term user interests from terms available in the browsed Web pages. The two scopes of interests are stored separately in the user model. The TF.IDF[1] weighting scheme (Baeza-Yates and Ribeiro-Neto, 2011) is used to assign weights to the terms in order to depict different degrees of user's interest. The short-term interests are implicitly updated whenever the user clicks on a document and are thus immediately employed for personalisation in the current search session. Long-term interests, and their weights, are also

[1] Term frequency multiplied by inverse document frequency

updated when the user clicks on documents, but the difference is that long-term interests are subject to a periodic weight-decaying mechanism that reduces term weights over time. This leads, in the long-run, to preserving only persistent interests that frequently appear in the user's browsing history. Mechanisms for updating user models are discussed in more detail in subsection 2.3.2.3.

2.3.2.2 Usage Information / User Model Representation

Various techniques and data structures can be used to represent user and usage information in PIR systems. This section starts by providing a discussion of the techniques used in systems that made use of an individualised user model. The discussion also involves knowledge sources that are sometimes used as the basis for representing users' interests. The section then moves on to discussing systems where usage information was represented at an aggregate level (i.e. without the use of an individualised user model).

In this study, user models which represent the user's interests are classified with respect to two dimensions: *data structure* and *content*. The data structure dimension is concerned with the underlying storage mechanism used to represent interest terms in the model. This can either be a vector-based model or a semantic network-based model. The content dimension is concerned with the nature of the terms maintained in the user model. The terms can either be **words** that are mined from user/usage information or **conceptual terms** (categorical terms) that are drawn from a knowledge source. The following review discusses the details of each of these types of user models and a summarised classification is shown in Table 2. This classification is an extension to the classification reported in (Gauch et al., 2007).

Table 2: classification of user models

Content: / Structure	Terms	Conceptual Terms
Vector-based	models where user's interests are maintained in a vector of weighted keywords	models where user's interests are maintained in a vector of weighted concepts
Semantic network-based	models where user's interests are maintained in a network structure of terms and related terms	Models where user's interests are maintained in a network structure of concepts and related concepts

A vector-based user model is made up of a feature vector, which is a vector of terms and associated weights. The weights can be determined, for example, using a term weighting scheme such as TF, TF.IDF, or BM25 (a.k.a. Okapi BM25) (Baeza-Yates and Ribeiro-Neto,

2011, Robertson et al., 1995). One way to represent the terms in the model is by using words or phrases that are mined from user or usage information.

Vector-based user models may be composed of one or more vectors. For example, in (Shen et al., 2005) and (Ruvini, 2003) only one vector was used to store the user's short-term interests. In (Sugiyama et al., 2004), two vectors were used; one for short-term and one for long-term interests. Gathering interest terms together in one vector may be appropriate for maintaining short-term interests, but perhaps not for long-term interests. This is because a short-term vector would naturally comprise fewer terms than a long-term vector as it is usually created for one search session. This is in contrast to a long-term vector where terms are continuously accumulated with every new search that the user performs. This may eventually lead to a noisy ocean of terms (i.e. a single vector that contains a wide variety of un-clustered terms) which may be ineffective to use for personalisation. To avoid this effect, some systems such as the systems described in (Psarras and Jose, 2006) and (Chen and Sycara, 1998) represented the user's long-term interests using multiple vectors; one vector per interest cluster. In such a case, terms are usually grouped together under un-labelled clusters using unsupervised text clustering techniques (Witten et al., 2011).

An example of how words or phrases are harvested from search history and how they are used to populate a vector-based user model is illustrated in (Chen and Sycara, 1998). The system builds a vector-based user model which comprises multiple vectors of interest. The terms in the vectors are weighted using the TF.IDF scheme. Interest terms are extracted from documents which the user has explicitly marked as relevant, where each vector in the model corresponds to important keywords obtained from a single document. The full text of the document is not actually used for term extraction, rather only terms which are in the query's context[1]. If a certain maximum number of vectors in the model is exceeded, then the terms from the two most similar vectors are combined together in one vector. The benefit of this approach is that, over time, similar terms will become clustered together in the model.

Another way to represent the terms in a user model is by using conceptual terms. When conceptual terms are used in a vector-based user model it is commonly known as a concept-based user model. In this kind of model, the user's interests are represented by categorical terms that are drawn from some sort of knowledge source. Knowledge sources could be in the form of any of the following:

[1] Terms that surround the query terms in the document, extending for example to five words before and five words after each query term

- Domain models that are developed by human domain-experts such as databases of domain-specific terminologies.
- General knowledge repositories developed by human contributors such as Wikipedia[1].
- Web taxonomies or concept hierarchies such as ODP[2].
- Rich ontologies such as SUMO[3].
- Thesauri such as WordNet[4] (although thesauri might not be regarded as knowledge sources in the formal meaning, they are considered as rich sources of linguistic and semantic language knowledge, and therefore are sometimes used to organise the terms in the model).

The use of conceptual or categorical terms in concept-based user models serves to organise the user's interests with respect to the common terms used in a domain. The combination of a knowledge source with a user model is also known as an overlay model (Brusilovsky and Millán, 2007).

In MiSearch (Speretta and Gauch, 2005) two alternative vector-based user models are proposed to represent the user's long-term interests. The first comprises concepts extracted from the user's queries, and the second comprises concepts extracted from the snippets of clicked documents. Each user model is made up of multiple vectors; one per interest category. Both user models represent their categories and concepts based on the ODP hierarchy. The study concludes that the two kinds of user models are equally capable of modelling the user's interests. A number of other systems also represented their vector-based user models using concepts from the ODP hierarchy (Qiu and Cho, 2006, Liu et al., 2004, Pitkow et al., 2002).

Another example of concept hierarchies that were used for constructing user models was the Magellan[5] concept hierarchy which was used in the OBIWAN system (Pretschner and Gauch, 1999) to represent the user's long-term interests. The TF.IDF weighting scheme was used to weight the concepts stored in the user model and the document terms from which the concepts were extracted. Similarities between documents and the user model were computed using cosine similarity.

User models can be represented using a semantic network structure. In this case the model is made up of nodes and associated nodes that capture terms and their semantically-related or co-

[1] http://www.wikipedia.org/
[2] Open Directory Project: http://www.dmoz.org
[3] Suggested Upper Merged Ontology: http://www.ontologyportal.org/
[4] http://wordnet.princeton.edu/
[5] Magellan was a project associated with the *Excite* search engine. According to (Gauch et al., 2007), when Magellan ceased to exist, the authors of OBIWAN switched to ODP

occurring terms respectively. Weights can be assigned to the nodes, their associated nodes, and the links between them. A key feature of semantic network-based models is that they can model the relationship between a key term or concept and its associated terms (e.g. synonymous terms or co-occurring terms in a document collection). The mapping between terms and related terms can be achieved using a thesaurus or a domain model. For example, the SiteIF system (Stefani and Strapparava, 1998) uses WordNet to obtain semantic similarity between words (e.g. synonyms). A main node holds a weighted term that represents a user's interest. The terms come from documents that were clicked by the user. Moreover, semantically related terms to the main term are obtained from WordNet and stored into associated nodes. The associated nodes are connected to the main nodes using weighted links. The link weights represent the frequency of appearance of the associated terms with the main term in a document. Another example is the user model used in the WIFS system (Micarelli and Sciarrone, 2004) which is, in part, based on a semantic network representation.

Some studies performed search personalisation on an aggregate level. Aggregate personalisation involves the utilisation of usage information in a collective manner where the search process is adapted to the needs of the many rather than the specific needs of the individual. In these studies, no user model is used for storing interests; rather, a general representation of usage information is used. For example, the I-SPY system (Smyth and Balfe, 2006) keeps track of all users' queries and their clicked documents in a matrix called the hit matrix. The rows of the matrix represent the queries and the columns represent the documents (document identifiers). A cell in the matrix holds the number of times that the designated document was clicked for the corresponding query. This representation can be thought of as storage of query-document pairs along with their click frequency (hits). Click frequencies are used by the system for assigning higher ranks for frequently clicked documents in the list of retrieved results to a common query.

Similarly, in (Agichtein et al., 2006a) aggregate usage information was maintained in the form of query-document pairs in a model called the Implicit Feedback Model. However, the model stored a richer set of information about each pair. The model, which is represented in a vector-based manner, comprised a wide range of query-document aspects, such as clickthrough information (e.g. tracking the document's click frequency in relation to the click frequency of other documents that appeared higher or lower in the ranked result list), browsing information (e.g. average page dwell time), and textual information (e.g. overlap between the query terms and the terms of the document's URL, title, and snippet).

The authors in (Gao et al., 2007) also process pairs of queries and documents but with the aim of deducing the degree of similarity between queries of different languages. The goal is to improve cross-language search by keeping records of queries and candidate similar queries from other languages that could be used for cross-lingual query suggestions.

The work reported in (Yin et al., 2009) is also motivated by the idea that query logs reflect the wisdom of the crowds, where users may seek the same information using different queries. Queries and clicked documents are represented using a Query-URL graph. The Query-URL graph is a bipartite graph where the first set of vertices represents the queries and the second set represents the documents. The edges connecting the vertices of the two sets represent clickthrough information. The random walk algorithm is applied on the graph, which generates probabilities between queries, where higher probabilities reflect higher query similarities. These similarities are then used to improve future searches by query adaptation. A limitation that was addressed in the study was that the random walk algorithm does not work well with queries for which no results were clicked. This challenge was overcome by textually comparing such queries with all other queries in the logs (using cosine similarity) to find ones that can be deemed similar.

2.3.2.3 Dynamism of user model and information update scheme

Some user models are static, while others are dynamic in nature (Golemati et al., 2007, Hothi and Hall, 1998, Rich, 1983). Static user models are ones that maintain user information that is less likely to change over time and are therefore not subject to continuous updates. Examples of static information are personal characteristics, background knowledge, and demographic information. Maintaining static information allows PIR systems to group users into stereotypes and make high-level personalisation decision (e.g. localise the system's GUI based on the user's language, or adapt some of the services based on the user's geographic location). Dynamic user models, on the other hand, are ones that keep track of information that evolves over time. For example, models that maintain short-term user interests are usually created on-the-fly and are updated frequently over the span of the user's search session. Long-term interests can be considered as dynamic information as well if the system has a revision or update mechanism for them in place (e.g. increasing or decreasing the weights of the interests on a periodic basis, or adding new interests). More user-focused personalisation decisions can be made when the system maintains dynamic information; decisions that cater for the current user's context and interests.

Many PIR systems implement update mechanisms in order to ensure that they maintain accurate and up to date information about the user. This mechanism can be triggered upon a certain user action, such as a click on a document or the provision of explicit relevance feedback about a document (Shen et al., 2005, Sugiyama et al., 2004, Ruvini, 2003, Chen and Sycara, 1998); at which point, information can be updated on-the-fly and the newly available information may be immediately employed for personalisation in the current search session. Update procedures can also be invoked by configuring the system to periodically revise the weights of learnt interests; a mechanism known as decaying or aging (Micarelli and Sciarrone, 2004, Stefani and Strapparava, 1998, Asnicar and Tasso, 1997).

In systems where personalisation is based on short-term interests, the update mechanism may be invoked several times within the same search session. This is to ensure that any new piece of information that becomes available, following a user action, would reflect in the model and would be immediately employed for personalisation. For example, in (Ruvini, 2003) the model only keeps track of the user's current interests for a given query and the results browsed for it. That is, for every new query submitted by the user a new user model is created and is continuously updated as the user clicks on results. An insight into this updating mechanism can be gained by having a closer look at the system; the personalised search system is wrapped around the Google search engine and is mainly intended for use on limited-display devices (e.g. mobile phones) where only a small number of results can be displayed per page. A supervised machine learning approach (Support Vector Machines) is used to construct and update the model where a text classifier is trained on features extracted from the snippets of clicked result. The classifier operated under the assumption that clicked results are positive examples (of what is relevant to the user) and unclicked results are considered negative examples. When the user clicks on a result from the displayed page of results, the positive and negative examples are passed to the classifier to form a model of user's interests. Then, behind the scenes, the same query is re-submitted to Google and the top-N retrieved results are passed to the classifier to be labelled. Two groups of results are then formed; one for relevant result and one for non-relevant results. The ranking of Google is preserved for the results within each group. The user then actually avails of personalised results when s/he clicks to view the next result page. On the new page, a set of previously unseen results is displayed, where relevant results are displayed above non-relevant ones.

An example of updating schemes implemented in systems where usage information was maintained at an aggregate level can be found in the I-SPY system (Smyth and Balfe, 2006). In I-SPY, where a matrix was used to represent hits with respect to query-document pairs, two update issues were discussed by the authors: (1) documents that were indexed by the system at

earlier times will tend to have higher click frequencies than more recent ones, which may cause their rank to be higher even if more recent documents are more relevant to a given query; and (2) documents could be removed from the Web, thus leaving erroneous entries in the documents index. These issues were addressed by implementing two update schemes: first, the hit values (click frequencies) are reduced over time, and second, a garbage collection mechanism is run periodically to verify that indexed documents still exist on the Web.

2.3.3 Summary and Discussion of Information Representation

The previous section provided a review and analysis of how information is represented in different PIR systems. Table 3 provides a condensed view of the approaches discussed in this section.

Table 3: summary of information representation approaches

Existence of User Model and Scope of Interests	Usage Information /User Model Representation	Use of Thesaurus or Knowledge Source	Dynamism of User Model and Information Update Scheme	Example Publications
Yes, short-term interests	Vector-based user model (keywords)	No	Dynamic: immediate update when the user clicks on a document	Shen et al. 2005, Ruvini 2003
Yes, long-term and short-term interests	Vector-based user model (keywords)	No	Dynamic: updated upon every new Web page that is browsed by the user. & Periodic decaying of interests.	Sugiyama et al. 2004
Yes, long-term interests	Vector-based user model (keywords)	No	Static+Dynamic: updated upon user's explicit relevance feedback for a document, or model construction process can be repeated when needed	Chirita et al. 2007, Teevan et al. 2005, Chen and Sycara 1998
Yes, long-term interests	Vector-based user model (conceptual terms)	Yes (ODP, Magellan, etc.)	Static+Dynamic: model construction process can be repeated when needed	Qiu and Cho 2006, Speretta and Gauch 2005, Liu et al. 2004, Pretschner and Gauch 1999

Yes, long-term and short-term interests	Vector-based user model (conceptual terms)	Yes (Hybrid ontology of ODP, WordNet, MultiWordNet, SUMO)	Static+Dynamic: model construction process can be repeated when needed	Stamou and Ntoulas 2009
Yes, long-term interests	Semantic-network-based user model (weighted nodes of keywords and weighted links connecting co-occurring words)	No	Dynamic: updated upon user's explicit relevance feedback for a document. & Periodic decay of weights of nodes and links	Asnicar and Tasso 1997
Yes, long-term interests	Semantic-network-based user model (weighted nodes of keywords and weighted links connecting semantically related or co-occurring words)	Yes (WordNet or domain stereotypes built by human experts, etc.)	Static+Dynamic: updated upon user's explicit relevance feedback for a document or periodic reconsideration of weights of nodes and links	Micarelli and Sciarrone 2004, Stefani and Strapparava 1999
No	Multiple history matrices (one per community) to keep track of queries and frequency of clicked documents	No	Dynamic: click frequencies are decayed over time. & Periodic garbage collection mechanism (to check that indexed pages still exist on the Web)	Smyth and Balfe 2006
No	Statistical/Probabilistic information involving the relations between users, queries, clicked documents, or other features	No	Static+Dynamic: machine learning process can be repeated when needed	Yin et al. 2009, Gao et al. 2007, Agichtein et al. 2006

The literature analysis reveals that relatively few PIR studies performed personalisation based only on short-term user interests. The benefit of keeping track of the user's short-term interests is that it accounts for the user's ad-hoc information needs and allows systems to perform personalisation on-the-fly (Ruvini, 2003, Shen et al., 2005). To this end, the authors in (Shen et al., 2005) argue that the majority of the user's searches come from ad-hoc information needs which are usually satisfied by a small number of searches. Thus, they conclude that personalisation should target the scope of short-term user interests.

However, a concern that is associated with performing personalisation based only on short-term user interests (i.e. operating only on information obtained from the current search session) is that very little information is available to base the personalisation decisions on. For example,

the analysis carried out in (Jansen et al., 2000) gives an idea of how limited the information from one search session could be. The analysis shows that the average number of queries per session is 1.6 queries and that the average number of results clicked in a session is approximately 2.4 results. Following on this, the large-scale PIR study conducted by (Teevan et al., 2005) shows that the amount of user and usage information available indeed affects the degree to which personalisation can be effective.

The success reported by the rather contradicting studies in the literature (concerning the use of short-term vs. long-term interests) may actually suggest that a good practice would be to combine evidence from both scopes of interests to personalise the user's searches. This can be achieved by partially basing personalisation decisions on short-term interests, yet relying on long-term interests when it makes sense to do so. This combined approach was shown to be useful in a number of studies such as (Stamou and Ntoulas, 2009) and (Sugiyama et al., 2004).

The PMIR framework proposed in this thesis follows on the combined approach that keeps track of both the short-term and long-term search interests in the user model. This is achieved by continuously updating the user model in the framework with evidence from every new search that the user performs. Furthermore, the PMIR framework adopts the technique of representing the user's interests as keywords using multiple clusters of interests. Thus, the vector-based representation is used as the underlying structure of the user model. The user models proposed in this thesis are partially based on the user models presented in (Speretta and Gauch, 2005) and (Chen and Sycara, 1998).

It should be noted that the use of individualised user models in PIR was mostly investigated for monolingual PIR systems (especially for English, probably due to its inherent popularity). In a multilingual world, information that is relevant to the user's information need may exist in languages other than the language that the user used to query the system. With the advent in automatic translation techniques, users can access documents that are beyond their native language. Furthermore, a proportion of the users may very well be familiar with multiple languages and are able to comprehend documents in those languages. **A key contribution of the study carried out for this thesis is taking into consideration that both the users and/or the content can be multilingual. The study investigates the effectiveness of the multilingual approach to search personalisation, and especially how to construct user models that depict the attributes and interests of a multilingual search user.**

2.4 Personalisation Implementation and Execution

2.4.1 Overview

This section focuses on the implementation and execution of the personalisation process. Personalisation in PIR systems is generally performed by adapting the query and/or the results. Adaptation can either target specific individualised user needs, or target common needs of groups of users. This section also discusses the types of services provided by the reviewed systems.

As this section explores the details of how personalisation is implemented and executed in different systems, it can thus be regarded as the core part of this state-of-the-art survey. The analysis presented in this section is carried out over three criteria: the system's type, the personalisation scope, and the personalisation approach. The following is an overview of these three criteria.

- **Type of service:** the first criterion is concerned with the domain or the type of IR service that a system offers, such as monolingual Web search, multilingual Web search, personalised news, eLearning, etc.
- **Personalisation scope:** the second criterion is the scope on which personalisation is performed. This survey classifies PIR systems according to the scope of personalisation into three categories: individualised scope, community scope, and aggregate scope.
- **Personalisation approach:** the third, and most important, criterion in this part of the analysis is how personalisation is performed. This can be by query adaptation, result adaptation, or both.

2.4.2 Review

2.4.2.1 Type of Service

This section discusses the types of services provided by a variety of PIR systems, and shows how the aspects of personalisation offered by these systems differ based on the services they provide.

Textual search is a prominent application of IR. Many systems presented in academic literature and ones which are currently deployed on the Web offer search services (Vallet et al., 2010, Stamou and Ntoulas, 2009, Yin et al., 2009, Speretta and Gauch, 2005). Some systems extend

this to cross-language search, where a translation mechanism is used to translate the query or the document in order to allow the retrieval of documents that are not necessarily in the same language as the query (Nie, 2010, Oard, 2010, Gao et al., 2007, Ambati and Uppuluri, 2006, Cao et al., 2007).

Since this section entails discussions of personalisation in cross-language search systems, it is important at this point to give a brief account of the commonly used translation techniques. Translation techniques generally fall under three categories: *Bilingual Dictionaries*, *Example-based* (a.k.a. Corpus-based), and *Machine Translation* (MT). Bilingual Dictionaries are machine readable dictionaries that can be used to obtain multiple suggestions of translations for a given word in a source language into a certain target language. The idea of Example-based translation techniques is to apply statistical analysis on words or phrases in parallel or comparable corpora in different languages to obtain probabilities of translations between them. MT software is optimised for translating whole sentences while maintaining proper grammatical rules and well-formed sentences in its output. Several studies have investigated means of improving Machine Translation (MT) systems, which are widely used in the industry and in recent research in the fields of IR and localisation (Magdy and Jones, 2011, Stroppa and Way, 2006).

In personalised search, personalisation is often implemented by adapting the query (e.g. automatically or semi-automatically modifying the query terms to obtain a better description of the user's information need), adapting the results (e.g. re-ranking the list of results so that more relevant results are displayed higher in the list), or both. A detailed discussion of these approaches is provided in subsection 2.4.2.3.

Some systems study the provision of a personalised search service on hand-held devices (e.g. mobile phones). In these cases, the study considers several HCI (Human-Computer Interaction) factors in the adaptation process. For example, in addition to investigating how to adapt the results with respect to the user, the authors in (Ruvini, 2003), also investigate the adaptation of result lists with respect to the limited display offered by mobile devices.

In most search systems, results are typically presented to the user in the form of a ranked list of results. To this effect, if result adaptation takes place, it mainly involves altering the ranks of these results in the list. However, the authors in (Steichen et al., 2009) present a different approach to result adaptation and presentation. The authors propose a search system that operates in an eLearning environment, where instead of displaying a typical ranked list of results, the content of the results is dynamically re-composed to generate a tailored hypertext

presentation. The search is performed over a closed corpus of domain-specific Web pages that were harvested from the Web. Furthermore, the harvested Web pages were manually annotated to indicate the level and nature of the content presented in them (e.g. introductory level information, advanced level information, theoretical/conceptual content, technical illustration/example, etc.). The user model contained information about the user's prior knowledge with respect to the learning domain and the level of the user's experience in that domain. This information, together with the document annotations, provided adequate information to an adaptive engine so that it can re-structure the content of documents and display it in a presentation-style format that suits the user.

Moreover, the authors extend the work and evaluate its application in the customer support domain (Steichen et al., 2011). They propose a search system that is intended to assist users who are searching for solutions to technical problems concerning a certain product. The system performs adaptive composition of a personalised hypertext presentation based on technical support content from heterogeneous data sources (open corpora, closed corpora, social networking, etc.). This content comprises technical information obtained from the product's manuals as well as user-generated content related to that product (e.g. discussion forums on the Web). Personalisation is based on the user's level of expertise with respect to individual product features and the user's query intent (e.g. "find out about product features" or "get a how-to"). In order to compose the result presentation, multiple versions of the query are submitted to the retrieval component. The queries are expanded using different terms and meta-data information obtained from the domain and content models. The results retrieved for these queries are then re-composed according to the user information and query intent.

A number of systems offer personalised news services. In such systems, personalisation is concerned with "guessing" which pieces of news would be of interest to a particular user. For example, the PersoNews system described in (Katakis et al., 2009) disseminates RSS-feed news items to users based on their interests. Machine learning techniques are used to learn the users' interests based on the kind of news feeds that they subscribe to and the news items that they explicitly mark as relevant. The learnt interests are used to filter out news that is not relevant to the user. Another example is the WebMate system (Chen and Sycara, 1998) which also operates on the news domain. WebMate offers two services to its users: searching on news corpora (retrieval of information) and filtering news items according to the user's interests (filtering of information).

The area of Information Filtering (IF) (Belkin and Croft, 1992, Oard, 1997) is closely related to IR. Yet, there are a number of differences that distinguish between the two areas (Hanani et al., 2001, Belkin and Croft, 1992, Brusilovsky and Tasso, 2004). First, IR systems are generally intended for ad-hoc information needs, while IF systems are intended for persistent information needs that are exhibited on the long-run. Second, in IR, information needs are represented as queries, while in IF, the user models themselves can be considered as the representation of the user's information need. Third, the purpose of IR systems is to locate information, while the general purpose of IF systems is to disseminate information.

It may be deduced from the argument above that PIR based on user's long-term interests is essentially IF. Yet, with respect to the analysis and scope of this survey, the thin line that separates the two is how information is sought. The analysis in this PIR survey is concerned with search systems where the initial action in the information seeking process is the user submitting a query to the system with the aim of satisfying an information need (be it ephemeral or persistent). On the other hand, IF systems are regarded as systems where there is a dynamic flow of unsolicited information that needs to be disseminated to users. The initial action in the IF process is thus the arrival of an incoming document. This distinctive feature was stated in (Belkin and Croft, 1992, Hanani et al., 2001) as one of the features which generally differentiate between IR and IF. However, given the gray area between PIR (with long-term interests) and IF, a number of systems, such as (Chen and Sycara, 1998) provided a combination of both services in a unified interface (i.e. a system that supports both an information-initiated approach and a user-initiated approach to the provision of information).

The WIFS system (Micarelli and Sciarrone, 2004) is another example of systems which offered both, a search service and a filtering service. WIFS operated on domain-specific corpora, where the system allowed users to search for academic publications in the field of computer science, as well as recommend publications to them based on their exhibited interests in the user model.

2.4.2.2 Personalisation Scope

Various approaches to personalisation are exhibited in PIR systems. A key distinguishing aspect of these approaches is the level of information detail on which they operate. Some systems operate on aggregate usage information as exhibited in search logs, while other systems take a more fine-grained approach by operating on the scope of individual user information. This section discusses the three scopes on which personalisation is performed, which are *individualised*, *community-based*, and *aggregate-level* personalisation.

Individualised personalisation is when the system's adaptive decisions are taken according to the information about each individual user as exhibited in the user model (Steichen et al., 2011, Stamou and Ntoulas, 2009, Speretta and Gauch, 2005, Shen et al., 2005). The advantage of this approach is that the system becomes truly personalised as it addresses the needs of a specific user; taking into consideration this user's individual characteristics, interests, prior knowledge, language, country, and so on. This approach may lead to higher satisfaction degrees for the user. Yet, one of the issues associated with the individualised approach is the fresh start problem (a.k.a. cold start) where it is the case that a new user has just registered with the system and there is very little or no information available about him/her to work with at that point.

Among the challenges facing individualised personalisation, and personalisation in general, are the *effect of getting it wrong* and *risk vs. reward* (Wade, 2009, Vassiliou et al., 2003, Espinoza and Höök, 1995, de La Passardiere and Dufresne, 1992). As personalisation may sometimes "go astray", PIR systems have to take into consideration that delivering an inaccurate personalisation service can have a profound negative effect on the user's perception of the system. In other words, inferences made by personalised system about their users are essentially a "guess"; the harm of getting it wrong can be greater than the benefit of getting it right. Moreover, some personalised systems may attempt to perform a limited form of personalisation based on a limited, yet reliable, set of attributes and information available about the user. In spite of such limitation, the reward of such cautious form of personalisation may be considered sufficient –to a certain degree– to satisfy the users of the personalised service. Performing a more aggressive form of personalisation entails a higher degree of risk, yet it might not produce huge transformations in the personalised service; in which case, the reward may not be worth the risk. Thus, it is important for PIR systems to investigate successful tradeoffs for delivering the right amount of personalisation in a careful manner. It is also important for PIR systems to take into consideration the effects that personalisation introduces to the interface of the system; users should not be surprised or disoriented by the changes incurred by the adaptive service. Therefore, designers of PIR systems should provide adequate balance between the usability of the interface and the potential effectiveness of the system.

Community-based personalisation takes a step further from individualised personalisation as information can be shared between the user models (Teevan et al., 2009, Sugiyama et al., 2004, Mei and Church, 2008). The system's adaptive decisions are then based on a wider scope of users, and not just a single user. This may be the case when a system groups the users into stereotypes (Brajnik et al., 1987) according to certain similarity criteria between their user models; at which point the system can judge the relevance of a certain document or item to a

user based on the information of other users who belong to the same group in a collaborative manner. It can also be the case when information from some user models is used to determine or alter the weights of interests in other user models. Community-based personalisation is more prominently used in the area of Recommender Systems (Schafer et al., 2007).

The main consideration in community-based systems is how users are grouped together. This can be done in the following ways:

- Manually pre-defining labelled groups in which users can join when they sign up with the system. These groups can be related to topics of interests (e.g. music, sports, etc.).
- Using machine learning techniques (e.g. clustering techniques) to automatically form clusters of users based on similarity features between their user models (e.g. textual similarity of interest terms).
- Including content information, in addition to user information, when processing user models for similarity. This is, for example, the case with content-based recommendation systems (Pazzani and Billsus, 2007).
- Grouping users based on their demographic information (e.g. language, location, line of work, etc.).

The authors in (Mei and Church, 2008) argue that too much personalisation may sometimes degrade retrieval effectiveness just as severely as no personalisation at all. The authors suggest that personalisation should sometimes "back off"[1] to a larger number of users, rather than a single user, when not enough individual user information is available. To this effect, the authors in (Teevan et al., 2009), investigated how a user's model can be augmented with information from groups of similar users with the aim of improving retrieval effectiveness. Different ways to form groups of users were investigated, including demographic information. The authors called their approach "groupisation" (as opposed to personalisation).

Aggregate-level personalisation refers to the notion of a system that does not explicitly make use of a "per-user" model to represent users; in which case personalisation is guided by collective usage data as exhibited in search logs (Agichtein et al., 2006b, Gao et al., 2007, Sun et al., 2005, Smyth and Balfe, 2006). For example, this is the case when a system ranks Web pages in a result list based on the number of times each Web page was browsed by users.

[1] (Mei and Church, 2008) used the term "back off" to refer to the notion of broadening the scope of the number of users on which personalisation decisions are based.

It may be argued that systems which perform personalisation at an aggregate level should not be regarded as "personalised" systems, since they do not make use of a user model and thus do not tailor their service to a specific "person". However, when considering search personalisation in a broader sense, the objective is retrieving documents that satisfy users' information needs; this may indeed start at the higher level of adapting to the needs of the majority of users. Adapting to the needs of the majority can give some kind of guidance as to what an individual user may need. For example, a common information need can be inferred if at some point in time a large number of users issued the same query and clicked the same results for it. Therefore, drawing on this inferred common need may serve in adapting similar future searches. Yet, the success of aggregate level systems is reliant on their capability of accurately analysing and interpreting aggregate usage information so that they could deduce the true needs of the majority.

The classification of PIR systems in the manner presented in this section can be regarded as a way to identify the scope on which each system operates, rather than an attempt to define completely distinct categories. In this sense, the three introduced scopes may be regarded as special (or more generalised) cases of each other, where the individualised scope indicates that personalisation is performed per "only one user", the community-based scope indicates "more than one user", and the aggregate-level scope indicates "all users treated as one".

2.4.2.3 Personalisation Approach

Personalisation in PIR systems can be achieved by query adaptation, result adaptation, or a combination of both. In other words, adaptation can be performed over the information that users submit or the information that they receive. In systems that offer a multilingual service to the users, the adaptation process may also include query and result translation (Oard, 2010, Oard and Diekema, 1998).

Query Adaptation:

Studies, such as (Furnas et al., 1987), show that users may not always be successful in using representative vocabulary when locating objects in a system. Therefore, query adaptation attempts to expand the terms of the user's query with other terms, with the aim of retrieving more relevant results (Manning et al., 2008). In some cases, source query terms may be completely replaced by other terms. Query adaptation also involves altering the weights (significance) of the query terms when submitting them to the retrieval component of the system.

Six techniques are mainly used for obtaining terms for query expansion, which can be classified in terms of whether they are *user-focused or not* and whether they are *implicit or explicit*:

1. Processing the user model: this involves the implicit selection of expansion terms from the user model (Chirita et al., 2007, Psarras and Jose, 2006, Shen et al., 2005).
2. Processing aggregate usage information: this involves implicitly obtaining expansion terms from the query logs and/or their associated clicked documents under the assumption that the majority of user clicks would be on documents that are relevant to the queries they submitted (Yin et al., 2009, Gao et al., 2007, Cui et al., 2003, Billerbeck et al., 2003).
3. Pseudo-Relevance Feedback (local analysis): this involves performing an initial retrieval round (that takes place behind the scenes) using the source query and then implicitly selecting expansion terms from the top N retrieved documents (or their snippets) under the assumption that most of them would be relevant to the source query (Leveling and Jones, 2010a, Ogilvie et al., 2009, Cao et al., 2008, De Luca and Nürnberger, 2006).
4. Global analysis: this involves the implicit selection of expansion terms from a thesaurus (e.g. WordNet), a knowledge source (e.g. Wikipedia), or a large corpus (based on co-occurrence statistics in this corpus) (Callan et al., 1995, Xu and Croft, 1996, Nguyen et al., 2008).
5. Relevance feedback (a.k.a. explicit relevance feedback): this requires that the user explicitly provide relevance feedback about a number of documents from an initial set of retrieved results where documents marked as relevant are processed to obtain expansion terms (Ruthven and Lalmas, 2003, Harman, 1992b, Salton and Buckley, 1990).
6. Interactive Query Expansion: this involves GUI (Graphical User Interface) that allows the user to explicitly select expansion terms from a candidate list of terms suggested by the system (Bast et al., 2007, Ruthven, 2003, Efthimiadis, 2000, Harman, 1988).

Table 4 shows a summarised classification of query expansion techniques. Furthermore, details of these techniques are analysed across a number of example systems below.

Table 4: classification of query expansion techniques

	User-focused		Not user-focused
	Individualised	**Aggregate**	
Implicit	User Model	Usage Information (Search Logs)	Pseudo-relevance Feedback (Local Analysis) & Global Analysis
Explicit	Relevance Feedback & Interactive QE		

Processing the user model. The work presented in (Chirita et al., 2007) is an example of systems where query expansion terms are obtained from the user model. The user's interests are inferred from his/her Personal Information Repository, which is the collection of their desktop documents, emails and cached Web pages. The first step towards the selection of terms for expansion involves identifying documents in the user's repository which contain the source query terms. Second, these documents are sorted in descending order with respect to the source query terms, based on a modified term frequency (TF) weighting scheme. Third, query-focused summaries of the top K documents are produced. Fourth, all the terms of the summaries are extracted and are sorted according to document frequency (DF) weighting based on the number of summaries they appeared in. Finally, the top four terms are used as expansion terms for the source query. The authors also conducted a set of experiments to determine the adequate number of terms to use for expansion. They suggested that the decision should be dynamically based on query features such as query length (number of terms in the query), query scope (IDF score of the query), or query clarity (query ambiguity). The use of such features in dynamic decisions for query expansion is an emergent approach that is known as *selective query expansion*.

In (Koutrika and Ioannidis, 2004) query adaptation is performed by re-writing the whole query based on a set of rules maintained in the user model. The rule-based query re-writing process is used for personalising structured search across a database of movie information. The system substitutes the submitted query with multiple queries using a set of rules that govern the process. These rules are based on the user's individual movie preferences. The queries are connected together in a disjunctive manner using the "OR" operator. For example, if a certain user, who is known to prefer comedy movies, enters a source query that requests a list of

movies in a certain year, then the system will replace the source query with a query that seeks a list of movies of the comedy type from that year.

Processing aggregate usage information. The study carried out by (Yin et al., 2009) is an example of performing query adaptation based on aggregate usage information as exhibited in search logs (submitted queries and snippets of clicked results). The authors are motivated by the idea that users may seek the same information but using different queries. The authors use machine learning techniques to learn the similarities between queries in the logs, and employ these similarities for query adaptation. They argue that traditional pseudo-relevance feedback has two drawbacks: (1) processing the full text of feedback documents (as opposed to processing only the snippets) obtained in the initial retrieval round is considered an overhead to the system; (2) not all feedback documents are guaranteed to be relevant, thus, some "bad" terms might be extracted from them (i.e. terms that may be harmful to retrieval effectiveness). The authors address these two issues by: (1) using the text of snippets instead of documents, which is further supported by the idea that, before clicking on results, users actually examine the result snippets in order to get a hint of how far a document is relevant to their information need; and (2) only selecting snippets that exceed a certain score threshold, where scores are assigned to snippets based on their rank and their similarity with the source query and similar target queries in the logs.

Query adaptation based on usage information is also investigated in (Gao et al., 2007). Furthermore, the authors extended into the multilingual dimension. Given a source query in a certain language, the system obtains related queries from other languages by analysing multilingual search logs. This technique is also known as Cross-Lingual Query Suggestion (CLQS). CLQS can be viewed as a technique that combines query translation and adaptation into a single process, where the formulation of the source query is expanded (or replaced) with common formulations of similar queries exhibited in the multilingual logs. The authors use machine learning algorithms in order to learn a cross-lingual similarity function that determines the degree of similarity between a query in the source language and another query in the target language. The process of determining cross-lingual similarity between two queries involves several features of monolingual similarity between the first query and the translation of the second query.

Pseudo-Relevance Feedback (PRF). Query expansion using PRF techniques (a.k.a. local analysis or blind relevance feedback) was subject to wide research in the field of IR. The main issue with PRF is that the process is prone to noise caused by the fraction of feedback documents that are not relevant to the query, which may degrade retrieval effectiveness. This

issue was addressed by a number of studies in a non-user-focused manner (Leveling and Jones, 2010a, Teevan et al., 2008, Cao et al., 2008, Amati et al., 2004). For example, the research reported in (Cao et al., 2008) and (Leveling and Jones, 2010a) carried out selective query expansion by investigating how automatic classification techniques can be used to identify good and bad terms for query expansion. Several features were used for the classification process, such as term distribution (the frequency of terms appearing in the feedback documents), term specificity (the number of documents in which the term appears in the entire collection), term co-occurrence (appearance of query terms with candidate expansion terms in the collection or in a thesaurus), term proximity (the number of terms separating co-occurring terms), and term string distance (the Levenstein distance between terms, which may detect terms that are morphological variants of each other).

In multilingual search systems, the PRF approach is often used to expand the query in two ways, namely: pre-translation query expansion and post-translation query expansion. Pre-translation expansion involves expanding the source query (in its source language) by terms obtained from a retrieval round performed over documents of the source language. Afterwards, the source query is translated into one or more target languages using a translation mechanism (e.g. bilingual dictionaries or machine translation systems). Post-translation expansion is then applied to expand the translated query (in its target language) by terms obtained from another retrieval round that involves documents of the target language. The authors in (McNamee and Mayfield, 2002) discussed this process and mentioned that pre-translation expansion helps in improving translation by increasing the terms that are used as input to the translation module. This helps in overcoming any limitations in the translation method or limitations caused by Out of Vocabulary (OOV) terms[1]. The authors also mentioned that post-translation expansion helps in overcoming any output errors that may be exhibited in the terms produced by the translation module. Moreover, a comparison between the two approaches was carried out and the authors concluded that combining both approaches significantly improved retrieval effectiveness more than using any one of them alone. It was also concluded that pre-translation expansion contributed more than post-translation expansion towards the observed retrieval improvement.

The work reported in (Cao et al., 2007) is another example of studies which performed non-user-focused query expansion in a multilingual fashion. In the study, Markov Chains was used to combine query translation and query expansion. Similar to the abovementioned CLQS work of (Gao et al., 2007), the process involved expanding the source query with semantically related

[1] Out of vocabulary terms are emerging words that existing translation systems may be unaware of.

terms in a different language. However, this was based on global analysis rather than local analysis.

Some other systems, such as the system presented in (Ambati and Uppuluri, 2006), investigated improving CLIR by analysing search logs but with a different focus; instead of search personalisation, the main objective of the system was to improve translation methods.

Global analysis. An example of systems where expansion terms were implicitly obtained using global analysis techniques is the INQUERY system (Callan et al., 1995). In INQUERY, query terms can be expanded with other semantically related terms. This is achieved by grouping all terms in the collection into noun groups, where each noun group consists of a phrase (up to three adjacent terms), along with all the terms that co-occur with that phrase in a pre-defined window size (e.g. within the distance of three sentences). TF and IDF are then used to weight the importance of the terms in the noun groups. Whenever a query is submitted to the system, the query terms are used to identify the appropriate noun groups. Then, related terms that exceed a certain weight threshold are selected from those noun groups and are used for expanding the source query.

Local and global analysis techniques are implicit (automatic) techniques, but are not user-focused. Opposite to those two techniques are relevance feedback and interactive query expansion, which are explicit feedback techniques that are user-focused (since the user is involved in the process).

Relevance feedback. In the relevance feedback approach to query expansion, users are asked to provide feedback about the relevance of result documents to their information need (Ruthven and Lalmas, 2003, Harman, 1992a, Salton and Buckley, 1990). This feedback can either be positive or negative, for example by marking documents on a binary scale of relevant vs. irrelevant. The system then analyses the feedback documents and modifies the source query accordingly. The new query is then used to retrieve documents that are similar to the positive examples, or filter out documents that are similar to the negative examples.

Relevance feedback is an iterative process, where users can keep providing feedback for every new result list provided to them. The process may eventually converge after a number of iterations (i.e. no more significant enhancements in the precision of the retrieved result list).

Although in relevance feedback there is no user model created (in the formal sense), the process can be considered personalised because the user is involved in specifying what is relevant and

irrelevant to him/her. Furthermore, in a search session, the adapted query itself can be roughly regarded as a representation of the user's short-term (ad-hoc) interests with respect to the current information need.

Interactive Query Expansion (IQE). The IQE approach encompasses more involvement for the user (Bast et al., 2007, Ruthven, 2003, Efthimiadis, 2000, Harman, 1988). In IQE, the system suggests a set of terms, from which the user can select the ones to be used for expanding the query. An important initial step for IQE is that the system automatically produces a ranked list of candidate terms, a subset of which is presented to the user. These terms can be obtained from documents which have been marked relevant by the user or from a thesaurus, where terms that are semantically related to the query terms are identified.

Several studies conducted comparative evaluations between interactive and automatic query expansion (i.e. IQE vs. explicit relevance feedback) (Ruthven, 2003, Magennis and van Rijsbergen, 1997). It was shown that interactive techniques can sometimes be more effective than automatic techniques. However, it was also concluded that this is not always the case because IQE depends on other human factors like the degree of user's prior knowledge of the domain and the GUI of the application used to present the terms to the user.

Result Adaptation:

The other common approach to search personalisation is result adaptation. Adaptation of result lists can be performed by *result scoring*, *result re-ranking*, or *result filtering*. Result re-ranking takes place after an initial set of documents have been retrieved by the system, where an additional ranking round is performed to re-order documents based on certain adaptation aspects (e.g. displaying certain documents at higher ranks in the result list based on the user's interests) under the assumption that users are more inclined to click on results further up the list. Result filtering can be considered as a special case of (or a step further from) result re-ranking, where, after the result list is sorted in descending order of relevance scores, results that fall below a certain threshold are not displayed to the user. Result scoring involves incorporating adaptation features directly in the primary scoring function of the retrieval component of the system.

Result re-ranking and result filtering. The result re-ranking approach is commonly used in many PIR systems. A good example is the MiSearch system (Speretta and Gauch, 2005), which

is wrapped around Google search. Following a user's search, the results and snippets[1] retrieved from Google are passed to the result re-ranking component. The snippets are then analysed using text classification techniques. This is performed in order to deduce their conceptual content so that they can be assigned under appropriate ODP categories. After the concepts of the snippets have been deduced, they are compared to the concepts in the user model using cosine similarity. The results are then re-ranked in descending order of the conceptual similarity score. Several modes of result re-ranking were tested in MiSearch, where the conceptual similarity ranking was combined with the original ranking of Google. An alpha factor was used to specify a certain weight for the conceptual ranking in relation to the original ranking. The value of alpha ranged between zero and one, where a value of zero led to completely ignoring the conceptual ranking (i.e. no adaptation applied), and a value of one led to completely ignoring the original ranking. Experiments with different values of alpha showed that a value of one achieved the highest improvements for retrieval effectiveness. Therefore, it was concluded that result re-ranking can be an adequate tool for adapting to the user's information needs.

Several systems, in which the retrieval components were wrapped around well-known search engines, do not apply the result re-ranking process on the full set of results retrieved from the search engine (which could be hundreds or thousands of documents). In fact, the process is often limited to the top N documents from the result list. For example, the authors in (Speretta and Gauch, 2005) decided to limit the re-ranking process to the top ten retrieved documents. This decision was based on an experiment that they carried out which involved a number of users using a non-personalised search system. The results of that experiment showed that 94% of users' clicks occurred on the top three results in the result list. To this end, the authors further investigated the effect of the position bias phenomenon[2]. The phenomenon was investigated by randomising the ranks of the top ten results retrieved from Google before displaying them to the user. The results of the investigation showed that the top three results of Google search only received 46% of the users' clicks when they were presented in a randomised order within the list of ten displayed results. The authors concluded that users are affected by the presentation order of the results and thus continued to randomise the top ten results retrieved from Google in their baseline system.

[1] Snippets, as discussed earlier, are a form of summary or surrogate of a document and are therefore regarded by several studies in the literature as query-focused document space representations. Thus, several studies opt to process the text of snippets instead of the full text of documents when personalising result lists.

[2] Position bias phenomenon (a.k.a. trust bias) is the tendency of users to "trust" the ranking of a search engine and thereby click on the higher ranked documents even though more relevant documents may sometimes exist at lower ranks.

The authors in (Stamou and Ntoulas, 2009) also propose a system where personalisation is performed by re-ranking results that are retrieved from Google. However, a notable aspect about their re-ranking process is that the weights of user interests are not only based on historical evidence (long-term interests) but also on evidence from the current search at hand (short-term interests). The authors implement this through a number of steps:

1. The user's past conceptual preferences are identified by examining past queries and their corresponding clicked documents and then mapping them to concepts.
2. The user's current conceptual preference is identified by examining the current query (i.e. the system attempts to determine the user's current information need from the new query that has just been submitted to the system).
3. If a query that is similar to the current query was found to exist in the logs, then the conceptual preferences that were determined for the existing query in the first step are used. Otherwise, the system attempts to determine the similarity between the query and the documents listed under each of the ontology concepts (pre-classified).
4. It might make sense to perform the re-ranking process only according to the identified conceptual interests of the current query (since it is the given evidence of the current information need); however, the evidence from the current query is weak evidence to some extent because it was supported only by a few terms in a single query. Therefore, the authors determine the degree of user's interest in conceptual topics by computing a combined value of historical evidence and current evidence. To do this, an alpha value is used to explicitly specify weight for historical evidence in relation to current evidence. Lower values of alpha indicate a conservative approach that favours historical evidence (from past queries), while greater values of alpha indicate an aggressive approach that favours current evidence (from the current query at hand).
5. The retrieval process then takes place, where the current query is submitted to Google and corresponding results are retrieved.
6. The conceptual topics present in the documents are determined with respect to the ontology, and are assigned weights.
7. Finally, for each document, a relevance score is computed. The computed score involves the value obtained from the third step (user's conceptual interests) and the value obtained from the fifth step (documents' conceptual weights). The results are then re-ranked in descending order of the computed score.

Social information has also been used for result re-ranking in PIR. For example, the authors in (Vallet et al., 2010) investigated how the ranking of search engine results can be improved with respect to users if the users' interactions with social applications are taken into consideration. This was achieved by re-ranking results retrieved from the Yahoo search engine based on a user

model comprising tags extracted from the user's participation on the del.icio.us social bookmarking website. Users and documents were both represented by associated tags, where the tag distribution across 2000 users and about 160,000 documents were considered. A similar approach was also explored in (Noll and Meinel, 2007) where the system performed re-ranking of Google search results based on social bookmarks and tags harvested from del.icio.us. An advantage stated in both studies is that this approach is independent of a specific search engine, and thus any search engine can be used. However, the data sparsity problem poses a challenge to this approach, as not all Web pages returned by search engines are tagged in the del.icio.us dataset.

Result filtering can be considered as an additional step that takes place after re-ranking the results with respect to the user's interests. An example of systems which employ result filtering is the WIFS system (Micarelli and Sciarrone, 2004). WIFS offers two services to its users: Web search and Web filtering. The filtering service autonomously retrieves Web pages and filters them according to the user's interests. The pages are first sorted in descending order of relevance scores and then pages that fall below a certain threshold are discarded.

The aforementioned result adaptation systems operated on an individualised scope. Opposed to this, are other approaches where result adaptation is performed on an aggregate-level scope. For example, in the I-SPY system (Smyth and Balfe, 2006), personalisation is collectively based on the deduced interests of the majority of users as exhibited in search history. As briefly discussed earlier, usage information in I-SPY is represented in a matrix that keeps track of each query and the number of times a corresponding document was clicked for that query. In order to re-rank results for a new search, the current query is checked for similarity against all the past queries recorded in the matrix. The similarity between queries is computed using term-based similarity measures which determine the degree of textual similarity between them. The outcome of this procedure is a list of candidate queries (ones which passed a certain similarity threshold). The click frequencies of the documents that were associated with the candidate queries are obtained from the matrix. For each document, the multiple frequencies that come from considering the multiple candidate queries are combined using a normalised weighted relevance metric which combines relevance scores for document-query pairs1. The new relevance scores are then used to re-rank the documents for the current query at hand.

[1] The relevance scores were combined by calculating the weighted sum of each relevance score, and then obtaining the average. The weighing was based on the degree of similarity between the source query and candidate queries.

An innate characteristic of the result re-ranking process is that two rounds of computation take place. In the first round a function is used to score the relevance of the documents with respect to the query in a pure IR manner. In the second round, another function is used to score the documents (or the top N documents) with respect to the user. Research studies which depend on an external retrieval component (i.e. where a search engine other than their own is used, such as Google, Bing, or Yahoo) are obliged to work with the extra round of re-ranking, since they have no control over the factors of the first scoring function.

Result scoring. A number of other systems for which a retrieval component was implemented (i.e. they did not depend on one of the existing search engines) followed another approach for result adaptation: result scoring. In result scoring, only one round of scoring is performed. The adaptive factors (variables) that are used to score the documents according to the users' needs are combined together with the IR factors in the original scoring function. For example, in (Agichtein et al., 2006a) result re-ranking and result scoring were both implemented and compared to each other. The two approaches operated on the scope of aggregate usage data. In the first approach, the one based on result re-ranking, the authors used machine learning algorithms to learn a function for relevance weighting based on implicit feedback features from the search logs. However, the rank orders that were obtained from the original scoring round were not totally ignored as they were combined with the ranks produced by the new learnt function. In other words, the first approach "honoured" the original scoring method by using an additional re-ranking function that combined the rank orders obtained from both the original method and the new method. Furthermore, a factor was used to specify a certain weight for the ranks obtained from the new implicit feedback method in relation to the original method. This allowed control over the degree of bias towards the new method. In the second approach, which is based on result scoring, the authors included the implicit feedback features together with the original features in the main scoring function of the retrieval component. This allowed avoiding the extra scoring round. The experiments carried out by the authors showed that the second approach was more effective, thus they recommended performing personalisation by result scoring, rather than by result re-ranking.

Another technique for result scoring is the topic-sensitive PageRank algorithm (Haveliwala, 2002). In this algorithm, the system assigns multiple PageRank scores (Brin and Page, 1998) to each document, where each score is calculated with respect to one of ODP categories. In other words, each document is given multiple scores derived from its popularity and from its similarity with each ODP category. This information comes into play when a query is submitted, where the query's topic is used to identify which category score for a document will be used when ranking the documents. This work was extended by (Qiu and Cho, 2006) where

the authors incorporated individual user interests into the process. This was done by using clickthrough information to construct user models that are based on concepts from ODP. The evidence from these user models (i.e. the conceptual interests) was then factored into the equation, together with evidence from the deduced query's topic, to select the most appropriate document score to use in the ranking process.

Query Adaptation and Result Adaptation:

Some systems employ both query adaptation and result adaptation. For example, in the UCAIR system (User Centred Adaptive Information Retrieval) presented in (Shen et al., 2005), the authors argue that the two main aspects of personalised search are: the user's interests and the search context (i.e. query disambiguation). The authors focus on modelling the user's short-term interests, in an approach called eager implicit feedback. In this approach, the current query's context is deduced using evidence from the immediate previous query (within the search session) and the results clicked for it. To determine if two successive queries are related, the system performs two searches; one with the previous query and one with the current query. The retrieved result lists for the two queries (50 results for each) are then compared to each other by checking how many terms are common between the titles and snippets of the two lists. If the two queries are related (based on a textual similarity threshold), the current query is expanded using terms from the short-term user model created for the previous query. Following the submission of the adapted query to the retrieval component, the retrieved result list is re-ranked based on the user model. The user model is updated in a live manner whenever the user clicks on a result from the displayed list. Based on the updated model, further result re-ranking takes place if the user clicks on the next link (i.e. live re-ranking is performed when the user requests to see the next results page).

A rather different approach for query and result adaptation was presented in (Liu et al., 2004). In one of the proposed systems, a vector-based user model of conceptual terms was maintained. The conceptual interests were based on Google Directory[1], which is a Web taxonomy that is based on ODP. Google Directory provides a facility to specify the category to which a query is to be submitted. Query adaptation was not performed by expanding the query terms, but rather by specifying the category of the query (e.g. Health, Arts, etc.). In other words, the system attempts to infer the concepts related to the submitted query and then use these concepts to provide context information to the retrieval system when submitting the query. An automatic and a semi-automatic approach were used to deduce candidate conceptual categories to which the query may belong. In the automatic approach the query terms were mapped into candidate

[1] http://www.google.com/dirhp

concepts and then these concepts were scored against the concepts in the user model. The top N categories (up to three) related to highly scored concepts are then identified and specified when the query is submitted. In the semi-automatic approach, an additional step takes place, which is that candidate categories are shown to the user (three at a time). The user is then allowed to select the appropriate categories related to the query. After the categories are identified, the query is actually submitted multiple times for retrieval; once without specifying any categories, and one time for each of the identified categories. This leads to the retrieval of multiple result lists. The system then performs result adaptation by ranking and merging the results into a single list. A weighted voting based algorithm was used, where results that appeared on more than one list were favoured.

The advantage of this technique, besides catering for the user's long-term interests, is that it also accounts for the possibility of ad-hoc queries. This is because the fact that multiple result lists are sought by the system, one of which is based on the non-adapted version of the query, allows for some diversification in the kind of results presented to the user. This is opposed to other systems where only one result list is sought based on an inferred user interest; an approach where if the system's guess about the query's topic is wrong, the result list might be dominated by results that are irrelevant to the user's current information need. This approach is related to an approach in the IR field known as result diversification, where retrieval systems deliberately diversify the set of results presented to the user, especially on the first page of results (Santos et al., 2010, Minack et al., 2009, Gollapudi and Sharma, 2009). The rationale behind this approach is to guarantee that users with random or different intents will find at least one relevant document to their information need in the result list. Furthermore, this approach encourages users with explorative behaviour to learn more about diverse topics, which they may have not learnt about otherwise.

2.4.3 Summary and Discussion of Personalisation Implementation and Execution

The previous section provided an analysis of the personalisation approaches exhibited in several existing systems. The analysis focused on the core process of executing personalisation using different techniques for query adaptation and result adaptation. Table 5 presents a summary of the analysis, along with some example systems.

Table 5: summary of personalisation approaches

Personalisation Approach	Personalisation Scope	System Type	Example Publications
Query Adaptation (query expansion using terms from user model)	Individualised	Web search	Chirita et al. 2007, Psarras and Jose 2006
Query Adaptation (query expansion using terms from query logs or generated thesaurus)	Aggregate-level	Web search	Yin et al. 2009, Cui et al. 2003
Query Adaptation (query suggestions using similar queries from query logs in other languages)	Aggregate-level	Cross-language Web search	Gao et al. 2007, Ambati and Uppuluri 2006
Result Adaptation (result re-ranking)	Individualised	Web search	Vallet et al. 2010, Noll and Meinel 2007, Speretta and Gauch 2005, Stamou and Ntoulas 2009, Teevan et al. 2005, Ruvini 2003, Pretschner and Gauch 1999
Result Adaptation (result re-ranking)	Individualised	Search and recommendations on computer science literature	Micarelli and Sciarrone 2004
Result Adaptation (result filtering and re-ranking)	Individualised	News	Katakis et al. 2009, Chen and Sycara 1998
Result Adaptation (result re-ranking)	Community-based	Web search	Teevan et al. 2009, Sugiyama et al. 2004
Result Adaptation (result re-ranking)	Aggregate-level	Web search	Smyth and Balfe 2006, Sun et al. 2005
Result Adaptation (result scoring)	Individualised	Web search	Qiu and Cho 2006
Result Adaptation (result scoring)	Individualised	Web search and document recommendations	Stefani and Strapparava 1999
Result Adaptation ((1)result scoring & (2)result re-ranking)	Aggregate-level	Web search	Agichtein et al. 2006
Result Adaptation (re-structuring and tailoring content of results into a hypertext presentation)	Individualised	eLearning (search on domain-specific corpora for education purpose)	Steichen et al. 2009
Query & Result Adaptation	Individualised	Web search	Shen et al. 2005, Liu et al. 2004, Pitkow et al. 2002
Query & Result Adaptation	Individualised	Customer support (search on domain-specific corpora for technical support)	Steichen et al. 2011

The analysis above shows that individualised user models in PIR systems were mostly used for result adaptation compared to relatively much fewer systems where individualised models were used for query adaptation. Personalised query adaptation was often based on aggregate usage information as exhibited in search logs. A broader consideration of IR literature reveals that the majority of studies which investigated query expansion were based on approaches that are not user-focused; mainly PRF. **The study carried out for this thesis furthers PIR research in the direction of investigating the effectiveness of query adaptation based on individualised user models. Moreover, a contribution of this study is extending this investigation to MIR systems.**

As discussed earlier, the selective query expansion technique, used in conjunction with PRF, has shown success in detecting, and therefore avoiding, cases where query expansion would harm retrieval effectiveness (Leveling and Jones, 2010a, Cao et al., 2008). However, this technique is not personalised as it only depends on information drawn from the document corpus. **A contribution of the study carried out for this thesis is performing personalised selective query expansion by basing the dynamic decision of whether or not to expand a query on evidence from the user model itself.**

The authors in (Cui et al., 2003) compared query expansion based on search logs to query expansion based on PRF and showed that the former leads to higher retrieval effectiveness. A mixed approach of the two was used in (Shen et al., 2005) and showed improvements over a baseline that was wrapped around Google search. The system inferred the user's current information need in a search session and expanded the query based on the inferred short-term interest. This was done by examining the snippets of the top result retrieved in an initial retrieval round using the given query (as in typical PRF), as well as examining the immediately preceding query in the same session and snippets of the clicked results associated with it (recent search history). **The PMIR framework reported in this thesis builds on this successful approach and further extends it into the area of multilingual search.**

Although a number of systems used both query adaptation and result adaptation, no study attempted to compare the improvements achieved by the two approaches or investigate which one of them contributes more to the improvement of retrieval effectiveness. **Part of the research carried out in this study involved evaluating retrieval effectiveness based on each approach individually as well as based on the combination of the two.**

2.5 Evaluation Approaches

2.5.1 Overview

This section discusses the various approaches to evaluating PIR systems. Although evaluation is not literally a stage in the personalisation process itself, it was nonetheless important to include it in the review. This is because it shows the effectiveness and the efficacy of the different personalisation approaches and techniques discussed in the previous sections. Moreover, as the study reported in this thesis is associated with the areas of IR, Personalisation, and Adaptive Hypermedia, surveying existing evaluation approaches in these areas helps in selecting the most appropriate approaches to evaluating the PMIR framework proposed in this thesis.

Four criteria are used to derive the discussion in this section: the aspect of evaluation targeted by the system, the evaluation metric or instrument used for evaluation, the datasets used in the experiments, and the experimental setting for evaluation. An overview of these four classification criteria is given below.

- **Aspect of evaluation:** the first criterion in this section is concerned with *what* is being evaluated in the system. Two aspects of a PIR system are subject to evaluation:
    1. *System performance*, which is usually concerned with measuring retrieval effectiveness (Yin et al., 2009, Chirita et al., 2007, Smyth and Balfe, 2006, Teevan et al., 2005). The advantage of evaluating PIR systems in terms of retrieval effectiveness is that it stands out as a well-defined quantitative comparison across different systems. However, a concern regarding this kind of evaluation is that it is more system-focused than user-focused.
    2. *Usability*, which is concerned with the user's perception of, and satisfaction with, the system. As personalisation is concerned with adapting to the user's needs, the benefit of evaluating usability of a personalised system is that it pays attention to these needs and measures the degree of user satisfaction with respect to the adaptive service. A weak point in this type of evaluation, however, is that it is hard to standardise across different systems and that it is subject to user bias.
- **Evaluation metric or instrument:** the second criterion is concerned with the quantitative and qualitative metrics or instruments used for evaluation. Respectively following on the aspects of evaluation discussed in the previous criterion, the metrics are as follows:

1. *Retrieval effectiveness,* which 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):
    1. Precision: the number of retrieved relevant documents over the total number of retrieved documents.
    2. Recall: the number of relevant documents that are retrieved over the total number of known relevant documents in the document collection.
    3. Precision at K: the fraction of retrieved relevant documents within the top K retrieved documents.
    4. Recall at K: the fraction of retrieved relevant documents within the top K documents over the total number of relevant documents in the document collection.
    5. Mean Average Precision (MAP): 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.
    6. Normalised Discounted Cumulative Gain (NDCG): 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 is usually used in the evaluation of result re-ranking as it gives higher scores for more relevant documents being ranked higher in the ranked list of results.
    7. R-precision: measures precision with respect to a given number of documents that are known to be relevant:
    8. 11-point Precision: the precision of retrieved results at 11 fixed values of recall.
    9. F-Measure: the weighted harmonic mean of precision and recall.
    10. Break-even Point: determines the point at which precision equals recall.
2. *Usability,* which can be qualitatively evaluated using usability questionnaires (Brooke, 1996, Harper et al., 1997) or quantitatively evaluated by measuring the user's performance in fulfilling certain tasks using the system, for example by keeping track of the time and number of actions needed to complete the task.

- **Datasets:** the third criterion is concerned with the datasets used in the experiments. In PIR, two kinds of datasets are used: 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[1], CLEF[2], and NTCIR[3], which are widely used in the IR community. These collections, together with a set of manually selected information needs[4], 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; experiments can also be 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 usually not over-fitted on the domain or characteristics of a specific document collection. However, a concern associated with this approach is that it is hard to perform "apples-to-apples" comparisons between the results of different studies. Search logs, as discussed earlier, 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.

- **Experimental setting:** the fourth criterion in this section is concerned with the experimental setup put in place for evaluation. Some studies conduct experiments in a controlled setting that involves a relatively small number of users and tasks (Stamou and Ntoulas, 2009, Steichen et al., 2009, Speretta and Gauch, 2005). The advantage of this setting is that it allows the establishing of control groups and conducting a richer evaluation of usability aspects. On the other hand, other studies base their evaluation on a large amount of data drawn from a realistic setting (e.g. well-known Web search engines) (Yin et al., 2009, Gao et al., 2007, Agichtein et al., 2006a). Large-scale experimental settings contribute towards more conclusive results. However, it is increasingly becoming difficult for the academic research community to gain access to search logs of major search engines (e.g. Google, Bing, Yahoo, etc.) because major companies are reluctant to release these logs and they use it for their own research and development.

The following section presents a collective review of the evaluation carried out by several systems in the literature.

[1] TREC: Text REtrieval Conference: http://trec.nist.gov/
[2] CLEF: formerly known as the Cross-Language Evaluation Forum: http://www.clef-campaign.org/
[3] NTCIR: NII Test Collection for IR Systems: http://research.nii.ac.jp/ntcir
[4] Test queries that are associated with each collection

2.5.2 Review

The discussion in this section starts with systems where experiments targeted the evaluation of the system's performance in terms of retrieval effectiveness (i.e. IR-style evaluation of retrieval precision or recall). The discussion then moves on to systems where experiments targeted the evaluation of other usability aspects of the system (i.e. AH-style evaluation, which is more user-focused).

In the area of IR research, a common quantitative evaluation approach is to compare the effectiveness of a proposed search system to a baseline search system. For example, for the I-SPY system (Smyth and Balfe, 2006), the authors evaluated the precision and recall of their experimental PIR system against a non-personalised version of the system. The underlying retrieval component in both systems comprised a meta-search engine that performed search over open Web corpora by collating results from several well-known search engines. The experiments were conducted in an in-lab experimental setting that involved 92 users. The users were divided into two groups; one for training (45 users) and one for testing (47 users). In the first group (i.e. the training group), each user was assigned 25 information needs to satisfy using a Web search interface (live meta search engine). The users were free to formulate any number of queries that described the given information need. The interactions of the first group with the baseline Web search system were logged and used for training I-SPY. The logs contained the submitted queries and the clicked results. The logged information was then used in two ways: (1) to create ground-truth relevance judgements, where the relevance of clicked documents with respect to the queries was manually assessed on a binary scale (i.e. relevant vs. irrelevant); and (2) to generate the hit matrix (i.e. to train the personalised system based on click frequencies on result documents).

Users in the second group (the test group) used the personalised Web search system and were also given 25 information needs to fulfil using any number of queries. It is to be noted here that the approach of dividing the users into two groups was applicable because the baseline system did not perform personalisation in an individualised manner, but rather in an aggregate manner based on general usage history. Thus, it was not a must that the same group of users be subjected to both systems. Several IR metrics were used for retrieval evaluation based on the ground-truth relevance judgements generated earlier. The results show that the I-SPY system achieved significant improvements between 117% and 266% over the baseline system using the Precision at K metric, where K varied between 5 and 30. The results also show improvements between 138% and 280% using the Recall at K metric with the same range of values for K.

Moreover, the F-measure metric was also used to evaluate the personalised system where the results showed improvements up to 380% over the baseline system for K = 30.

It is worth mentioning here that majority of studies in the PIR field report precision or recall improvements between 10% and 50% over a baseline; it is not very common to achieve improvements over 100% such as in the I-SPY study. Besides the possibility that their proposed system was a very successful one, another viable possibility is that the baseline system used in the comparisons was a weak one. A major challenge that faces researchers in the PIR field is user expectation; it is not easy for researchers to achieve improvements over major search engines, which are considered to be very good by the users and stand out, to many users, as the standard of how a search service should operate and deliver.

In (Teevan et al., 2005) an in-lab experimental setting was also used to compare a personalised system to a baseline system. The retrieval component was wrapped around MSN Search. Relevance judgements were performed in a non-binary manner, where documents were judged on a three-level scale: highly relevant, relevant, or not relevant. The NDCG metric was used to evaluate retrieval effectiveness. The experimental results showed that their personalised system significantly outperformed the baseline system with a 24% improvement.

As discussed earlier, systems which implicitly infer users' search interests can harvest terms from the queries that the users submitted, the documents that they clicked on, or the snippets of the clicked documents. With respect to these different sources, an interesting study was reported in (Speretta and Gauch, 2005) where a system in which terms are extracted from queries was compared to a system in which terms are extracted from snippets of clicked documents. The retrieval effectiveness of the experimental systems was evaluated against a non-personalised system. All systems used a retrieval component that was wrapped around Google.

The experiments involved six users who used the baseline system for their own daily searches (i.e. users' own information needs) over a period of six months. The baseline system randomised the top ten Google results before displaying them to the user. All the users' interactions with the system were logged. From the logs, 47 queries per user were extracted, where 40 queries were used for training the personalised systems (i.e. for constructing the user model either from the text of the queries or the text of the snippets), 5 queries were used for testing a number of parameters of the system (fine tuning), and 2 queries per user were used for validating the system. A notable difference between this study and other studies is that relevance judgments were not based on manual assessments. Rather, an implicit approach was

used where documents that were clicked by users while using the baseline system were deemed as relevant. However, this approach only produced a very small number of judged documents with respect to test queries. A simple rank scoring measure was used to evaluate retrieval where each system was evaluated according to the rank it assigned to the relevant documents of the query (i.e. in which position in the list the system placed the few documents that were implicitly judged as relevant). The results showed that both proposed systems were equally capable of improving retrieval effectiveness over the baseline system, with a very slightly higher improvement (statistically significant) for the snippet-based system (34%) compared to the query-based system (33%).

A number of studies, especially ones that were carried out by research teams who are affiliated to major search engine companies, conducted their experiments on large-scale datasets. This is compared to the relatively smaller datasets that are generated by in-lab experimental settings. For example, in (Agichtein et al., 2006a) a realistic experimental setting was arranged, where a dataset of usage data was obtained from a well-known search engine[1]. The dataset comprised search logs recorded for user interactions with the search engine over a period of eight weeks. The dataset contained over 1.2 million unique queries and over 12 million user interactions (post-search actions, including clicking on results). A random sample of 3,000 queries was drawn from the dataset and was used for the experiments. For each of the queries, 30 result documents on average were manually judged for relevance.

The authors noted that one of the characteristics of a realistic experimental setting is that implicit feedback can be noisy (e.g. inconsistent or incomplete). Nevertheless, they argued that this characteristic actually counts towards the reliability of the experimental results. Several personalised systems, in addition to the baseline system, were tested against each other. Personalisation was performed on an aggregate usage level where the systems made use of part or the entire evidence of implicit feedback. The systems mainly involved two personalisation approaches: result scoring and result re-ranking. Three metrics were used for retrieval evaluation: Precision at K, NDCG, and MAP. The experiments showed that: (1) making use of implicit feedback information is useful in realistic Web search environments, despite the existence of noise in the recorded logs; (2) result scoring, where implicit feedback features are incorporated into one scoring function together with other existing scoring features, is more effective than result re-ranking; (3) using several implicit feedback features leads to better results than just using clickthrough features (i.e. it is recommended to make use of additional pieces of evidence of implicit feedback such as dwell time on a page).

[1] The study was conducted at Microsoft Research, but the name of the underlying search engine was not specified.

The experiments carried out by (Gao et al., 2007) were also conducted in a large-scale setting. As discussed earlier, the authors proposed a Cross-Lingual Query Suggestion (CLQS) system. Given a source query in a certain language, the system obtained related queries from other languages by analysing multilingual search logs. The proposed CLQS method was intended to be used as a method that combines query expansion with query translation instead of the typical use of a translation component in CLIR. The experiments were conducted on large datasets of English and French search logs. The first dataset included 7 million unique English queries, obtained from MSN Search logs over a period of one month. The second dataset included 5000 randomly selected French queries out of 3 million queries from a French query log. The TREC-6 CLIR document collection and its 25 information needs were used in the experiments. The cross-lingual retrieval effectiveness of the proposed CLQS system was evaluated using the 11-point Precision metric against three systems: a monolingual system, a system that used Google French to English machine translation, and a dictionary-based query translation system using co-occurrence statistics for translation disambiguation. The proposed CLQS system achieved 7.4% improvement over the machine-translation-based system and 25% improvement over the dictionary-based system. It was able to achieve 88% of the monolingual system performance. A rather similar setting was also used in (Yin et al., 2009) where the experiments were conducted on a dataset of search logs obtained from Microsoft Live Search over a period of ten months. The dataset contained 12 million unique queries.

Evaluation in the area of Adaptive Hypermedia (AH), especially in the educational domain, has often focused on the efficacy of the adaptive service within the given domain (Conlan and Wade, 2004, De Bra et al., 2003, Brusilovsky and Peylo, 2003). This type of evaluation reflects the two-fold challenge of evaluating adaptive systems: how to uniformly test a system which changes in response to the user and how to evaluate a complex user experience with an unbiased measure. This gives rise to the use of measures such as task time completion and user satisfaction as a basis for testing the adaptive experience.

For example, the authors in (Conlan and Wade, 2004) proposed an adaptive eLearning system based on the content of an undergraduate-level SQL (Structured Query Language) online course. The course was divided into two parts, a database theory part (given as face-to-face lectures) and a practical part (online) concerning the learning of SQL. Only the SQL part was presented via an adaptive eLearning course and was evaluated in large-scale experiments. The experiments involved a total of over 500 students, spanning a period of four years. The experiments aimed at evaluating the effectiveness of the course provided by the adaptive system by examining the students' performance over a number of years in exams specifically related to SQL topics. This included exam scores over the period of the evaluation (four years)

and also the two preceding academic years when an online non-adaptive version of the course was used. Evaluation was concerned with comparing how the students performed using the non-adaptive online course (before the introduction of the adaptive one) to how they performed using the proposed adaptive system. The results of the experiments demonstrated the success of the proposed adaptive system where an average of 13% increase in students' exam scores was reported for the adaptive system over the non-adaptive one. Furthermore, analysis of differences in student capabilities across the years was performed to ensure no natural bias between years.

The authors in (Steichen et al., 2009) carried out an assessment of the knowledge gain of the students in a domain-specific eLearning environment. The knowledge gain was assessed by comparing the students' initial knowledge, measured in a pre-test, with their answers to task-based questions in the adaptive system. The experimental setting involved 12 students who were asked to complete 3 learning tasks that were randomly selected from a pool of 6 tasks. The knowledge gain was calculated by scoring the students' answers in the pre-test on a scale from 0 to 5 (where 0 indicated that the student had no prior knowledge of the task area, and 5 indicated that the student had the knowledge needed to carry out the task) and by assessing the students' answers to the given tasks on the same scale (where 0 represented complete failure to solve the task and 5 represented complete success). The average knowledge gain of the 12 students using the system was 4.25, which reflected the educational impact of the proposed adaptive system. Moreover, the students were also asked to fill questionnaires to evaluate the usefulness and the usability of the system. The results suggested that students were satisfied with the relevance of the presented content to their information needs and that they liked the presentation of results in the form of adapted hypertext presentations (dynamic composition of results and eLearning content).

Task-based evaluation was carried out in (Pitkow et al., 2002) where the system recorded the time and number of actions that the users needed in order to successfully complete a number of given search tasks. The experiments were carried out in an in-lab setting that involved 48 users using two systems: the experimental personalised search system (which was wrapped around Google) and any of the following well-known search engines: AOL[1], Excite[2], Yahoo, or Google. Each user was given 12 search tasks and a maximum time of 3 minutes to complete it. The results showed that the proposed personalised system enabled users to complete their tasks in less time and a smaller number of actions compared to the use of one of the search engines. It should be noted that the proposed system offered a rich user interface that comprised a number

[1] http://www.aol.com/
[2] http://www.excite.com/

of special additional features that are not present in other search engines. Thus, the authors argue that a bias towards their system in the experimental results may be observed because some of the tasks were tailored to make use of those special features which enabled users to use them and finish their tasks faster and with fewer actions. A notable drawback in the evaluation process was that, due to experimental limitations, a default user model (i.e. the same user model) was used for all the users. The default user model contained information mined from browsing history of documents that are related to the search tasks. The users were given the chance to view the content of the user model prior to the experiment. Such use of a default user model is not a common approach in PIR studies and may render the experimental results doubtful with regards to the efficacy of the personalised service.

2.5.3 Summary and Discussion of Evaluation Approaches

Several evaluation approaches were discussed in the previous section. Table 6 presents a brief summary of these approaches and gives some example publications of each approach.

Table 6: summary of evaluation techniques

Scope of Evaluation	Evaluation Metric & Instrument	Datasets	Experimental Setting	Example Publications
System Performance (retrieval effectiveness)	Quantitative (P@K, Recall@K, F-measure, Break-even point, NDCG, R-precision)	Documents: open Web corpora. Logs: in-lab generated logs.	In-lab setting (6 to 47 users)	Smyth and Balfe 2006, Teevan et al. 2005, Speretta and Gauch 2005
System Performance (retrieval effectiveness)	Quantitative (MAP, 11-Point Precision)	Documents: TREC collections. Logs: search engine query logs.	Large-scale setting (large number of live user interactions with a Web search engine: 3 to 12 million unique queries)	Yin et al. 2009, Gao et al. 2007
System Performance (retrieval effectiveness)	Quantitative (P@K, NDCG, MAP)	Documents: open Web corpora. Logs: search engine logs.	Large-scale setting (large number of live user interactions with a Web search engine: 1.2 million queries)	Agichtein et al. 2006
User Evaluation (task-based)	Quantitative (time and number of actions needed to complete search tasks)	Documents: open Web corpora. Logs: in-lab generated logs.	In-lab setting (48 users)	Pitkow et al. 2002

User Evaluation and System Usability	Quantitative & Qualitative (task score & usability questionnaires)	Corpora: domain-specific corpora, harvested from the Web	In-lab setting (12 users)	Steichen et al. 2009
User Evaluation and System Usability	Quantitative & Qualitative (exam scores & usability questionnaires)	Corpora: domain-specific eLearning corpus	Large-scale setting (500 users)	Conlan and Wade 2004

A key challenge that faces academic researchers in the field of PIR is obtaining realistic search logs that can be used to infer users' behavioural patterns and search interests. Major search engines do not prefer to release their search logs to the public or even to the academic community. This may be attributed to two reasons: privacy concerns and competitive business or technological advantage. Thus, the alternative for researchers becomes in-lab-style experiments.

Although an in-lab experiment would not yield a relatively large dataset of search logs, it has a number of advantages (Borlund, 2000, Teevan et al., 2005, Speretta and Gauch, 2005). Among these advantages are: (1) more focused user studies and usability evaluations can be conducted by providing questionnaires to the users or by directly interviewing them; and (2) the experiments can be repeated with different settings using the same test group of users, and therefore comparisons can be conducted between different experimental runs.

The evaluation carried out in this thesis uses an in-lab experimental setting and employs proven quantitative and qualitative evaluation techniques from both areas: IR and AH. On the IR side, the evaluation comprises measuring the effectiveness of the proposed personalised systems against a baseline system and against each other (individually and in combination). **A distinctive advantage of the relevance judgments carried out in the experiments in this thesis is that the users themselves are the ones who judged the results.** This truly captures the notion of relevance as the judgments reflect the opinion of the users themselves (i.e. *personal* judgments as opposed to assigning other users to judge the relevance of the results on behalf of the original users –an approach that is commonly used in IR studies). On the AH side, the evaluation comprises questionnaires for evaluating the usability of the system and also the user's perception of specific system features (e.g. presenting interleaved search results from multiple languages in the result list, quality of instant translation of web pages, etc.).

2.6 Summary and Conclusion of the Survey Findings

This chapter provided a critical review and analysis of state-of-the-art approaches in the field of PIR. The analysis was carried out over four stages: (1) information gathering, which was concerned with approaches to collecting information about system users; (2) information representation, which focused on different approaches to maintaining and modelling usage and user information; (3) personalisation implementation and execution, which presented an in-depth analysis of approaches to search personalisation; and (4) system evaluation, which provided a review of the experimental settings and evaluation techniques involved in the evaluation of PIR systems.

Furthermore, the chapter presented a classification of PIR systems into three categories according to the scope of personalisation addressed, namely: individualised personalisation, community-based personalisation, and aggregate-level personalisation. The chapter also presented a classification of query adaptation techniques from a personalisation perspective. This classification featured two attributes: (1) user-focused vs. non-user-focused techniques; and (2) implicit vs. explicit techniques.

In conclusion, the state-of-the-art survey shows that the majority of existing studies investigated personalisation in monolingual search, and that relatively fewer studies extended to multilingual search. Furthermore, with respect to the use of an individualised user model for PIR, it should be noted that no studies attempted to investigate the construction of user models that would specifically represent and cater for the needs of users in MIR. **The research carried out for this thesis addresses this gap and shows that MIR systems can benefit from the use of individualised user models. This is demonstrated as part of the proposed multilingual approach to search personalisation which comprises multilingual user modelling, multilingual query adaptation, and multilingual result adaptation.**

The survey also shows that a few studies in PIR literature addressed the challenge of personalised query adaptation based on information from the user model. More specifically, there is an exhibited gap in MIR literature with respect to performing query expansion based on terms obtained from the user's search interests. The key challenge facing this kind of research is how to determine which terms in the user model are most related to a given query so that they can be selected for expansion**. The research carried out for this thesis investigates query expansion and selective query expansion in a user-centred manner where the information in the user model is used for making the dynamic decision of whether or not**

expand the query and for obtaining the most relevant terms to be used in expanding the query.

A higher tendency is noted in PIR research towards evaluating systems in a quantitative manner compared to evaluating them in a qualitative manner. This tendency can be attributed to the wide usage of precision-based evaluation metrics in the field of IR where retrieval effectiveness is the focus of evaluation and where the standardised IR metrics allow bench-mark testing across many systems. Since PIR can be recognised as a research area where there is a hybrid fusion of techniques from both the IR and AH areas, then it is necessary for PIR studies to also pay attention to the user side in the evaluation process. **Therefore, the evaluation conducted for this thesis focuses on both quantitative and qualitative aspects of the proposed PMIR framework.**

Finally, the thorough analysis carried out in this state-of-the-art survey helped in **gaining an insight into the key elements of the search personalisation process and served as a basis for designing the components and workflow of the PMIR framework proposed in this thesis**. Thus, the survey served in addressing the following challenge of the thesis (Section 1.2): ***Challenge #1:*** *What are the key components of the search personalisation process and how can the process accommodate a personalised multilingual search service?*

The design presented in Chapter 3 also contributes to challenge#1, especially to the part concerning extending the process to accommodate personalised multilingual search.

Chapter 3: Design

This chapter describes the design of the study reported in this thesis. The chapter builds on the lessons learnt from the state-of-the-art survey presented in the previous chapter and proposes solutions to address the challenges outlined in it. First, the chapter discusses the design of exploratory investigations concerning MIR and users' search behaviour. Second, it discusses design considerations concerning user models that reflect the aspect of multilinguality and search personalisation algorithms that cater for multilinguality. Finally, it proposes a framework for the delivery and evaluation of PMIR which unites all the elements of the solution. This involves a discussion of the PMIR process and workflow, and a discussion of design considerations for the framework.

3.1 Design Overview

As shown in the state-of-the-art survey, there is a lack of studies which investigate personalised search in a **multilingual environment**. This study aims to address this gap in the literature.

It is important at this point to explain the notion of a multilingual environment, and the implications of factoring multilinguality into the search personalisation process. Since both the users and the content may be multilingual, this leads to the existence of three scenarios that pertain to the multilingual dimension:

1. **A multilingual user who interacts with multiple search engines separately:** for example, in the case where a user speaks English and French and so s/he sometimes uses an English search engine and at other times s/he uses a French search engine. In this case, the user's multilingual search interests (i.e. the user's interests across languages) will be reflected in the queries that s/he submits to each search engine and the results the results that s/he browses. The implication of this on personalisation is that each search engine may only have a partial view of the user's overall interests, since the search engines are separate. It is worth highlighting here that this scenario is possibly the most common of the three scenarios; in today's world, a multilingual user interacts with various search systems (e.g. Web search, library search, online-shopping search, etc.) in different languages. Another example is a Chinese user residing in an English-speaking country who uses *Baidu*[1] when searching for things in Chinese and uses *Google* when searching for things in English.

[1] http://www.baidu.com/

2. **A multilingual user who interacts with a multilingual search engine:** a user who is able to speak/understand multiple languages uses a search engine that returns mixed search results from those languages. In this case, the search engine can maintain a complete view of the user's multilingual search interests. This will allow the search engine to recognise the multiple personas of the user and devise personalisation approaches that cater for these personas.
3. **A monolingual user who interacts with a multilingual search engine:** a user who only speaks one language uses a search engine that returns mixed results from multiple languages (applying result translation where necessary). Even though the user in this case is monolingual, s/he may still exhibit interests that are distributed across languages due to browsing results that came from different languages. Therefore, from a personalisation perspective, this scenario is similar to the second scenario. Moreover, this scenario may also be extended to the case of a multilingual user who receives multilingual results coming from languages that s/he does not understand.

The research reported in this thesis focuses on the second and third scenarios and shows that:

- Users have multiple behavioural personas when seeking information on the Web, dependent on the combination of their language capabilities, and the availability and variety of content in various languages.
- Users, whether monolingual or multilingual, may choose to browse search results originating from certain languages depending on the type of information sought.
- The user modelling approach should reflect this kind of language influence on the user's interests.

Moreover, it is worth noting here that the user modelling approaches proposed in this study are inspired from the first scenario, and could actually cater for it in the following special cases:

- When a company provides multiple facets of its search engine in different languages (e.g. *google.ie*, *google.fr*, *google.de*). In this case, the search engine can maintain the user's interests from the multiple facets in a centralised repository.
- When a third-party service offers personalisation across multiple websites (cross-site personalisation using cookies, browser plug-ins, etc.) (Koidl et al., 2011). In this case, the third-party service can track the user's interactions across multiple websites and maintain a centralised repository of the user's multilingual interest.

A key point that is important to emphasise here is that there exists a spectrum of PMIR use-cases which differ in language emphasis. On one side of the spectrum there is the notion of *multilingual approaches to personalised search* which is concerned with catering for a multilingual user who uses multiple search systems, and may interact with each of those systems in one or more languages. On the other side there is the notion of *personalisation approaches to multilingual search* which is concerned with catering for users (both monolingual and multilingual) who use a search system that provides results from multiple languages. **The objective of the PMIR framework proposed in this thesis is to create a platform that facilitates the evaluation of a broad range of these approaches**.

In light of the aforementioned discussions, the following paragraphs provide an overview of the remaining sections of this chapter.

As a preliminary step, prior to developing personalisation techniques, this study investigates two important features concerning the multilingual dimension in user interactions with search systems. First, the study investigates the usefulness of multilingual search to users in realistic use-cases. Second, the study examines the way users from different linguistic backgrounds behave when using multilingual search systems. This helps to gain a better understanding of the process and elicit features that would guide the design of the user model and the personalisation techniques. Sections 3.2 and 3.3 discuss the design principles concerning the investigation of those two aspects.

With regards to user modelling for personalised search, the state-of-the-art survey discussed the various ways of representing user models in terms of structure and content. However, the introduction of multilingual aspects to search personalisation has two key consequences on the design of user models. First, this study takes into account that terms which represent the user's search interests exist in multiple languages (as opposed to a single language in the case of monolingual search) and therefore investigates how to represent them in a suitable manner. Second, this study takes into consideration that additional attributes should be added to the user model, such as the user's native language and preferred language. The details of designing multilingual user models for this study are discussed in Section 3.4.

The survey discussed techniques of search personalisation and showed that adaptation can take place either by altering the query and/or the results. As part of the multilingual approach to search personalisation proposed in this thesis, the study develops these techniques in a way that suits multilingual search, in accordance with the proposed user model. Furthermore, with respect to query adaptation in specific, the survey discussed how query adaptation can

sometimes degrade retrieval effectiveness and showed how content-based selective query adaptation techniques are used to inform a system's decision of whether or not to adapt a query. As part of the *individualised personalisation*[1] approach proposed in this thesis, the study investigates how to individualise the process of selective query adaptation so that adaptation decisions are taken based on the information available in the user model itself and not the content. Section 3.5 discusses the details of the design of the personalisation algorithms proposed in this study.

The review and classification carried out in the survey showed the stages and components of the search personalisation process. The main contribution of this thesis is proposing a framework that orchestrates the elements of multilingual search personalisation in a way that facilitates delivering the service and evaluating the elements, both in isolation and in combination with each other. Section 3.6 presents the design of the framework and discusses the functional and non-function requirements that should be taken into consideration when implementing the framework.

3.2 Investigating the Usefulness of MIR in Realistic Scenarios

As discussed in the literature review, several studies were concerned with how to improve retrieval effectiveness in MIR. However, this was only approached from an IR perspective. In other words, those studies disregarded two important design issues associated with MIR evaluation:

1. Whether or not there is a need for MIR in practice (i.e. in application areas, such as in the industry). For example, the studies did not pose questions like: *do enterprises perceive a need for providing online multilingual search facilities to their customers? Is it possible to achieve the same improvements in retrieval effectiveness if the experiments were conducted on enterprise or vendor-specific content?*
2. Whether or not users would accept the notion of getting search results from multiple languages in response to their queries. For example, the studies did not pose questions like: *What is the user's perception of browsing search results that come from different languages? Is the user's information need satisfied when browsing search results that were machine-translated to their native/preferred language?*

[1] As discussed in Chapter 2, the term *individualised personalisation* is used in this thesis to refer to the notion of adapting a service to a specific individual, hence the existence of a user model that represents this individual and that is used as the basis for the adaptation algorithms.

Therefore, it is important for PMIR studies to bridge the gap between academia on one hand and industry and users on the other hand. This can be done by investigating MIR in realistic scenarios. In order to address the first design issue stated above, this study perceived a need to investigate the usefulness of MIR in an industry case-study. This involved two aspects: (1) confirming that there is a need in the industry for providing multilingual search services; and (2) quantitatively evaluating the retrieval effectiveness of MIR based on enterprise content. As for the second design issue discussed above, this study perceived the need to make use of qualitative evaluation techniques to evaluate the usefulness of multilingual search to the users. This involved administering a questionnaire to users of a multilingual search system to ascertain their perception of the various features of the service (e.g. perception of interleaved results from multiple languages presented in a single list, perception of instant-translation of search results, etc.).

The abovementioned design issues provided guidance to a set of experiments reported in the Evaluation Chapter of this thesis (Sections 5.1 and 5.5). The experiment reported in Section 5.1 stands out as an industry case study that demonstrates how MIR can be useful in an online technical support scenario. The experiment showed that enterprises can provide relevant search results to their customer queries in languages that are different from the query's language in the case of absence of content in the query's language. The case study was concerned with multilingual users (i.e. users who could comprehend content in both the query's language and the target language).

The questionnaire reported in Section 5.5 was administered to users after they interacted with a multilingual Web search system. This included both multilingual users and monolingual users (i.e. the system translated search results to the user's preferred language where necessary). The users' responses to the questionnaire showed that they found that the provisioning of search results from multiple languages blended in a single list was useful in satisfying their information needs. Moreover, the users found that the quality of instant machine translation, even though not perfect, was good enough to convey the information in the documents they viewed. The outcome of the questionnaire demonstrated the efficacy of multilingual search and its perceived usefulness to both multilingual and monolingual Web search users.

3.3 Exploring Users' Search Behaviour

After verifying the efficacy of multilingual search, the next logical step was for the study to explore the personalisation side of the PMIR process. Before developing personalisation algorithms, it was necessary to gain insight into how users from different linguistic backgrounds behave in search. The outcome of this step served to inform the design decisions of the user model and the personalisation algorithms (e.g. *what kind of information needs to be in the user model? How should the adaptation algorithms be developed to suit PMIR?*).

The method used to investigate users' search behaviour in this study comprised analysing search logs**.** This involved statistical analysis and investigation of behavioural patterns in a dataset of multilingual search logs. An import issue that was considered before using this method was the availability of (or lack thereof) such dataset and whether it was accessible to the research community. The nature of information available in the dataset was also taken into consideration (e.g. queries, clickthrough information, and languages of queries). The investigation reported in Section 5.2 shows that users behave differently in search depending on their linguistic backgrounds. This finding helped shape the design of the multilingual user model proposed in this thesis.

Furthermore, in Section 5.3 an exploratory experiment was conducted to investigate the efficacy of including the language attribute (query language and content language) as a factor in the result adaptation algorithm. The findings from this experiment and the abovementioned behavioural analysis informed the design of the personalisation approach proposed in this thesis in terms of: linguistic attributes that should be in the user model, the way the user's interests should be represented in the model, and they way personalisation algorithms should operate in conjunction with the user model.

3.4 Designing User Models for PMIR

In PMIR, multilinguality can be present in two aspects: (1) users: in terms of the languages that they understand and in terms of their choice of query language when using the search system; and (2) results: in terms of content that is retrieved from multiple languages. As shown in the literature survey, existing research only focused on designing user models that capture the search interests of users in a monolingual manner (i.e. queries and results were in a single language, thus the interest terms in the user model were maintained in that language). For a user model to cater for the interests and attributes of a multilingual search user, the user model has

to reflect the aspect of multilinguality. This calls for novel user model designs that extend existing user models in the multilingual dimension. This extension has subtle implications on system design.

The user model has to capture two types of information about the user: demographic information and information about the user's search interests across languages. As for demographic information, the user model should store attributes associated with language, such as the user's preferred language and a list of languages that the user is familiar with (i.e. ones that the user can comprehend). These attributes will be used in the adaptation algorithms and in making translation decisions (i.e. deciding which result documents to translate from languages that the user is not familiar with to the user's preferred language). Furthermore, the user model may also store information about the user's country of origin and country of residence (current location). Such demographic information is generally used in the localisation industry to adapt content to users depending on their linguistic and cultural backgrounds. Since the attribute of language is the main focus of this study, the personalisation approaches proposed and evaluated in this thesis operated on linguistic information only.

As for maintaining the user's multilingual search interests, two design approaches were considered and compared to each other in the evaluation reported in this thesis:

1. **Fragmented representation:** one way to maintain the terms that represent the user's interests is to keep them grouped by language and in their original form (without translation). That is, the user model stores the terms in multiple languages, where a term is maintained in the same language of the document or query from which it was extracted. Thus the model will be made up of language fragments (language groups); each fragment holding interest terms that correspond to its language. The terms within a fragment are divided along one or more clusters of related terms. The underlying assumption of this kind of representation is: *users exhibit different interests depending on the language they use in search and the language of the available content associated with the search topic; and so, the personalisation process may be more effective if it takes this phenomenon into consideration.* Accordingly, the design of the adaptation algorithms involves making dynamic decisions regarding which fragment(s) to use in the personalisation process.
2. **Combined representation:** another way to maintain the interest terms is to store them all in one language (i.e. a single fragment, containing all clusters of terms); in which case, terms that are extracted from documents or queries that are not in that language are translated to that language (or extracted from the translated versions of the documents/queries). The underlying assumption of this kind of representation is: *the*

personalisation process may prove to be more effective if conducted on the full set of information available about the user's interests (i.e. not just a subset as is the case with the Fragmented representation which divides the model into fragments by language). One important factor to take into consideration in this kind of user model representation is translation quality.

A considerable third approach to designing the user model is to maintain a linked multilingual representation of the user's interest. This entails establishing relationships between interest terms stored in different languages (i.e. detecting similarity between terms across languages) in order to identify common interests across languages. However, this would require semantic mapping between terms (e.g. using ontologies), which is out of the scope of this thesis.

Figure 2 shows an outline of the Fragmented representation and the Combined representation. The study compares the use of the two types of user models in the personalisation process by evaluating the improvement (in terms of search effectiveness) that each one achieves in conjunction with the adaptation algorithms.

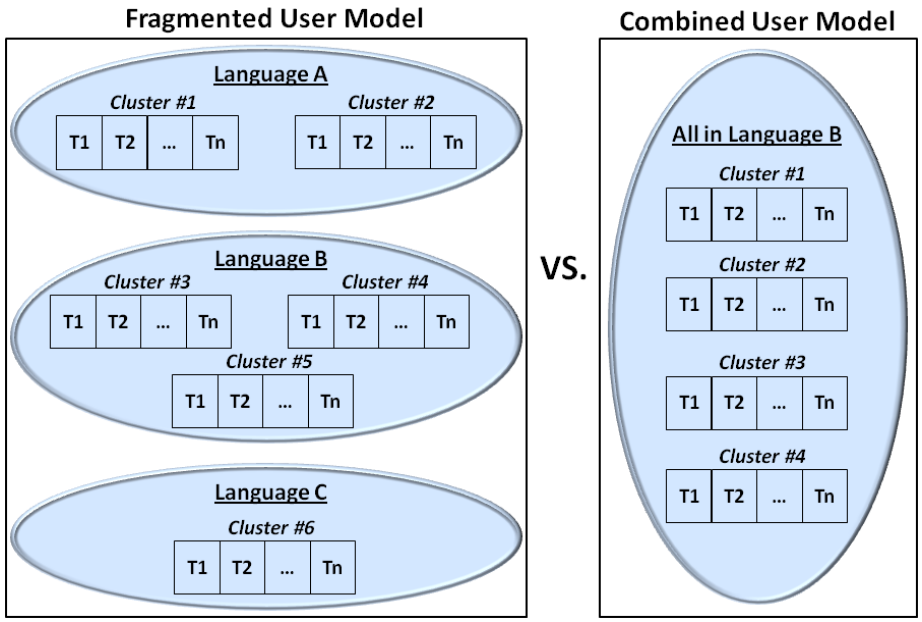

Figure 2: outline of the Fragmented representation vs. the Combined representation of the user's interests

A lesson learnt from the literature review is that the design of the user model should take into consideration the weighting and updating scheme of the interest terms maintained in the model. The terms, and whole clusters of related terms, should have weights that indicate the degree of user's interest in them. Furthermore, there should be a mechanism to update those weights

whenever new information about the user's interests becomes available (e.g. updating the interests whenever the user carries out a new search). This contributes towards a more up to date and accurate representation of the user, and may therefore lead to more effective personalisation. The implication of these design considerations on the user models proposed in this thesis are discussed in further details in the Implementation Chapter (Chapter 4 - Section 4.2).

3.5 Designing the Adaptation Algorithms

Introducing multilinguality to the search personalisation process not only affects the way the user model is represented, but also the way the adaptation algorithms operate. For example, factoring the translation step into the process influences the algorithms in terms of when and how it is carried out. Moreover, the design should also pay attention to the notion of *individualised personalisation* when extending existing query adaptation algorithms in MIR (which, as discussed in the survey, mostly operate on an aggregate level). The key point here is that the adaptation process should be based on information obtained from the user model and not only from the content.

3.5.1 Query Adaptation

A vital factor for the success of the query expansion process is the identification of appropriate expansion terms. Since this study is concerned with individualised query expansion, the mechanism of determining the relevance of user model terms to the source query terms becomes of particular importance. Moreover, the decision of whether or not to expand a query in the first place should be based on the user model (as opposed to content-based selective query expansion discussed in the literature review). The implication of this on the design of the query adaptation algorithms is that the algorithms should comprise a mechanism to determine if there is enough evidence in the user model that indicates that the user has shown previous interest in the topic of the given query; hence, rendering the query as one that may benefit from expanding it based on terms from the user model.

This section outlines two proposed algorithms for query expansion in PMIR (the implementation details and the pseudo-code of both algorithms will be discussed in Chapter 4):

1. **Query expansion based on the user model:** as discussed in the previous section, the user's interests can be represented in one or more clusters of terms. Hence, a challenge that faces the design of the algorithm is how to detect clusters that are relevant to the query. One way to address this challenge is to identify clusters in which the **query terms** appear. However, this may not be sufficient on its own, for example in cases where the specific terms of the query do not explicitly appear in the clusters. Therefore, supplementary mechanisms should be considered to help identify candidate clusters that are of relevance to the **topic of the query**, such as examining similarity between the clusters and alternative representations of the query.
2. **Selective query expansion based on the user model:** the second algorithm is a step further from the first algorithm; query expansion is done in a selective manner, where the decision of whether or not to expand the query is based on evidence from the user model. Thus, a query is only expanded if the degree of similarity between existing terms in the user model and the topic of the query exceeds a certain similarity threshold.

To cater for multilinguality, the query adaptation algorithms take into account the underlying representation of the multilingual user model. The assumption for both algorithms is that the language of the terms used to expand a query match the language of the terms of that query. Thus, when the algorithms are applied in conjunction with the Fragmented User Model representation, only the fragment that corresponds to the query's language is used. Alternatively, when the algorithms are applied in conjunction with the Combined User Model representation, the interest terms are translated to the query's language where necessary.

3.5.2 Result Adaptation

The result adaptation process involves merging and/or re-ranking the search results coming from multiple languages (e.g. operating on three lists of results: English, French, and German). Furthermore, it involves translating the results before displaying them to the user, where necessary. This section outlines two proposed algorithms for adapting result lists in PMIR, and discusses the implication of the presence of multilinguality in the personalisation process:

1. **Merging and re-ranking the results based on similarity score with user model:** this algorithm can be applied in conjunction with both the Fragmented and the Combined User Model. When applied with the Fragmented User Model representation, each result is assigned a score based on its textual similarity with the interest terms present in the

corresponding language fragment of the user model (e.g. scoring the results of the French list against the group of French terms in the model, and therefore, no translation is required). All the results will then be put together in a single list and then sorted in descending order of the assigned scores. Alternatively, when the algorithm is applied in conjunction with the Combined User Model representation, all the results from all the lists will be put together in a single list, translated where necessary to match the language of the user model, and then sorted in descending order of the their textual similarity scores with the terms in the user model.

2. **Re-ranking the results then merging them using a round robin approach:** this algorithm can be applied only in conjunction with the Fragmented User Model. As with the previous algorithm, the results of each list will be separately re-ranked in their original language against the interest terms of the corresponding language fragment of the model. However, the lists will not be merged based on the similarity score; rather, they will be merged using the round robin scheme.

In PMIR, translation plays a crucial role in the adaptation and presentation of results to the user:

1. Whenever there is a mismatch between the language of a result and the language of the interest terms in the user model, the result is translated.
2. When presenting the final result list to the user, the snippet of each result in the list may be subject to translation if it comes from a language that the user is not familiar with.
3. Whenever the user clicks on a result to view the full document, the whole document has to be instantly translated where necessary.

In summary, the key aspects that influence the design of the result adaptation algorithms in PMIR are: (1) how the algorithms operate in conjunction with the way the user's multilingual interests are represented; and (2) how the algorithms factor translation into the process.

3.6 A Framework for the Delivery and Evaluation of PMIR

The analysis and classification of PIR systems provided in Chapter 2, and the design considerations of PMIR discussed so far in this chapter, served to identify the role and specifications of the individual elements of PMIR. The literature review showed that no studies have considered a holistic approach to the PMIR process. To address this gap, this section proposes a framework that orchestrates the elements and workflow of the PMIR process. The framework is designed to be used by researchers in the field of PMIR (and PIR in general) as an

experimentation test bed. It provides a platform for delivering a PMIR service and for evaluating the collective effectiveness of the various components that make up the PMIR process as well as evaluating the net effect of individual components. A high-level design diagram of the framework's components and workflow is shown in Figure 3. The following sections discuss the design of the PMIR framework in terms of functional and non-functional requirements. The discussions draw attention to design decisions that are specific to this study and also to general design considerations that would serve as guidelines for researchers wishing to implement the framework in other studies.

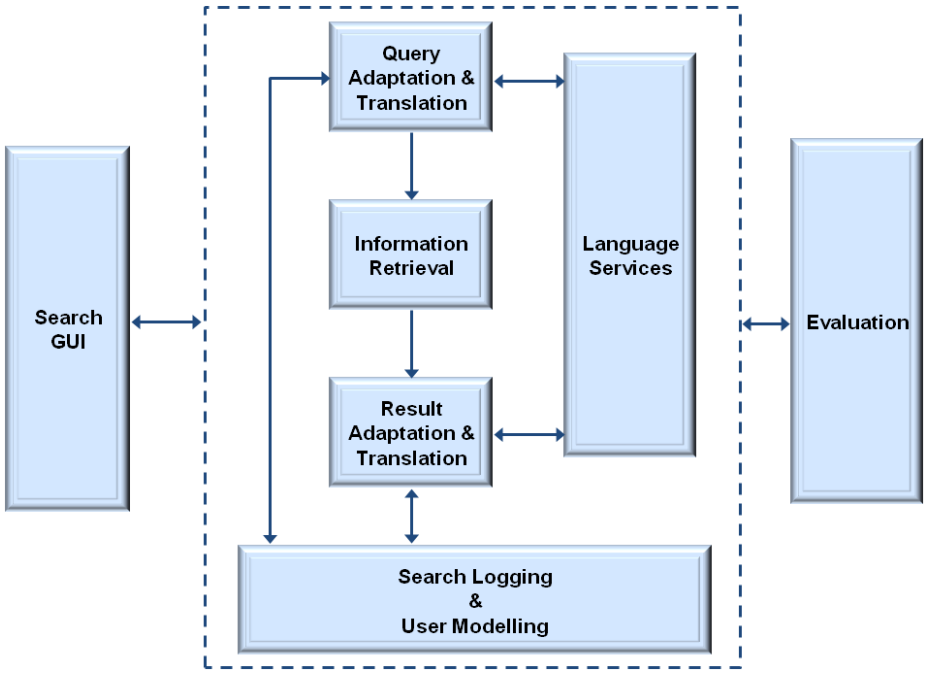

Figure 3: high-level design of the PMIR framework

3.6.1 Functional Requirements

3.6.1.1 Language Services Component

A key component in MIR is the component that carries out the language services. This component is mainly responsible for translating the user's query (translating the source query into multiple target languages) and translating the results (translating the snippets and the whole documents to the user's language of choice). This component is also responsible for language detection, which is used to detect the language of the source query before translating it (in case the user did not specify the language of the query).

As discussed earlier, translation can be carried out using bilingual dictionaries, corpus-based techniques, or Machine Translation (MT). MT has proven well-suited for MIR (Ferro and Peters, 2009, Lavrenko et al., 2002) in terms of translation quality and in terms of availability (as readily available stand-alone systems that support many languages). Therefore, MT lends itself well to the framework.

The plan for implementing the framework should take into consideration the domain (content base) on which the framework will operate in order to select an appropriate MT system. Online MT services such as *Bing Translator*[1] and *Google Translation*[2] are trained on open Web corpus and would therefore be appropriate to use if the framework will be used to provide multilingual Web search. MT systems developed by the research community, such as *MaTrEx* (Stroppa and Way, 2006), can be trained on domain-specific corpora and would therefore be more appropriate to use if the framework will be used to provide a search facility within a certain domain (e.g. training the MT system on technical enterprise content to provide multilingual customer support for that enterprise).

The framework implementation should allow for a configuration parameter to indicate the languages on which the system will operate. The choice of languages to be supported by the system involves three aspects:

1. The languages that are supported by the MT system of choice (this may also involve taking into account the availability of content for training the system in pairs of languages).
2. The languages that the user chooses (if the system intends to enable the user to specify source and target languages for the search).
3. The experimental setup (if the evaluation will focus on certain languages to experiment with).

3.6.1.2 Query Adaptation and Translation Component

As discussed earlier, in PMIR, query adaptation can be performed along two stages: pre-translation query expansion (expanding the query in its source language) and post-translation query expansion (expanding the query after translating it to the target languages). Therefore, when designing the workflow for this component, the design took into account that the component needs to communicate with:

[1] http://www.bing.com/translator
[2] http://developers.google.com/translate/

1. The resource from which the expansion terms will be obtained. This resource can either be the user model (in case of applying individualised adaptation), or a document collection (in case of applying content-based adaptation, e.g. the Pseudo-Relevance Feedback technique).
2. The Language Services Component, which will be used to translate the source query.

This creates the need for a controller module, and a set of configuration parameters, to accompany the component so as to control the execution of, and workflow between, the algorithms and elements involved in query adaptation and translation.

The output of this component is a set of adapted and/or translated queries in multiple languages. These queries will then be passed to the Information Retrieval Component to fetch a list of search results for each query.

3.6.1.3 Information Retrieval Component

This component involves multiple IR systems; one for each language that the system intends to support. Each of these systems operates on a document collection that corresponds to the designated query's language[1].

As with the Language Services Component, the plan for implementation should consider the content on which the framework will operate in order to select or implement an appropriate IR system. The framework can either interface with an online search engine or configure a retrieval (and indexing) system. Availing of a Web service provided by any major search engine (which has an existing index of documents on the Web) can be useful if the intention for an experiment/system is to provide a Web search facility; in which case, the design has to ensure that the selected Web service allows specifying a target language in the service request, in order to perform the search only on documents of that language. On the other hand, configuring a retrieval platform, such as *Lucene*[2] (McCandless et al., 2010) or *Terrier*[3] (Ounis et al., 2005), gives more flexibility and control over the parameters of the retrieval process and may thus be more appropriate if the intention for the framework is to provide a search facility over a closed corpus (e.g. a set of documents for an enterprise); in which case, the design has to

[1] *Caveat:* in some corpora, especially when considering content on the Web, a document may belong to a corpus of a certain language yet partially contains content from another language. So, for example, a possibility that cannot be ruled out is the presence of some English words or sentences inside a German document. The study of this content issue is out of the scope of this thesis.

[2] http://lucene.apache.org/

[3] http://terrier.org/

take into consideration how the framework will communicate with the selected retrieval platform (i.e. taking into account that the framework may have to manipulate it on code level).

For each search, this component retrieves multiple result lists, depending on the number of adapted/translated queries that were submitted to it. The framework should have a configuration parameter to indicate the number of required search results per result list (the value of which can be controlled by the experimental setup). The result lists obtained by this component will be passed as input to the Result List Adaptation & Translation Component.

3.6.1.4 Result Adaptation and Translation Component

The input to this component takes the form of multiple lists of results; one list per language. A result consists of a URL and a snippet (title and summary obtained from the underlying IR service). As discussed earlier, this component is responsible for two functions: (1) re-ranking the multilingual search results; and (2) preparing the results for presentation to the user, where the result lists will be displayed in a separate or merged form (i.e. displayed as multiple lists or as a single merged list). The framework should allow a configuration parameter to indicate whether or not to merge the result lists; the setting for this parameter can either be determined by the user (as a preference stored in the user model) or by the experimental setup.

The workflow for this component involves communicating with two other components:

1. The User Modelling Component:
    a. The re-ranking algorithm will compare the retrieved results to the user's interests stored in the user model in order to determine the degree of relevance of each result to the user. The implication of considering this communication procedure is that the design of the result re-ranking algorithm has to take into account the structure of the underlying user model; if the user model is represented in a single language then the algorithm has to ensure that there is a language match between the results and the interest terms when comparing them to each other, and therefore translate where necessary.
    b. The result preparation process will consult the user model to make translation decisions (i.e. to determine the user's language capability and preferences so as to translate results that are in languages that the user is not familiar with).
2. The Language Services Component. The communication with the translation system may occur in two stages:
    a. Translating the result snippets presented to the user.

b. Instant translation of whole documents upon request (i.e. translate a Web page when the user clicks on its URL in the result list).

Design considerations regarding the quality and speed of result translation will be discussed further below.

3.6.1.5 Search Logging Component

As discussed in the survey, keeping track of the user's search history is a key element in the personalisation process. The information made available in the logs is used as the basis for inferring the user's search interests. Therefore, the framework includes a component that records the user's interactions with the search system, mainly the queries that the users submit and the results (URLs) that they click on to view.

An important matter for this component to consider is that the user may sometimes view a translated version of a document, not the original document. In that case, the search logger should record information about both versions of the document (original and translated version). This information will be useful to the algorithm that extracts interest terms from results that the user clicked; depending on the underlying user model representation, the algorithm may choose to operate on either the original document or the translated one.

3.6.1.6 User Modelling Component

This component is concerned with analysing user and usage information and representing this information in individualised user models. The component can make use of both implicit and explicit information gathering approaches to construct the models. The instance of the framework that is currently implemented for this study (Section 4.1) supports the implicit processing of the users' search logs and explicitly asking the users to supply demographic information about themselves upon signing up. Future/Other implementations of the framework may extend the implicit approach to harvesting information from the users' publically available profiles on the Web (e.g. social networks profiles, professional profiles, etc.) and extend the explicit approach by allowing the users to scrutinise the inferred interests stored in the user model.

The main element in this component is the algorithm that processes the information given in search logs and infers the user's search interests. The algorithm is responsible for analysing the text of the submitted queries and the clicked results to extract key terms from them and

populate the user model. The algorithm is also responsible for assigning weights to the terms using a weighting scheme (e.g. TF, TF.IDF, etc.).

An important design consideration for this component is *updatability*. The interest terms and their weights should be updated regularly in order to ensure that the model is a reliable representation of the user at any point in time. Two mechanisms are put in place for this:

1. A pull-based mechanism, where the User Modelling Component invokes the search-logs analyser algorithm on periodic basis (e.g. on a daily basis).
2. A push-based mechanism, where the User Modelling Component receives notifications whenever new information becomes available in the logs (e.g. with every new search the user carries out)[1].

In terms of workflow design, the User Modelling Component should provide a means for responding to the requests from other components to supply information about the user's interests and demographic information.

3.6.1.7 Evaluation Component

The review of system evaluation provided in Chapter 2 showed how quantitative techniques are used to evaluate the effectiveness of IR systems. From an experimental setup perspective, this involves automating the process of query submission (i.e. repeating the search in a simulated manner) and generating pools of results using various retrieval algorithms. The effectiveness of each algorithm is then computed based on the degree of relevance of the generated results, using various IR metrics.

As the proposed framework is experimental in nature, it includes a component that is responsible for the search automation procedures and for the metrics computation procedures (e.g. calculating Precision, Mean Average Precision, etc.). These procedures will facilitate the process of conducting multiple experimental runs with minimal setup and configuration effort.

[1] In many experimental setups, constructing the user model is done as a one-off step, without the need for further updates, because the full information of the search logs is available *a priori*.

3.6.2 Non-functional Requirements and Other Design Considerations

3.6.2.1 Extensibility and Flexibility for Experimentation

A fundamental design requirement of the PMIR framework is allowing the seamless integration and manipulation of components. The framework's implementation should allow plugging-in, removing, enabling, or disabling alternative components or algorithms at runtime as well as design time. This will facilitate configuring and running multiple sets of experiments; thus, it will enable researchers to focus on implementing and evaluating their own algorithms and components through the framework without having to worry about the rest of implementation details.

Three elements are needed in order for the framework to support extensibility and flexibility:

1. A master controller module to manage the components and the workflow of the whole process from end to end.
2. A master configuration file that has parameters to specify, enable, or disable alternative components and algorithms.
3. Well-defined programming interfaces (e.g. Java interfaces) to guide the implementation of algorithms and communicating with external systems in a unified manner (e.g. communicating with the translation service).

Such flexibility renders the framework well-suited for conducting a diversity of experiments; therefore the framework lends itself well to the fields of IR, PIR, and MIR evaluation.

3.6.2.2 Usability

Although usability is not the main focus of this study, a qualitative evaluation of the multilingual search results feature[1] was nonetheless required (Section 5.5). Therefore, the design took into consideration that the HCI (specifically, the search GUI) should not be very different from what users are used to when dealing with major search engines (e.g. Google, Bing, Yahoo, etc.). From an evaluation perspective, this will help in isolating the effect of good or bad HCI design when evaluating the usability of the multilingual search results feature in specific. Moreover, this will help users get used to the system quickly.

[1] This refers to the notion of interleaving search results from multiple languages in the result list presented to the user.

The adaptability of the HCI should also be taken into consideration. For example, the system should adapt any displayed menus or text to the user's preferred language.

3.6.2.3 General Performance

When using a search system, the users do not expect to have to wait for long periods of time before getting back results in response to their queries –especially given the near instantaneous response offered by modern-day search engines. Although enhancing search engine performance is not the subject of this study, the framework implementation takes into consideration the importance of optimising the overall system performance as much as possible for the sake of usability. This was done on both component level and workflow level.

This consideration will have its effect on implementation decisions, such as the choice of search service and translation service. It will also have its effect on configuration decisions, such as limiting the retrieval of results to a relatively small number of results (e.g. working with the top 10 or 20 search results in each language).

3.6.2.4 Translation Quality and Speed

As translation plays a crucial role in PMIR, especially when translating result documents, the implementer of the framework should pay particular attention to selecting a high quality MT system. From a practical perspective, the best available MT systems cannot produce "perfect" translation. Nevertheless, the MT system may be deemed "successful", if it is capable of producing *acceptable-quality* translations of documents. The notion of *acceptable* refers to translation that is good enough to convey the original information of the document to the user. To this effect, this study carried out qualitative evaluation of the users' perception of document translation quality when they interacted with the PMIR system (Section 5.5). The outcome of the evaluation was in agreement with the notion of *acceptable quality translation*.

In terms of speed, the time required for translating result snippets or documents should be kept to a minimum. Therefore, using a highly responsive MT system (or Web service) is essential. Furthermore, on workflow level, the framework has to optimise the speed of communication that takes place with the Language Services Component. For example, if the system needs to translate 30 titles and 30 document summaries then it would be faster to submit them in chunks in fewer batches than to submit them one by one.

3.6.2.5 Privacy

In this thesis, the approach to inferring the user's interests is based on analysing the user's search logs. Although client-side logging may sometimes provide a higher degree of privacy for users of online systems, the design decision was for the framework to employ server-side logging. Two reasons are behind this decision:

1. The framework is experimental in nature; so, from a functional perspective, tracking the users' interactions on the server and storing the logs in a centralised repository facilitates the implementation and the operation of the experiments.
2. The study of privacy issues is out of the scope of this thesis. Nevertheless, other studies, especially ones in which privacy is the focus of the research, can alter the design of the framework to operate with client-side logging; this will require alterations to the design of the User Modelling Component, the Search Logging Component and the workflows associated with both of them.

3.6.3 Summary of Features

This short section presents a summary of the features/requirements of the PMIR framework and where the need for each one has originated from. For example, some of the features were inspired from the literature review; others were inspired from best practices in the field of Software Engineering (SWE), and so on.

Table 7: summary of the framework's features/requirements

Feature/Requirement	Originated From
The decision that a framework was needed in the first place	Existence of IR frameworks in the research community (e.g. Terrier), and the lack of one for PMIR.
A master controller for the PMIR workflow	Best practices in SWE
Query Adaptation & Translation	PIR/MIR literature
Result Adaptation & Translation	PIR/MIR/AH literature
Search logging	PIR literature
User modelling	AH/PIR literature
Evaluation	IR research community & IR literature
Extensibility and flexibility	Best practices in SWE

Usability	Best practices in HCI & in industry
Performance and responsiveness	Best practices in industry & in SWE
Using an MT system to carrying out translations	MIR literature & best practices in industry

The solution design presented in this chapter partially addressed the following challenges:

Challenge #1: *What are the key components of the search personalisation process and how can the process accommodate a personalised multilingual search service?*

Challenge #4: *How should query adaptation and result adaptation algorithms be extended in order to incorporate the aspect of multilinguality?*

Chapter 4: Implementation

This chapter discusses the implementation of the PMIR framework and the personalisation algorithms proposed in this thesis. Regarding the framework, the discussion involves: (1) illustrating the technical and functional details of the various components of the framework; (2) highlighting how the design considerations stated in the previous chapter guided the implementation; and (3) explaining the reasons for certain implementation decisions. The goal is to show how the framework enables the delivery and implementation of PMIR. Regarding the algorithms, three categories are discussed: user-model construction, query adaptation, and result adaptation. For each algorithm, the discussion involves: (1) providing the pseudo-code; (2) illustrating the operation details; and (3) highlighting some insights about the algorithm.

4.1 An Implementation of the PMIR Framework

This section describes an implementation of the PMIR framework proposed in this thesis. This implementation provides a system for evaluating the collective effectiveness of the various components that make up the PMIR process, as well as evaluating the net effect of individual components. It also provides the necessary GUI and components to carry out user studies. The implementation served as the basis for a set of experiments reported in Chapter 5 (Sections: 5.4 and 5.5). The implementation is also intended to serve as a complete test bed (i.e. a fully functional experimental system) that can be used by researchers in the fields of IR, MIR, PIR and PMIR.

The system is fully implemented in the Java language (Java SE and Java EE) and follows the MVC architecture (Model-View-Controller –for Web Applications) (Leff and Rayfield, 2001). The system allows for plugging-in, removing, enabling, or disabling alternative components or algorithms at runtime as well as design time. The seamless integration and manipulation of components aims to facilitate the process of running multiple sets of comparative experiments and evaluations. This enables researchers to focus on implementing and evaluating their own algorithms without having to worry about implementing the rest of the components involved in the PMIR process. Thus, the framework provides a platform for comparisons of PMIR approaches. The components of the system are shown in Figure 4 and are described in details in the following subsections.

An early version of the PMIR system was demonstrated in the public showcase event of: *Autumn 2010 Innovation Showcase at Microsoft Ireland (in conjunction with Innovation Dublin 2010)*[1]. Furthermore, another version of the system was used to carry out experimentation by a collaborating research group from *Dublin City University* in the *Personalized and Collaborative Information Retrieval Track at the Forum for Information Retrieval Evaluation (FIRE 2011)* (Ganguly et al., 2013).

Those two versions served as mock tests for the functionality of the PMIR system and helped in detecting flaws in the implementation of the system features. The feedback from both trials helped in enhancing the system from an architectural perspective (e.g. altering the design and the workflow of the components). The feedback also helped in enhancing the system from a usability perspective (e.g. introducing multi-threading and other implementation decisions that greatly enhanced the performance and responsiveness of the multilingual search service).

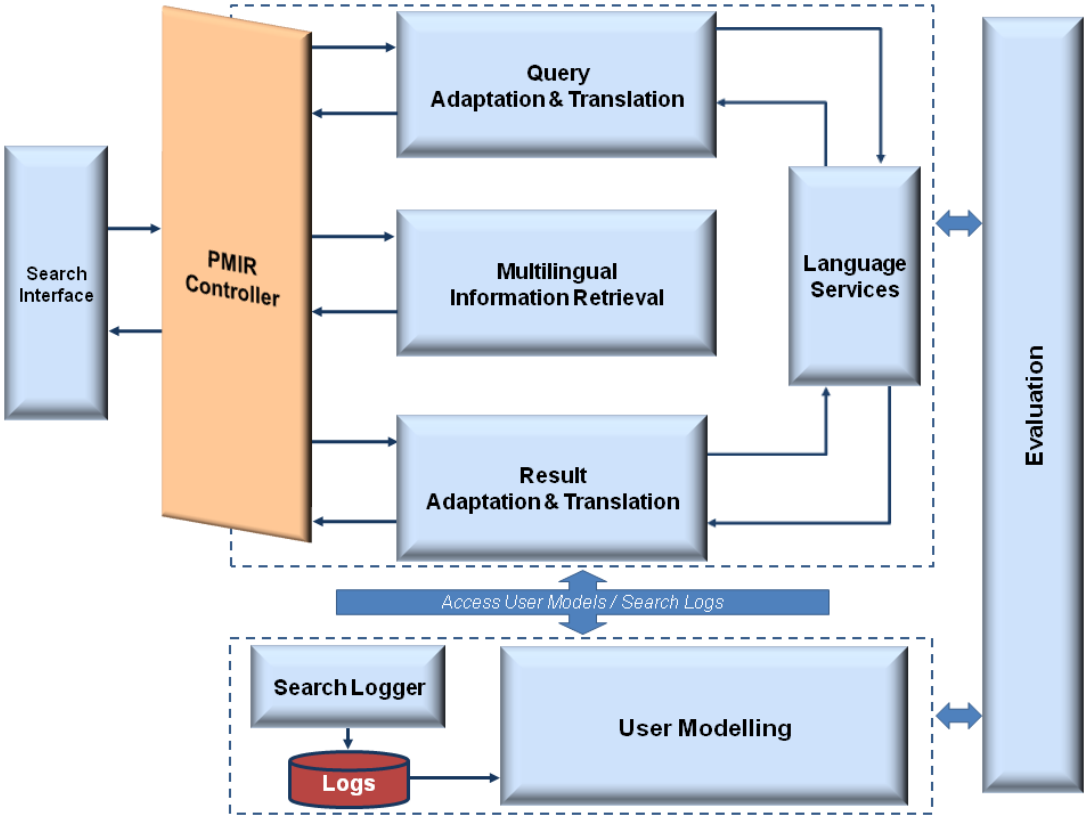

Figure 4: overview of the system's components

[1] A brief about the event can be found at:
http://www.cngl.ie/drupal/sites/default/files/CNGL_Localisation_Innovation_Showcase.pdf
and at:
http://www.isin.ie/go/news_events/events/centre-for-next-generation-localisation-cngl-showcase

4.1.1 Search Interface

The system implemented for this thesis offers a multilingual Web search service. The system was used in the experiments reported in Sections 5.4 and 5.5. This section describes the Graphical User Interface (GUI) of the online search system and provides screen shots.

As per the requirements specified in Chapter 3, the search interface is designed in a simple way that resembles common search engines on the Web. The objective of this is to separate any HCI factors from the evaluation of the multilingual search system. In other words, this was done so that the only new feature that the users encounter when dealing with the system would be the feature of merging (interleaving) search results from multiple languages. A description of the search interface from a user perspective is provided below.

The default search page (i.e. the landing page of the system) is shown in Figure 5. In addition to the search box, the user is asked to specify the language of the query s/he is submitting. The current implementation of the system supports searching in three languages: English, French, and German (a discussion about the choice of languages to support in the system is provided in Section 4.1.3). The default interface language is English. The menu provides an option that allows the user to change the interface language to any of the three languages.

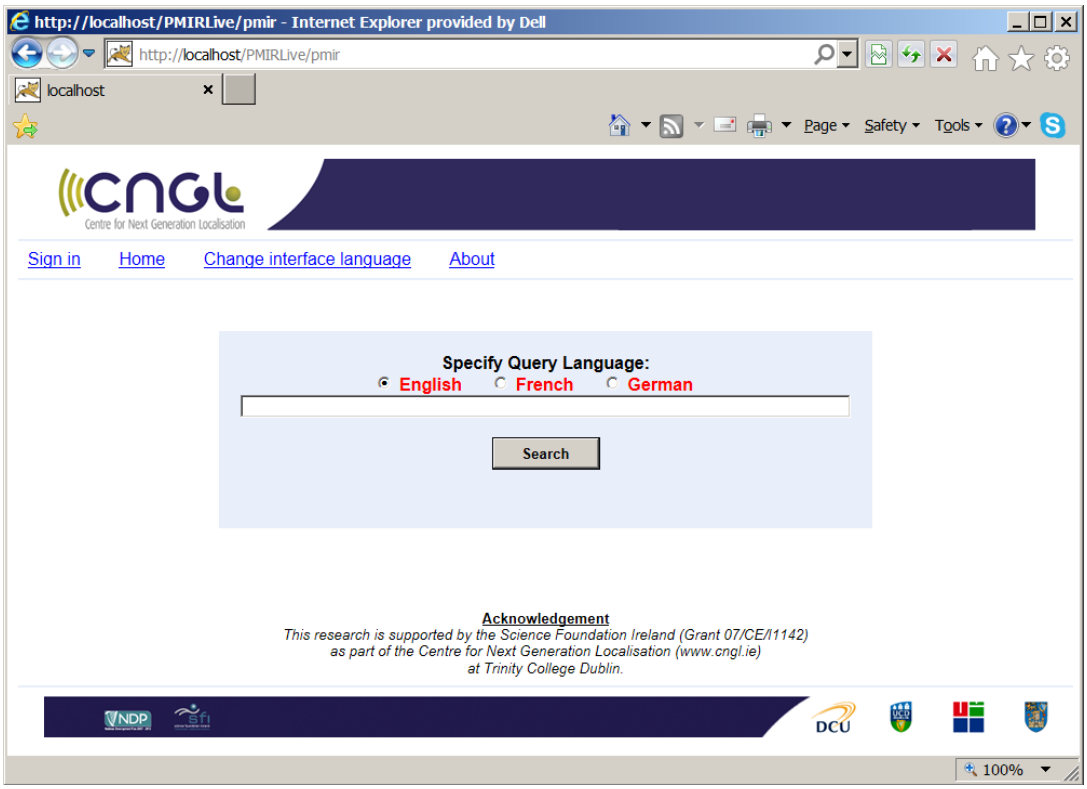

Figure 5: default search page (no user signed-in)

Figure 6 shows the search page after a user "*French User 1*" has signed in. This user is assumed to be a user whose native/preferred language is French and who also understands English. The interface is adapted to the user's preferred language, which is French.

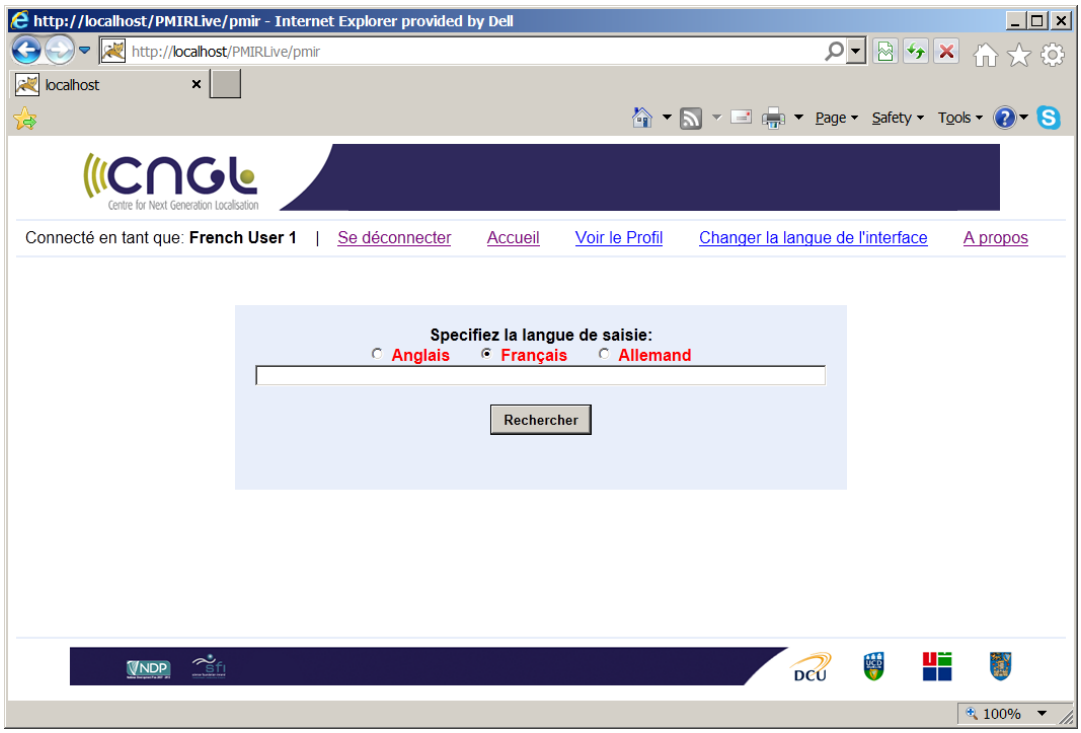

Figure 6: search page with adapted menu (French user signed-in)

The search-results page is shown in Figure 7. It shows the results of submitting the French query "aliments biologiques" (which means "organic food"). The result list shows the snippets of results from the three languages, merged using the round robin scheme. Since the user does not understand German, the German results are translated to French. Translated results are marked with an orange piece of text underneath to indicate the language that they were translated from (displayed as *"Traduit du: Allemand"* under the third result shown in the figure); if the user clicks on this result, a fully translated version of the Web page will be displayed (instant translation). Moreover, the user is also given an additional link to view the original (non-translated) page if s/he wants to.

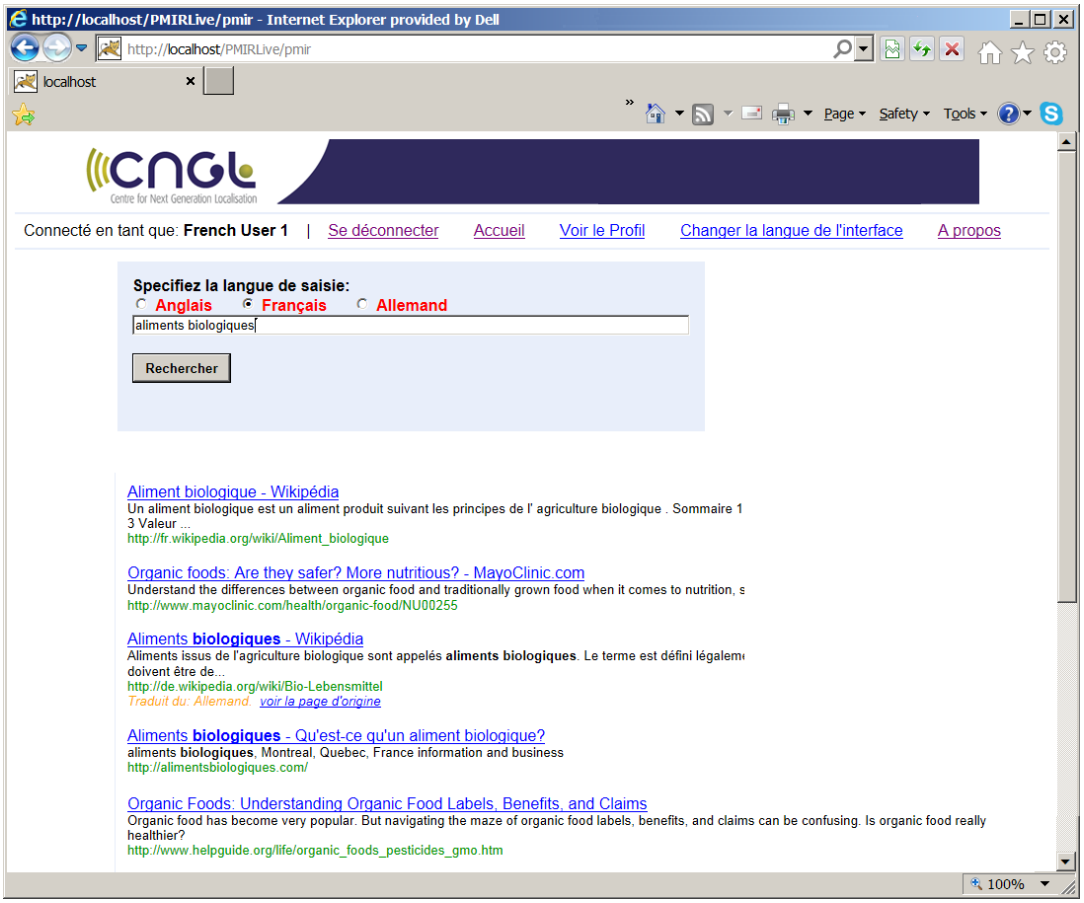

Figure 7: multilingual search results (merged/translated)

The Search Interface Component interacts with the PMIR Controller by passing it the submitted query, the language of the query, and a user identifier (optional: if a user is signed in). The PMIR Controller executes the PMIR cycle and then returns the search results to the Search Interface to be displayed to the user –translated to the user's preferred language where necessary. If the user is not signed-in then the system operates in a non-personalised mode[1]; in which case, all the results are translated to the language of the source query.

This section showed a visual overview of the multilingual Web search front end. This provided a representative view of the user experience. The following sections discuss the details of the whole process and the workflow between the components.

4.1.2 PMIR Controller and Configuration File

In order to control the various components of the system and manage the flow of data between them, it was necessary to implement a master controller for the whole PMIR process. The controller acts as a pivot for communicating with the main components of the system and is responsible for passing processed data between them. Figure 8 shows a detailed version of the framework and illustrates the system's workflow. The operation details of the PMIR Controller are as follows:

- **Input:** a search query, the language of the query, and a user identifier. The query language and user identifier are optional parameters; if the query language is not specified, it will be automatically detected by the Query Adaptation & Translation Component; if there is no user identifier, the system will operate in a non-personalised mode (i.e. will provide an MIR service rather than a PMIR service). The input can either come from the Search Interface (in case of a live user trial) or from the Evaluation Component (in case of running a search automation procedure for experimentation).
- **Output:** a set of result lists (one list in each language that the system is configured to operate on) and/or a single result list (a merged list).
- **Operation:** upon receiving the query and the parameters, this data is stored in the search logs (by communicating with the Search Logging Component). The data is then passed to the Query Adaptation & Translation Component which responds by returning a set of adapted/translated queries in different languages. The set of queries are then passed to the Multilingual Information Retrieval Component which returns multiple

[1] For the sake of the experiments reported in Chapter 5, the system was configured to always require signing-in before using the search system.

lists of search results (URLs and snippets), where each list is in a different language. Afterwards, the multiple lists of results, along with the optional user identifier, are passed to the Result List Adaptation & Translation Component which returns either of the following: a single merged list of adapted/translated results, multiple separate lists in their original languages, or both. The final result list(s) are then returned to the component that submitted the query (e.g. the Search Interface Component, the Evaluation Component, etc.).

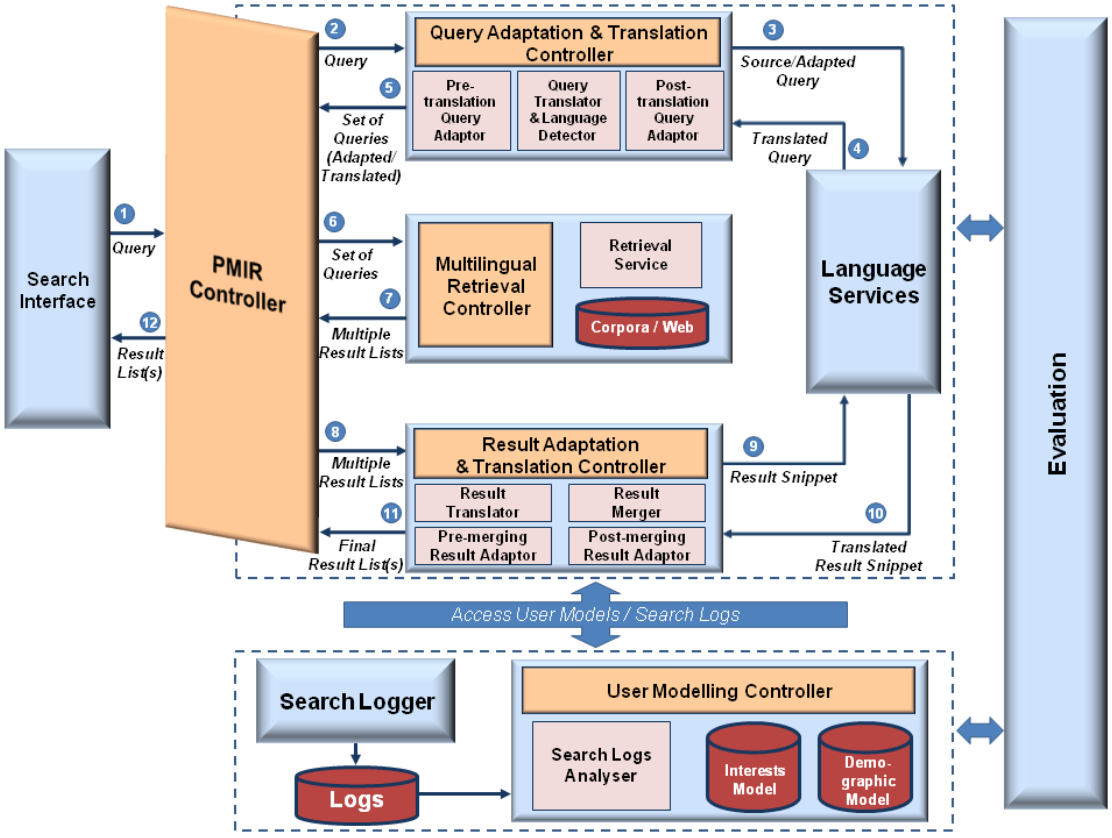

Figure 8: workflow and detailed view of the system's components

In addition to the master controller, the implementation features a master configuration file which holds properties (parameter-value pairs) that manipulate many aspects of the PMIR process. The properties include: specification of operating languages, number of search results to retrieve, database connection settings, external services' connection settings, specification of adaptation algorithms to use, user modelling parameters, experimentation-specific settings, and so on. A full list of properties is given in Appendix-D. This configuration file greatly minimises the need to alter the implementation of the system at code level. It therefore facilitates controlling the behaviour of the framework when deploying a live version of the system and when running experiments.

Upon initialising the system, the PMIR Controller reads and loads the configuration file into memory. It then becomes accessible to all the components of the system. Altering the system configuration can be done offline (at design time) or online (at runtime). The PMIR Controller, the configuration file, and the overall design of the system workflow enable seamless integration of components into the framework and alternating between various algorithms. For example, the activity of trying (running/evaluating) a new adaptation algorithm becomes factored down to two simple steps: implementing the desired algorithm and then altering the system's configuration to plug in the new algorithm (either by editing a single parameter in the configuration file to point to the designated algorithm or by calling a certain method that communicates with the loaded version in memory). This gives flexibility to experiments that run multiple successive evaluations using a series of alternative algorithms.

4.1.3 Language Services Component

In this implementation of the PMIR framework, the Language Services Component internally uses *Bing Translator API*[1] to carry out the language operations: translation of pieces of text (queries and result snippets), translation of whole Web pages, and language detection of queries. Following on the design considerations discussed in the previous chapter, this API was chosen for the following reasons:

1. It is generally known to perform relatively well in terms of translation quality and speed.
2. It supports a range[2] of languages.
3. It provides a well-defined RESTful[3] Web service to communicate with it, including a *batch-translation feature* (allows the submission of an array of strings at once which greatly reduces the communication overhead time by reducing the number of service calls).
4. It is available for free and allows an unlimited number of translations.

A limitation, however, that was observed with the Bing API –and in fact other similar APIs– is that sometimes they come to a temporary halt when they receive a large number of successive requests from the same source (rate limits). This causes some practical inconvenience when running multiple successive experimental runs which involve a great deal of translation.

[1] http://www.bing.com/translator/

[2] The number of supported languages is 43 (in the year 2013).

[3] Representational State Transfer (REST) is a type of Web-service that can be invoked using simple HTTP requests: http://en.wikipedia.org/wiki/Representational_state_transfer

For the multilingual Web search experiments conducted in this study, three languages were used: English, French, and German (however, the system is not limited to these languages). The reason behind choosing these languages is that they have a rich set of content available on the Web[1]. This is an advantage that reflects on two aspects: (1) it contributes to the quality of translation expected from Bing API because of the availability of content for training the translation system; and (2) it provides a variety of Web search results to the users of the system. Further discussion of languages and character-encoding issues is provided later (Section 4.1.10.2).

4.1.4 Search Logging Component

This component is responsible for keeping records of the queries submitted by the users, the results retrieved for each query, and the results that the users clicked on for each query. Logging the clicked results involves logging the URL and snippet (title and summary) of each result, but not the content of the document itself. Because the users sometimes view translated results, the logger keeps track of both the original result and the translated result; this information is used later by the User Modelling Component when extracting keywords from the original or the translated version of the result (depending on the user model representation sought). For each user action logged by the system, the system also allows logging additional data: timestamp, IP address, and session identifier. This data is useful for session-based analysis of search activity.

This implementation of the framework used the open source database *PostgreSQL*[2] as the underlying repository for the logs[3]. In order to enhance the system's performance, the Search Logging Component makes use of multi-threading where the procedure of communicating with the database (reading/writing logs) runs in a separate thread.

[1] http://www.internetworldstats.com/
[2] http://www.postgresql.org/
[3] Other implementations of the framework can use other sorts of data structures, not necessarily a relational database, such as XML files.

4.1.5 User Modelling Component

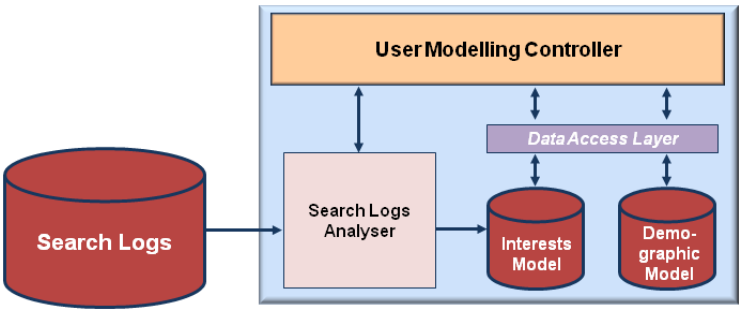

Figure 9: detailed view of the User Modelling component

This component harnesses a set of repositories and subcomponents that together are responsible for constructing, maintaining, and updating the user models, and are also responsible for the authentication of system users (user sign-up/sign-in). The following is an overview of the user modelling operation:

- **Input/Resources:** user and usage information.
- **Output:** a set of user models on which the system bases the personalisation process.
- **Operation overview:** the User Modelling Component maintains a repository of user interests (*the Interests Model*) that are inferred by automatically analysing the user's search history. It also maintains the demographic information that the user supplies upon signing up with the system (*the Demographic Model*). This information provides a basis for the system to make personalisation decisions, which can take place in individual components or several ones combined. More operation details are discussed in the following subsections.

4.1.5.1 User Modelling Controller

This controller is responsible for managing the Search Logs Analyser subcomponent and the repositories of the user modelling process. Moreover, it provides methods that can be invoked by other components of the framework to request information for the personalisation process (e.g. requesting the necessary user information for adapting queries, results, or the GUI).

4.1.5.2 Search Logs Analyser

This subcomponent features the algorithm that processes the search logs in order to extract terms that represent the user's search interests and to assign weights (degrees of interest) to them. These terms can be extracted from: the text of the queries, the text of the result snippets,

or the actual text of the result documents. The following is a description of how the Search Logs Analyser operates:

- **Input/Resources:** the user's search history.
- **Output:** a weighted set of terms that represent the inferred interests and the user's degree of interest in each term.
- **Operation overview:** following on the design considerations discussed in Chapter 3, it was important to clearly identify when and how this subcomponent is used to populate and update the Interests Model. The current implementation of the framework supports both a pull-based and a push-based mode:
    1. Pull-based mode: the User Modelling Controller can invoke the procedure for analysing the search logs in order to populate the Interests Model. This is the mode used in the experiments reported in Section 5.4 (performed as one-off step before executing and evaluating the various adaptation algorithms).
    2. Push-based mode (live mode): the PMIR Controller notifies, and passes necessary information to, the User Modelling Controller whenever the user performs a new action (submits a query or clicks on a result); at which point, the search-logs analyser algorithm is invoked specifically for the newly available items of information.

The details and pseudo-code of the search-logs analyser algorithm (i.e. the user model construction algorithm) are presented in Section 4.2.

4.1.5.3 Interests Model

This is the repository that holds information about the inferred search interests of the users. The current implementation of the framework uses PostgreSQL database, where a set of tables are used to maintain the interest terms along with their weights and languages (the language of the text from which a term was extracted).

The Interests Model is accompanied by a dedicated subcomponent that acts as an interface layer for accessing the data stored in the repository (controls read/write operations). This is a common practice in the Software Engineering field and is part of conforming to the MVC design pattern.

4.1.5.4 Demographic Model

This is the repository that holds demographic information about the users. The current implementation keeps track of the following attributes: native language, preferred language, familiar languages (languages in which the user can consume information), native country, and current country (current location)[1]. This information is supplied by the users when they sign up with the system.

This model is also accompanied by a data access layer that interfaces with the PostgreSQL database.

4.1.6 Query Adaptation and Translation Component

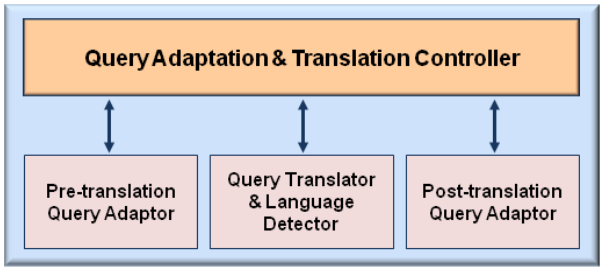

Figure 10: detailed view of the Query Adaptation & Translation Component

This component comprises a set of subcomponents that are responsible for translating and/or adapting queries. The following is an overview of the component's operation:

- **Input:** query, language of query (optional), and a user identifier (optional). The input is received from the PMIR Controller.
- **Output:** a set of adapted/translated queries in multiple languages. The output is returned to the PMIR Controller.
- **Operation overview:** this component holds the algorithms for expanding the query before or after translating it to multiple languages. If a user identifier is passed to this component then the query expansion operations are based on the user model; otherwise, the query may either be expanded based on the content-base or not expanded at all (depending on the settings in the configuration file). If the language of the query is not passed to this component, then it will be automatically detected prior to translation.

[1] The experiments reported in this thesis operate on the language attributes only. This is based on conclusions from exploratory investigations that will be reported in Chapter 5 (Section 5.3).

4.1.6.1 Query Adaptation & Translation Controller

This controller is responsible for managing the workflow of the query adaptation and translation process. Based on the specifications in the configuration file, it invokes the necessary subcomponents to execute various parts of the process. It is also responsible for communicating with the PMIR Controller (input/output).

Furthermore, this controller is responsible for communicating with the User Modelling Component to obtain information about the user's interests. This information is then passed to the query adaptation algorithms.

4.1.6.2 Query Translator and Language Detector

This subcomponent acts as an intermediary for communicating with the Language Services Component to detect the query's language (if not pre-specified) and to translate the source query. As part of the communication, this subcomponent makes decisions regarding how to submit the query to the Language Services Component in order to translate it into multiple target languages; if the underlying translation system supports batch translation (i.e. a translate-one-to-many feature) then the query is submitted once as a one-off step, which reduces overhead time; otherwise the query is submitted multiple times (each time for a different target language).

4.1.6.3 Pre-translation and Post-translation Query Adaptors

These two subcomponents represent the algorithms that adapt the source query before it is translated and that adapt the queries obtained from translating the source query. The operation of these subcomponents is controlled by a set of parameters in the master configuration file. These parameters allow the following:

1. Specifying the algorithms that are plugged into these subcomponents (in the form of the name of the Java class files in which the algorithms reside). This includes specification of which algorithms to use in case of running in personalised mode (i.e. if a user identifier was passed as input) or running in non-personalised mode.
2. Enabling or disabling the whole query adaptation process or parts of it (e.g. enabling pre-translation adaptation but disabling post-translation adaptation).

The pseudo-code and details of the query adaptation algorithms are explained in Section 4.3.

Following on the design requirements of flexibility and extensibility, the framework allows re-configuring these parameters at runtime. This provides the necessary means for running consecutive sets of comparative experiments using alternative algorithms and configurations.

4.1.7 Multilingual Information Retrieval Component

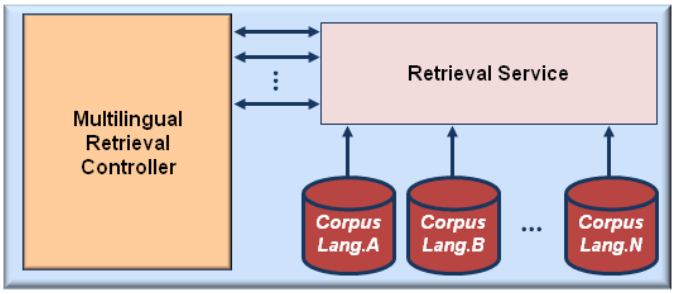

Figure 11: detailed view of the Multilingual Information Retrieval Component

This component is responsible for retrieving documents from document collections of different languages. The following is an overview of the mode of operation of the component:

- **Input:** a set of queries in multiple languages (multiple variations of the source query). The input is received from the PMIR Controller.
- **Output:** a set of result lists; one list per language, corresponding to the languages of the submitted queries. The output is passed back to the PMIR Controller.
- **Operation overview:** this component comprises a controller and a retrieval service (search engine). The retrieval service comprises multiple monolingual IR systems; each one responsible for retrieving content in a certain language.

4.1.7.1 Multilingual Retrieval Controller

This controller is responsible for managing the workflow of the retrieval process. It operates the retrieval service to obtain result lists in multiple languages. In order to meet the non-functional requirement of **system responsiveness**, the following mechanisms are used to **optimise the performance** of the process:

1. The controller makes use of **multithreading** where the set of search queries are submitted for retrieval in parallel (i.e. each query is submitted in a separate thread).
2. The configuration file contains a parameter for specifying the number of results to retrieve per language. For the multilingual Web search system reported in Section 5.4, the parameter was set to 10 results per language; yielding a final single list containing

30 merged results in response to the user's query[1]. The advantage of operating on a smaller set of search results is that it minimises the time needed for processing these results in further stages of the framework (e.g. adapting and translating all the results). This can be considered as a way to overcome scaling limitations with respect to the number of search results to process/return.

4.1.7.2 Retrieval Service

For the multilingual Web search system implemented for this study, the *Bing Search API*[2] was used as the Retrieval Service. Following on the design specifications of the framework, this service was chosen for the following reasons:

1. Bing is one of the major search engines on the Web.
2. It provides a well-defined RESTful web service to communicate search requests to it.
3. It allows specifying a target document collection with the search request, based on language[3]. For example, when submitting a request with a French query, the API allows specifying *French* as the target collection.

   ***Caveat*:** this gives a "hint" to the search engine that French search results are sought; while the search engine usually "honours" this specification, it is not guaranteed that the returned results will be entirely in French. This is because of two reasons: (a) if there is a limited set of documents (or no documents) in the French collection that are relevant to the query, the search engine may retrieve additional results from other document collections; and (b) as discussed earlier, it may be the case that a document is mostly written in French but contains some pieces of text written in other languages. Nevertheless, those are not common occurrences and studying this effect is not within the scope of this thesis[4].
4. It allows a large number of search requests for free.
5. It is a responsive API[5].

[1] This means that the users were presented with 3 pages of search results in response to their queries (10 results per page). Displaying 3 pages was considered sufficient for the experiment; this is based on lessons learnt from the literature review where it was found that users seldom browse beyond the second page of results and that the majority of users click on search results that are within the top 5 results presented on the first page.

[2] http://datamarket.azure.com/dataset/bing/search

[3] Microsoft uses the term "market" to denote a subset of the content (or a facet of the service) that is associated with a certain language, country, or geographical region.

[4] It is left to the experiment designer (other researchers who may make use of the framework) to decide how to deal with this issue.

[5] The API usually responds within a fraction of a second.

4.1.8 Result Adaptation and Translation Component

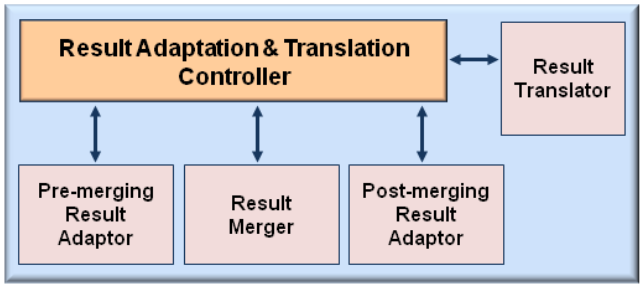

Figure 12: detailed view of the Result Adaptation & Translation Component

This component comprises a set of subcomponents that are responsible for carrying out three operations on the search results: adapting the result-lists, merging the result-lists, and translating the results. The following is an overview of the component's operation:

- **Input:** multiple lists of results (one list per language), and an optional user identifier.
- **Output:** either one of the following: (a) multiple lists of results (one list in each language on which the system operates); (b) a single list of results (results from all languages merged together); or (c) both: separate lists and an additional merged list.
- **Operation overview:** this component adapts the search results to the user's interests and the user's language and then prepares the results for presenting them to the user in a separate or merged form. The details of the process are described in the subsections below.

4.1.8.1 Result Adaptation & Translation Controller

This controller is responsible for managing the workflow of the adaptation and translation process. It communicates with the PMIR Controller concerning the input and output result lists and communicates with the User Modelling Component to obtain information about the user's interests and language preferences. If no user identifier was passed to this controller, then it operates in the non-personalised mode, which affects a number of adaptation/translation decisions (explained below).

This controller uses a configuration parameter that specifies the required form of output lists (multiple lists, single list, or both). The value of this parameter can either be specified by the user of the system (via the Search Interface) or by the implementer of the system/experiment. For the purpose of the experiments reported in Section 5.4 the system was pre-configured to return the merged list only.

4.1.8.2 Results Translator

This subcomponent acts as an intermediary for communicating with the Language Services Component to translate the snippets of the search results. The translation decisions are based on the user's language preferences; in order to support multilingual users, only results that are in unfamiliar languages to the user are translated to the user's preferred language. In case of non-personalised mode (i.e. no user identifier available) then all the results are translated to the language of the source query.

As part of the communication with the Language Services Component, this subcomponent pre-processes the result snippets and makes decisions regarding how to submit them to the translation system as follows:

1. If the underlying translation system supports batch translation then the titles and summaries of the results are prepared into arrays of strings and submitted in a one-off step.
2. Multithreading is used; each result list is processed and translated in a separate thread.

These measures were found to significantly enhance the system performance[1].

Concerning the experiments reported in Section 5.4, it is worth noting here that when a user clicked on a result to view the whole document (Web page) a special feature in Bing Translator API was used; this feature allowed instant translation of whole Web pages by passing the page's URL to the API instead of passing the pieces of text from within the document.

4.1.8.3 Result Adaptation and Merging Subcomponents

These subcomponents comprise **a set of algorithms that work in conjunction with each other** to personalise the search results. The process mainly involves re-ranking the search results based on the interest terms in the user model. The re-ranking procedure can take place before or after the results are merged. The Result Adaptation & Translation Controller manipulates the sequence of execution of the algorithms (compound stages) depending on the settings in the configuration file and depending on the structure of the underlying Interests Model.

[1] Based on human observation these measures improved the system's response time to become less than one-tenth of the original response time.

The Result Merger subcomponent supports three modes to merge the multiple result lists into a single list:

1. Round robin merging: each result list is taken in turn.
2. Score-based merging: these are scores assigned by the Pre-merging Result Adaptor subcomponent which indicate the similarity between each result and the user's interests. The lists are put together and then ordered based on those scores.
3. Putting the whole lists sequentially after each other in one list without actually interleaving them (this is not "merging" *per se*, but can be carried out by the merger subcomponent when needed).

The Result Merger is also responsible for removing any duplicate results that may appear between the multiple result lists. This could happen, for example, when searching for a person's name, in which case the underlying search engine API may sometimes return the same URLs in the lists of different language.

The Pre-merging Result Adaptor and the Post-merging Result Adaptor are mutually exclusive (otherwise the latter would be overriding the work of the former). The type of representation of the underlying Interests Model dictates which one to use. For example, in the case of the **Fragmented User Model** representation, each result list may be **scored separately** against the corresponding language-fragment **before** merging the lists (e.g. results of the English result list will be scored against the interest terms stored in the English fragment of the user model, and results in the French list will be scored against the French fragment, and so on). In the case of using the **Combined User Model** representation, then the adaptation process will take place **after** the result lists have been put together[1]. If operating in non-personalised mode then the results are merged using round robin and no adaptation takes place. The pseudo-code and details of the result adaptation algorithms are explained in Section 4.4.

Similar to the discussion of the Query Adaptation & Translation Component, the result adaptation/merging process is controlled by a set of configuration parameters that allow plugging-in/removing and enabling/disabling alternative parts of the process at design time and at runtime. This contributes to the flexibility and extensibility of the framework and renders it suitable for a variety of experiments.

[1] Logically speaking, performing a post-merging result re-ranking process would only make sense if the third mode of results merging is used.

4.1.9 Evaluation Component

This component comprises a set of procedures (i.e. methods in Java classes) for performing the following activities:

1. Constructing user models offline. This is done as a one-off step before running the experiments that evaluate the personalisation algorithms.
2. Automating PMIR cycles for experimentation. This is known as *result pooling*, which is a common activity in IR experiments where the system automatically generates multiple sets of candidate results for a test query using a variety of retrieval/personalisation algorithms. The automation procedure can dynamically alter the framework's configuration (i.e. the value of the parameters) in order to execute comparable experimental runs with alternative settings.
3. Conducting quantitative evaluation of the effectiveness of the personalisation algorithms. This involves computing IR metrics such as Precision.
4. Storing the evaluation results in a database and writing them to text files in tabulated form (e.g. excel/csv format).

Furthermore, this component comprises a set of GUI screens for carrying out the following tasks:

1. Relevance Judgments. This involves asking users to judge the degree of relevance of a set of pooled search results to a query; a common task in IR evaluation (Baeza-Yates and Ribeiro-Neto, 2011).
2. Topic Descriptions. This involves asking users to enter metadata about the queries that will be used for testing the system (known in the IR community as: *topics*). These are TREC-style[1] topic descriptions which comprise fields for: the text of the query, a short description of the query, a narrative describing the type of content that would be deemed relevant/irrelevant to the query. The topic descriptions are used in the Relevance Judgments process to assist users in accurately judging results with respect to the queries.

An important feature in the Evaluation Component is that it has the ability to access any other component in the framework; the enabling factor being the highly structured workflow put in place for the framework, which allows for executing the PMIR process either as a whole (i.e. from end to end) or in part (i.e. executing specific parts of the process). For example, to automate a search cycle, an evaluation procedure can either communicate with the PMIR

[1] TREC: Text REtrieval Conference: http://trec.nist.gov/

Controller (in the same way that the Search Interface does) or communicate directly with the individual controllers of the components under evaluation. As per the design considerations discussed earlier for the Evaluation Component, this facilitates testing specific parts of the PMIR process and conducting multiple experimental runs with minimal setup and with a consistent user experience.

4.1.10 Other Implementation Details

This section discusses a number of implementation decisions concerning the PMIR framework and highlights the significance and implication of these decisions. Moreover, the section also discusses a number of general issues that faced the implementation, and how these issues were dealt with.

4.1.10.1 Conforming to Good Software Engineering Practices

Since the framework is intended as a platform to be used by other researchers, it was important to follow good practices in Object Oriented Programming when designing and implementing its various components. This enables users of the framework to implement their own components or algorithms and to easily integrate them into the framework. This subsection highlights some of the key classes and interfaces of the framework.

In order to standardise the way each component (i.e. Java class) provides its functionality, a set of Java interfaces were put in place; one for each component. Any component wishing to provide a certain feature has to follow (i.e. implement/inherit) the specific Java interface that defines how this feature is provided. For example, if a Java class wishes to provide a method for query adaptation it has to implement the **`QueryAdaptor`** interface which dictates the signature of the method that is to hold the implementation of the query adaptation algorithm. In addition to defining the role of a component, these interfaces also stand out as a definition of the way other components (especially the controllers) can interact with that component. This enables framework users to freely re-implement the methods/algorithms within a component without having to worry about re-integrating it in the framework. In other words, this allows implementers to replace existing components with alternative components without interrupting the data workflow.

Two key objects in the PMIR workflow are the **query** and the **result**. Most of the processing that takes place within the framework operates on either one of them; thus, these objects are frequently passed on between components in one form or another as input or output (e.g. source/adapted/translated queries or results). This called for creating a class **`Query`** and a class **`Result`** for the purpose of standardising the representation of each within the framework in terms of attributes (e.g. text and language) and state (e.g. source/adapted/translated). Objects of these classes also hold references (links) to each other so as to indicate their transformation along the processing stream; for example, a *translated result object* would hold a reference to its *source result object*. The significance of this is:

- From an experimentation perspective, this facilitates studying the effect that each processing step has on the object (e.g. examining queries before and after being adapted or translated).
- From an operational system perspective, this facilitates making runtime decisions about which object to use at certain times (e.g. the decision of whether to display a source result or its translated version, depending on the user's language preferences).

As part of conforming to the MVC architecture, a set of *data access* Java classes were implemented to serve as a layer that governs the interactions with any underlying data source (i.e. reading/writing operations). For example, a set of classes were implemented to shadow the items in the search logs such as `SubmittedQuery` and `ClickedResult` along with all the associated attributes. Another example is the set of classes that shadow the storage of the user models: `DemographicModel` and `InterestsModel`. The significance of providing these layers of indirection is that they minimise the amount of changes needed in case framework users wish to use alternative data storage mechanisms –whether another relational database software or another data storage format altogether (e.g. using XML files to store data instead of databases).

As discussed earlier, the final output of the PMIR Controller is a set of adapted/translated search results. These results are returned as a collection of `Result` objects to the component initiating the search (which may be the Search Interface or an automation procedure in the Evaluation Component). An additional feature provided by the PMIR Controller is that it can return the results in two textual formats upon request: JSON[1] or XML; these formats are used as standard formats for representing data returned from Web services (e.g. RESTful Web services). This feature was implemented so that the PMIR framework can be exposed as a Web

[1] JavaScript Object Notation (JSON) is a lightweight data-interchange format that is both human-readable and machine-readable (http://www.json.org/).

service in the near future. This will allow the whole PMIR process to be available as a stand-alone unit that can be seamlessly integrated with other Web services or online systems wishing to include adaptive multilingual search as part of their service (API-style integration).

4.1.10.2 Issues Concerning Multilinguality

A major issue when working with content in languages other than English is *character encoding*. This is particularly the case when processing or displaying special characters (e.g. letters with diacritics, such as the accented letters in French and the *umlaut* in German letters). During early stages of implementing and testing the framework it was observed that sometimes the characters were not properly encoded and thus not properly displayed to the user. It was noted that these malformed characters either existed in the text of the original document or occurred as a result of a processing operation that took place over the text –whether internally (handling text within the framework) or externally (e.g. text returned from a translation/retrieval service[1]). In order to deal with such character encoding issues, a number of **server-side** and **client-side** mechanisms were put in place:

1. Configuring the Web **server** that hosts the framework to use UTF-8 encoding (which is a W3C recommendation[2]).
2. Adding HTTP headers to all Web pages to indicate to the client browser that the pages contain UTF-8 content.
3. In spite of the two abovementioned mechanisms, the following work-around was still found to be needed at times: a method was implemented to detect and fix specific character encoding problems that were sometimes encountered in the text that is propagated between the framework components. This should not be considered as a limitation to the framework, but rather a problem with content curation (i.e. the existence of documents that do not conform to standard character encoding guidelines).

Part of the decision regarding the choice of languages to support in the current implementation of the framework was based on character encoding issues encountered with various languages. This can be considered as a limitation in the current implementation. Furthermore, another limitation, especially with regards to the Search Interface, is that the current implementation of the framework does not support right-to-left languages (e.g. Arabic).

[1] This problem was humanly observed when working with Google APIs and Bing APIs. However, the possibility that the framework implementation itself was the root cause of the problem cannot be ruled out (e.g. mis-communicating with the APIs or mis-handling of returned textual formats).

[2] http://www.w3.org/International/questions/qa-forms-utf-8

The following is an issue that was encountered with MT services (aside from the matter of the quality of translation returned[1]): it was observed that, sparingly, MT services do not return a translation for the submitted piece of text. This was observed in two cases:

1. **Returning the same word(s) that was submitted to it**; thus indicating that no translation is available for that word. A limitation in the current implementation of the framework is that it does not provide any special handling for this case. The effect of this is that there will exist pieces of text in the PMIR workflow that are labelled as being in a certain language while partially they are not (e.g. existence of German words within a piece of French text). These were regarded as minor occurrences and were not investigated any further in this thesis. In future implementations, this can be handled by applying language detection on any piece of text returned from an MT engine to ensure that the returned text is in the expected language. To this effect, it will have to be taken into consideration that the effectiveness of that approach is highly reliant on the accuracy of the language detection mechanism/system used.
2. **Not returning any translation at all** (these are occurrences that are attributed to technical issues, such as miscommunication-with or unavailability-of the online MT service). For query translation, the current implementation of the framework handled this issue by eliminating (skipping) the processing associated with the language of the query in the PMIR workflow of that query; thus, no result list is sought in that language for that search query, and the rest of the process goes normally for the remaining languages. For snippet translation, this issue was handled by re-inserting the text of the original snippet in place of the translation.

It was also observed that, for certain words, if the word is submitted to the MT system with a capitalised first letter then the system yields a different translation than if the word is submitted in small letters. This was noted for some proper nouns. For example if the word "windows" is submitted for translation from English to German then the MT system returns "fenster" (meaning *window* as in a *room's window*), but if it is submitted as "Windows" then the MT system returns "Windows" (i.e. implicitly indicating that it was interpreted as the *Windows Operating System* and therefore should not be translated). As the study of specific translation issues and proper-noun handling is out of the scope of this thesis, the framework setup used in the experiments propagated the words in the search workflow exactly as the user submitted them.

[1] Translation quality is a corpus and domain issue, which is outside of the scope of this study.

Nevertheless, outside of the work carried out specifically for this thesis, an early trial of the framework involved the integration of a query adaptation component that addressed the word capitalisation issue in queries. This was part of the collaborative research carried out within the CNGL[1] project. The research underpinning that component was reported in (Leveling and Jones, 2010b) where the authors proposed an approach to query adaptation named *Query Recovery*. The approach involved applying natural language processing techniques to adapt queries in a way that recovers their original intent (e.g. capitalising certain letters, adding punctuation, etc. –which are things that many search users disregard when submitting queries). The study showed that adapting the queries in that manner before using them as input to MT and CLIR systems lead to improvements in translation quality and retrieval effectiveness. This query adaptation component was plugged into the PMIR framework as part of a live demonstration at an internal CNGL event.

4.1.10.3 Additional Features

The following are a number of additional features that are available in the current implementation of the framework:

- A property file for localising GUI items (menu items, labels, button texts, etc.). These are string attribute-value pairs used in adapting the interface according to the user's preferred language. The current implementation supports properties for English, French, and German strings. Additional properties can be added to support other languages.
- The *Snowball Stemmer*[2] package for grammatical processing of words in various languages.
- The framework comes with readily implemented Java classes to communicate with other well-known search/translation APIs (e.g. Google and Yahoo APIs).

[1] The research reported in this thesis was conducted as part of the ongoing collaborative research taking place in the Centre for Next Generation Localisation (http://www.cngl.ie).

[2] http://snowball.tartarus.org/

4.1.10.4 Experiment-specific Implementation Details

A set of additional classes and database tables were created specifically for the purpose of the experiments reported in Sections 5.4 and 5.5. This included:

- A set of Web pages for interacting with users who participated in the experiment (e.g. participant information screens, search-tasks explanation/selection screens, etc.).
- Code to control and run the phases of the experiment.
- Database tables to store experiment-specific data (e.g. the users' status in the experiment phases, the number of tasks completed, etc.).

Screen shots and further discussion will be given in the Chapter 5.

4.1.11 Summary of Framework Implementation

The previous subsections described key features of the PMIR system. The goal was to show how the implementation enables the delivery and the evaluation of PMIR. Moreover, a number of technical challenges and limitations were discussed; of which, the ones in scope of this study were addressed, and the rest were highlighted for the benefit of other researchers who may wish to provide their own implementation of the framework.

The system facilitates conducting experiments in the area of PMIR, and can also be easily re-configured to conduct experiments in the areas of IR, MIR, and PIR. In order to conduct an experiment that involves user trials, the experiment administrator only needs to focus on the following:

- Implementing the algorithm(s) that s/he wants to test and editing the framework's configuration file accordingly.
- Deciding upon the document corpora that the experiment will operate on, and adjusting the retrieval component accordingly.

Since the system caters for both service delivery and service evaluation, it is meant to be of benefit to two kinds of users: researchers and end users. With respect to researchers, the system offers a flexible experimentation platform that takes a major part of the burden of the experimental setup away from them. With respect to end users, the system provides a useful application for seeking information across languages; an application that features an easy-to-use search interface (as it resembles common search engines on the Web, which they are used to) that adapts to their language preferences.

4.2 Constructing User Models for PMIR

This section is organised as follows: first, an overview of the proposed user models is given below. Then, the details are given in the following subsections.

Introducing **multilinguality** to the user model affects the kind of **attributes** stored in the model as well as the **structure** of the model. In terms of attributes, this study proposes the inclusion of the following set of attributes:

1. **Native Language:** the user's native language.
2. **Familiar Languages:** a list of languages that the user understands[1].
3. **Preferred Language:** this language will be used for the following:
    a. Search results that come from languages that the user is not familiar with will be translated to this language.
    b. The search interface will be displayed in this language (menu items, labels, button texts, etc.)
    c. The user's interest terms in the Combined User Model representation will be maintained in this language.

These attributes will be included in all the types of user models discussed below[2].

In terms of structure, specifically concerning the representation of the user's multilingual search interests, two approaches were proposed in the design chapter: the **Fragmented** approach and the **Combined** approach. The underlying assumption of the Fragmented approach is that the users' search interests are language-biased, and therefore better personalisation may be achieved if the user model reflects this phenomenon. In turn, this will reflect on the way the adaptation algorithms are implemented to operate on the separate fragments of the user model. On the other hand, the underlying assumption of the Combined approach is that better personalisation may be achieved if the adaptation algorithms operate on the full set of information available about the user, instead of fragmented subsets. Figure 13 shows an example overview of the two user models.

[1] In the experiment reported in Section 5.4, the users were asked to enter a list of languages in which they had moderate proficiency or higher.

[2] It may also be sensible to include two more attributes in the user model: the user's country of origin and country of residence (current location). However, only the language attributes fall within the scope of this thesis. Nevertheless, the current implementation of the PMIR framework maintains country and language attributes for each user, which may be used in future research.

For both approaches, the model comprises multiple vectors of weighted terms (clusters of potentially related terms). The terms are obtained from the user's search history (submitted queries and snippets of clicked results). The weights represent the degree of user's interest in the terms and are computed based on the normalised Term Frequency (TF) scheme (Baeza-Yates and Ribeiro-Neto, 2011). Furthermore, each vector is labelled by a language, indicating the language of the terms stored in it, and is assigned an overall weight that represents the user's interest in that vector as a whole.

For the experiments reported in Section 5.4, the decision to extract interest terms from result snippets, as opposed to extracting them from result documents, was taken based on a number of reasons:

1. The experiments follow on the approach adopted by several studies reviewed in the literature survey, where result snippets are regarded as query-focused summaries of the documents (Speretta and Gauch, 2005, Ruvini, 2003).
2. Processing/Translating the full text of each documents is a very lengthy operation, especially when multiplied by many users and search sessions.
3. A current limitation in the current implementation of the framework is that it cannot process certain types of documents such as PDF documents and documents with Flash content.

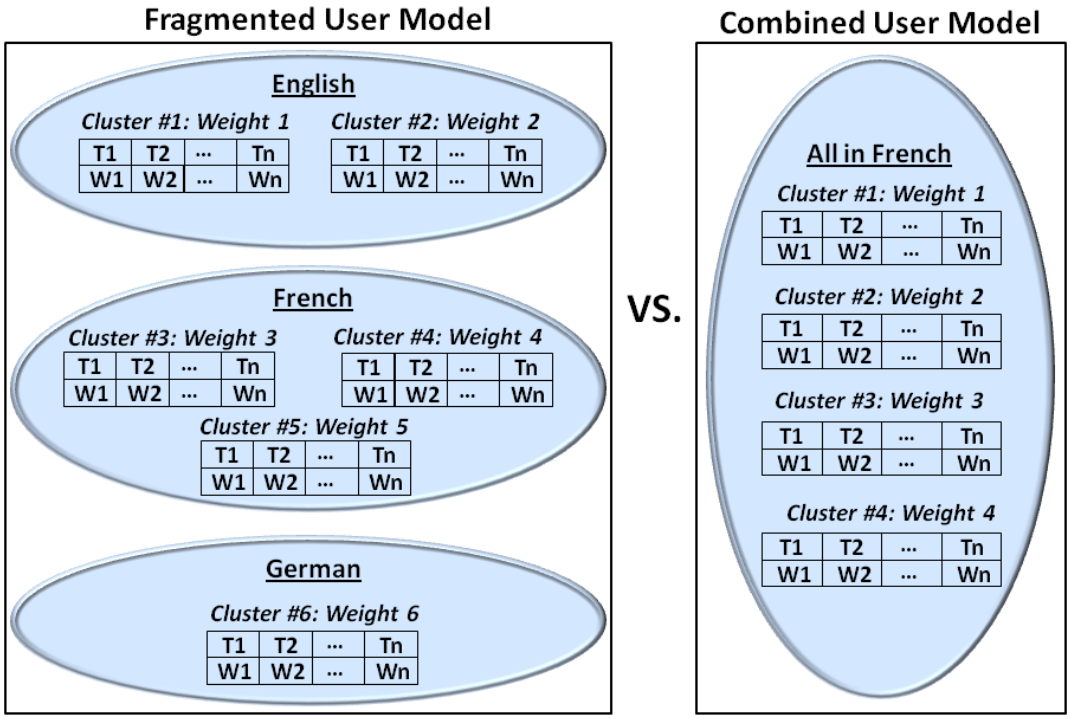

Figure 13: example overview of Fragmented vs. Combined User Model

From an implementation perspective, **the Combined approach itself can be constructed in two different ways**: either to create the Combined model as a translated version of an existing Fragmented model or to construct the Combined model from scratch by extracting terms from the translated queries and translated result snippets. This brings the total to **three types of user models** that cater for the user's interests across languages:

1. **Fragmented User Model:** the model contains a fragment for each language: English, French, and German. Each fragment may contain multiple vectors of weighted interest terms. A term is maintained in the same language of the object (query or result snippet) from which it was obtained, and is stored under the corresponding language fragment; thus, no term translation takes place.
2. **Early-Combined User Model:** the model is made up of a single fragment. It contains multiple vectors of weighted interest terms. All terms are maintained in the user's preferred language. To extract terms from objects, the objects are first translated to the preferred language (if they did not originally exist in the preferred language) and then term extraction takes place. The notion of "early" is used to denote the fact that translation is performed on the objects –before the model is created.
3. **Late-Combined User Model:** in order to create this model, the Fragmented model has to be created *a priori*. This model is made up of a single fragment. It contains multiple vectors of weighted interest terms that are maintained in the user's preferred language. The terms are obtained by translating the existing terms in the Fragmented User Model to the preferred language (if they did not originally exist in the preferred language). The notion of "late" is used to denote the fact that translation is performed on existing terms –after a Fragmented model has been created.

Each type of user model along with its accompanying set of adaptation algorithms (explained in Sections 4.3 and 4.4) can be considered as a complete personalisation suite for PMIR. The experiments reported in Section 5.4 compare the three suites to each other by evaluating the improvements in retrieval effectiveness that each one achieves over a non-personalised baseline.

The three subsections below present the algorithms used to construct the three types of user models. This involves the pseudo-code and implementation details of each algorithm.

4.2.1 Fragmented User Model

This subsection discusses the algorithms for populating and updating the Fragmented User Model. In the Fragmented User Model, the interest terms are obtained from the source version of the submitted queries and the clicked results; no translation takes place.

The user model is populated as follows:

Let L be the set of languages supported by the system such that
$L = \{lang_1 , lang_2 , \dots lang_x\}$, where $lang_i$ is a language that may have a fragment associated with it in the user model (depending on the available queries and results recorded in the logs).
Let Q be a set of queries such that $Q = \{q_1 , q_2 , \dots q_y\}$, where q_j is a query that was previously submitted by the user to the system.
Let $S_{(i,j)}$ be the set of result snippets belonging to $lang_i$ that were clicked by the user for q_j, such that $S_{(i,j)} = \{s_1 , s_2 , \dots s_z\}$.

Algorithm#1: Populating the Fragmented User Model
Input: *L, Q, S*
Output: Multiple sets of user model vectors, grouped by language.
Steps: **1: For each** $q_j \in Q$ **2:** **For each** $lang_i \in L$ **3:** **Select** $S_{(i,j)}$ **4:** **Gather** all terms appearing in $S_{(i,j)}$ together **5:** **Assign** TF weight to each term **6:** **Sort** terms in descending order of TF **7:** **Select** top N terms **8:** **Store** selected terms and their weights in a vector $\vec{v}$ **9:** **If** language of q_j matches $lang_i$ **10:** **then** **11:** **Assign** TF weight to each term appearing in q_j **12:** **Add** the terms of q_j and their weights to $\vec{v}$ **13:** **Sort** terms of $\vec{v}$ in descending order of their weights **14:** **Select** top N terms and discard the rest **15:** **Store** $\vec{v}$ in the user model under the fragment of $lang_i$ **16:** **Assign** an overall weight *w* to $\vec{v}$

Narrative: For each query that the user submitted, the snippets of the clicked results for that query are grouped by language. For each language, the snippets belonging to the language are processed together to extract the terms that most frequently appear in them. The extracted terms

are assigned TF weights and then stored in a vector. If the designated language matches the language of the query then the query terms are also assigned TF weights and stored in the vector. The terms in the vector are sorted in descending order of their weights, and only the top N terms are maintained (the rest are removed from the vector). The vector is then added to the user model under the corresponding language-fragment, and is given an initial overall weight *w*. In the experiments, the initial value of *w* is set to "**1**". This value indicates that this vector has just been added for the **first** time. This value will later increase as the vector is subject to merging with other vectors in the update mechanism.

Following on the design considerations discussed in Chapter 3, it was important to put in place an update mechanism in order to ensure that the user model is actively representing the interests of the user. The number of vectors to maintain per language-fragment (*M*) and the number of terms to maintain in a vector (*N*) are set to certain thresholds[1]. If the maximum number of vectors for a language $lang_i$ is reached, then for any new incoming vector $\vec{v}_{new}$ associated with $lang_i$ the model is updated as follows:

Let V be the set of vectors already maintained in the user model under the language $lang_i$ such that $V = \{\vec{v}_1, \vec{v}_2, \ldots \vec{v}_n\}$, where $\vec{v}_j$ is a vector of weighted terms that represents a cluster of user's interests within $lang_i$, and w_j is the overall weight of $\vec{v}_j$.
Define *F:cs* as a function that computes the cosine similarity between any two vectors (*A* and *B*) computed as (Manning et al., 2008):

$$cs(A,B) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

[1] The values of these thresholds will be discussed in the experimental setup in Chapter 5.

Algorithm#2: Updating the Fragmented User Model.
Input: $\vec{v}_{new}$, V, $lang_i$
Output: Updated user model vectors.
Steps: **1:** **Add** $\vec{v}_{new}$ to the set of vectors V **2:** **For each** $\vec{v}_j \in V$ **3:** **Let** $k = j + 1$ **4:** **For each** $\vec{v}_k \in V$ **5:** **Let** $Sim_{[j,k]} = cs(\vec{v}_j , \vec{v}_k)$ **6:** **Select** the two vectors $\vec{v}_j , \vec{v}_k$ which received the highest similarity score $Sim_{[j,k]}$ **7:** **Combine** $\vec{v}_j \; and \; \vec{v}_k$ in one vector $\vec{v}_{comb}$ as follows: ***Gather*** all terms from the two vectors and store them in $\vec{v}_{comb}$ ***Sort*** the terms of $\vec{v}_{comb}$ in descending order of weights ***Maintain*** the top N terms only. **8:** **Add** $\vec{v}_{comb}$ to the set V **9:** **Let** overall weight $w_{comb} = w_j + w_k$ **10:** **Remove** $\vec{v}_j \; and \; \vec{v}_k$ from the set V

Narrative: the new incoming vector is added to the existing vectors of the language-fragment that corresponds to its language. Using cosine similarity, all the vectors within the fragment, including the new one, are compared to each other and then the two most similar vectors are merged together. They are merged by grouping together all the terms from the two vectors into a single vector, and then sorting the terms in descending order of their weights. Only the top terms, according to the threshold, are maintained in the merged vector (the rest are discarded). The overall weight of the merged vector is set as the sum of the weights of the two vectors. Thus, higher vector weights will indicate that a vector was subject to merging several times; this implies that the topic (cluster) represented by this vector is of high importance to the user as it was repeatedly observed in the user's searches.

4.2.2 Early-Combined User Model

This subsection discusses the algorithms for populating and updating the Early-Combined User Model.

In this model, the interest terms are obtained from:

- The snippets of clicked results:
    - If the snippets are in the preferred language, the terms are obtained directly from them.
    - If the snippets are not in the preferred language, they are translated to the preferred language and then the terms are obtained from that translated version
- The source query, only if it is in the preferred language.

Terms are **not** obtained from **translated queries** because the chances of inaccurate translation are higher (resolving polysemy becomes a greater challenge for the MT system when dealing with shorter text that has little context). Inaccurate translation may lead to the presence of inaccurate interest terms in the user model (i.e. inferring wrong interests about the user). This is not considered the case with **translated results** because the fact that the user clicked on a result is assumed as an implicit indication that the translated snippet showed terms that were of relevance to what the user was searching for.

The user model is populated as follows:

Let Q be a set of queries such that $Q = \{q_1 , q_2 , \dots q_y\}$, where q_j is a query that was previously submitted by the user to the system.
Let S_j be the set of snippets of all results that were clicked by the user for q_j, such that $S_j = \{s_1 , s_2 , \dots s_z\}$.
Let $lang_{pref}$ be the preferred language of the user as exhibited in the user model.

Algorithm#3: Populating the Early-Combined User Model
Input: Q, S, $lang_{pref}$
Output: Multiple user model vectors, all in the preferred language.
Steps: **1: For** each $q_j \in Q$ **2:** **Select** S_j **3**: **For each** $s_j \in S_j$ **4:** **If** language of s_j does not match $lang_{pref}$ **5:** **then** **6:** **Replace** s_j with translated version of s_j to $lang_{pref}$ **7:** **Gather** all terms appearing in S_j together **8:** **Assign** TF weight to each term **9:** **Sort** terms in descending order of TF **10:** **Select** top N terms **11:** **Store** selected terms and their weights in a vector $\vec{v}$ **12:** **If** language of q_j matches $lang_{pref}$ **13:** **then** **14:** **Assign** TF weight to each term appearing in q_j **15:** **Add** the terms of q_j and their weights to $\vec{v}$ **16:** **Sort** terms of $\vec{v}$ in descending order of their weights **17:** **Select** top N terms and discard the rest **18:** **Store** $\vec{v}$ in the user model **19:** **Assign** an overall weight *w* to $\vec{v}$

Narrative: For each query that the user submitted, the snippets of all the clicked results for that query are identified. Snippets that are not in the preferred language are translated to the preferred language. The snippets are then processed together to extract the terms that most frequently appear in them. The extracted terms are assigned TF weights and then stored in a vector. If the language of the query is the same as the preferred language, then the query terms are assigned TF weights as well and are stored in the vector. The terms in the vector are sorted in descending order of their weights, and only the top N terms are maintained. The vector is then added to the user model and is assigned an initial overall weight *w*.

The update mechanism of the Early-Combined User Model is similar to the update mechanism of the Fragmented User Model. The only difference is that the procedure is applied on the whole user model instead of just being applied on a fragment.

4.2.3 Late-Combined User Model

The Late-Combined User Model is maintained in the user's preferred language. It is created based on an existing Fragmented model. This is done by copying the vectors of the preferred-language-fragment of the Fragmented model and then translating the vectors of the remaining fragments to the preferred language.

The update mechanism of this model is the same as the update mechanism of the Early-Combined User Model. Once the process of translating all the vectors of the Fragmented model is complete, the update mechanism is invoked to ensure that the number of vectors in the Late-Combined model is kept under the threshold. By invoking the update mechanism, a series of vector-merging operations will keep taking place until the number of vectors in the model meets the threshold.

4.2.4 Additional Details about the User Modelling Process

This subsection provides some additional details regarding: the user models, the interest terms, text processing, and translation operations.

A fact that is worth highlighting is that the vectors in the user model stand out as unlabelled clusters of interests. The key point here being that no text classification techniques are used to classify the terms into pre-defined categories or to attach labels to the clusters. It is expected though that clusters will end up holding terms that are of relevance to each other as a result of running the update algorithm (where vectors that are textually similar to each other are merged together)[1].

In the literature review, the notions of **short-term interest** and **long-term interest** were discussed. The review also discussed personalised systems that attempted to take both kinds of interests into consideration. Accounting for the user's long-term interests is useful for personalising searches about topics that are recurring over time (i.e. in multiple sessions). Accounting for short-term interests is useful for personalising ad-hoc searches in a single session. Following on the discussions of the population and update algorithms in the previous sections, it is worth highlighting that **the user models proposed in this study account for both kinds of interests.** This is explained as follows:

[1] Terms within a cluster were indeed observed to be of relevance to each other. This was observed during the demonstrations of the early versions of the PMIR framework (mentioned at the beginning of Section 4.1).

- Accounting for long-term interests: by selecting similar vectors to be merged together in the update operation, the user model is essentially catering for long-term interests. Furthermore, increasing the overall weights of the vectors whenever a merging operation is performed reflects the frequency of exhibiting these recurring interests.
- Accounting for short-term interests: a notable fact about the update operation is that when a new vector is added to the model, that new vector is not necessarily the one that gets selected for merging with an existing vector; rather, after the new vector is added to the model, whichever two vectors turn out to be most similar to each other are merged together. Thus, if the newly added vector represents a new topic of interest then it will not likely be selected for merging at the point of adding it, and will therefore affect personalisation for the current session.

A series of text processing operations is applied on the terms before storing them in the model:

1. **Stop-word**[1] **removal:** the framework has a list of stop-words to remove in each language. These words are removed from the queries and result snippets before being analysed for term extraction. This is a commonly used procedure in the IR field.
2. **Stemming:** grammatical processing was also applied on the terms of the queries and the snippets where the words (the letters therein) were reduced to shorter forms using the snowball stemmer. For example, the word "trying" is stemmed to: "try". This is also commonly used in IR.
3. **Maintaining two "faces" for each term:** in some cases, such as when re-ranking a result list based on the interest terms in the user model, the **stemmed version** of a term is more suitable for use in the process. In other cases, such as when expanding a query before submitting it to the search engine, the **original version** of a term is more suitable for use in the process. Therefore, a class `Term` was created in the framework; an object of this class maintains both versions of a term at all times. This enables each algorithm that operates on the terms to use the version that suits it. Moreover, when a query or a snippet is being analysed, if multiple words were stemmed to the same root then only one `Term` object will end up in the user model for them; holding the stemmed version and all the encountered original versions. For example if the words "globalised" and "globalisation" appear in the text, they will end up in the user model as one term whose stemmed form is stored as the string "globali" and whose original forms are stored as the array of strings {"globalised", "globalisation"}.

[1] Stop-words are words that commonly appear in sentences of a language. They include, but are not limited to, articles ("a", "an", "the"), some common verbs (e.g. "be", "have", etc.), common words (e.g. "after", "before", "because", "for", "also", etc), and so on. In the field of IR they are sometimes considered as words that do not add an information value to the query or to the document being analysed.

The abovementioned text processing operations are not only applied during the user modelling process, but are also applied whenever a piece of text is being processed. For example, when re-ranking a list of results against the user model, the same text processing is applied to the snippets (or documents) before they are compared to the user model.

In the discussion of the design considerations of the Search Logger Component it was recommended that, for a clicked result action, the Search Logger should record both the original snippet of the clicked result and the translated version that was displayed to the user (if the result was a translated one). This was taken into consideration when implementing the Search Logger. This was useful for the algorithms of populating/updating the Early-Combined User Model where the procedure of obtaining the translated version of a snippet did not always require communicating with the translation service. Rather, the search logs were first consulted to check if a translated snippet (in the preferred language) already existed and could therefore be used. This saved a relatively significant amount of overhead time by reducing the number of times of communicating with the external MT system.

4.3 Query Adaptation Algorithms for PMIR

This section presents the query adaptation algorithms proposed in this thesis. First, the algorithm that adapts the query based on the Fragmented User Model is given, followed by the algorithm that performs selective query adaptation based on the Fragmented User Model. Afterwards, a discussion is provided regarding the changes that are applied on the algorithms in order to make them suitable for use with the Combined User Models.

In the experiments reported in Section 5.4 the algorithms were used for pre-translation query adaptation. Nonetheless, these algorithms can also be used for post-translation query adaptation, taking the query language into consideration and translating the interest terms where necessary to match the language of the query.

4.3.1 Query Adaptation based on the Fragmented User Model

This subsection presents the algorithm for performing query adaptation (expansion) based on the Fragmented User Model. To expand a query, the algorithm first identifies the fragment which corresponds to the language of the query. Second, in order to ensure that the query is expanded with terms that are relevant to it, the algorithm attempts to **identify the vector that is most relevant to the topic of the query** from within that fragment. Given that the interest vectors are not classified under labelled categories, identifying the relevant vectors becomes a challenge. One way to address this challenge is to identify vectors in which the **query terms** appear. However, this is not sufficient on its own as it may be the case that some of the vectors are relevant to the **topic of the query** yet they do not contain the exact terms of this query (e.g. hyponyms/hypernyms). Therefore, the algorithm uses a supplementary method to help in identifying vectors that are of relevance to the query: an iteration of Pseudo-Relevance Feedback (PRF) is performed ("behind the scenes") and then the snippets of the top 10 retrieved documents are compared to the vectors of the user model. The key point about the process of identifying the degree of relevance of a user-model-vector to the query is that it is a combination of direct similarity with the query and indirect similarity with PRF documents associated with the query.

This process is carried out as follows (using *F:cs* to calculate cosine similarity as defined earlier).

Let $\vec{q}$ be the vector of weighted terms that represents the source query (i.e. the query to be expanded).
Let $lang_i$ be the language of the source query.
Let *V* be the set of vectors maintained in the user model that belong to the fragment of $lang_i$, such that $V = \{\vec{v}_1, \vec{v}_2, \ldots \vec{v}_x\}$, where $\vec{v}_j$ is a vector of weighted terms that represents a cluster of user's interests, and w_j is the overall weight of $\vec{v}_j$.
Let *D* be a set of PRF documents such that $D = \{\vec{d}_1, \vec{d}_2, \ldots \vec{d}_y\}$, where $\vec{d}_k$ is a vector of weighted terms that represents the snippet of a PRF document.

Algorithm#4: Query Expansion based on Fragmented User Model.

Input: $\vec{q}$, $lang_i$, V, D

Output: one or more terms suggested for query expansion

Steps:

1: For each $\vec{v}_j \in V$

2: **Let** $SimQ_j = cs(\vec{q}, \vec{v}_j)$

3: **Let** $SimD_j = \left(\frac{\sum_{k=1}^{y} cs(\vec{d}_k, \vec{v}_j)}{y}\right) \cdot w_j$

4: Let $SimQ_{tot} = \sum_{j=1}^{x} SimQ_j$

5: Let $SimD_{tot} = \sum_{j=1}^{x} SimD_j$

6: For each $\vec{v}_j \in V$

7: **Let** $SimT_j = \alpha \left(\frac{SimQ_j}{SimQ_{tot}}\right) + (1-\alpha)\left(\frac{SimD_j}{SimD_{tot}}\right)$

where α in [0..1] $\in \mathbb{R}$ is a constant that controls the influence of query similarity over document similarity.

8: Select the user model vector that received the highest *SimT* score and expand the query using *n* terms from that vector.

Narrative: for each vector in the user model (within the designated fragment): the cosine similarity between the vector and the query is computed as *SimQ*. Then, a round of PRF is performed using the source query. The sum of cosine similarities between the vector and the snippet of each PRF document is computed and then averaged over the number of snippets. This averaged sum is then multiplied by the overall weight of the user model vector to obtain *SimD* (which represents the similarity between a user model vector and all the snippets of the PRF documents). The two similarity scores (*SimQ* and *SimD*) are normalised and combined together such that *SimT* represents a final total score for the user model vector. After *SimT* is computed for each vector, the vector that receives the highest score is identified and the query is then expanded using *n* terms from that vector[1]. Figure 14 shows an illustration of the scoring operation of the algorithm.

[1] The actual number of terms to use to expand the query, and the value of any thresholds used in the algorithms, are specified in the experimental setup.

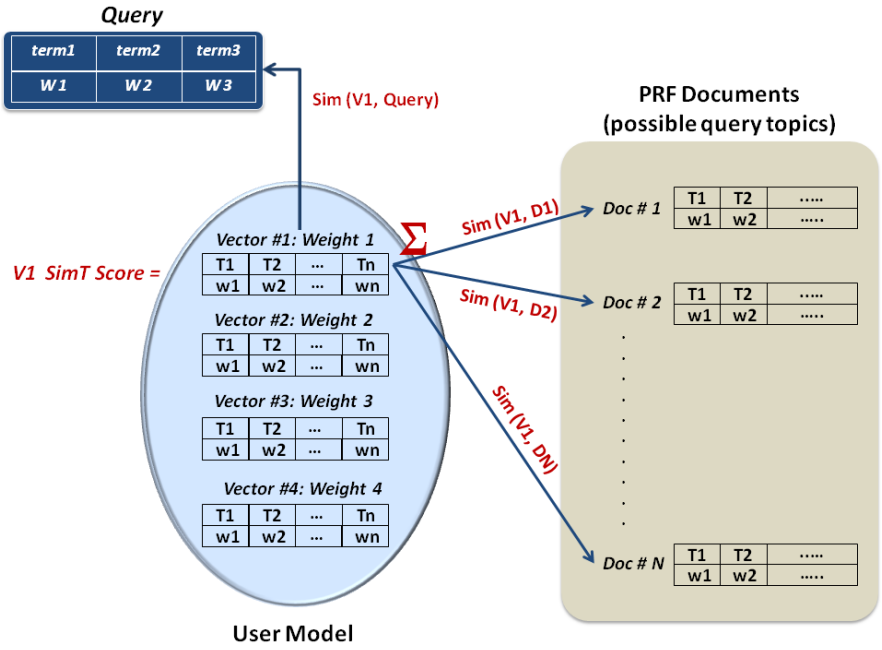

Figure 14: scoring a vector in the user model against the query and against PRF documents

4.3.2 Selective Query Adaptation based on the Fragmented User Model

This algorithm is an extension of the previous algorithm. In this algorithm, an additional step is taken to ensure that the query is expanded by terms from a user model vector **only if that vector's similarity score (*SimT*) exceeds a certain threshold**. In other words, this algorithm dynamically bases the decision of whether or not to personalise the query on evidence exhibited in the user model; a query is only expanded if there is enough evidence that indicates that the user has shown previous interest in the **topic** of that query. Accordingly, the previous algorithm is amended at **step#8** as follows:

Let t be a constant that represents an arbitrary minimum similarity threshold where t in $[0..1] \in \mathbb{R}$.

Algorithm#5 (amendment of Algorithm#4): **Selective Query Expansion based on Fragmented User Model.**
Input: $\vec{q}$, $lang_i$, V, D, t, $SimT$
Output: zero or more terms suggested for query expansion
Steps: **8: Select** the user model vector that received the highest *SimT* score **9: if** $SimT > t$ **10: then** expand the query using n terms from that vector **11: otherwise** do not attempt to expand query

Narrative: after *SimT* is computed for each user model vector, the vector that receives the highest score is identified. If that vector's score exceeds the minimum similarity threshold then it is used to expand the query. Otherwise, the query is not expanded; a low similarity score for the identified vector indicates that the vector may not be relevant to the query's topic, and that it may therefore degrade retrieval effectiveness if it is used to expand the query –thus, resulting in user dissatisfaction.

4.3.3 Adapting Queries based on the Combined User Models

The aforementioned query adaptation and selective query adaptation algorithms were also used in conjunction with the Combined User Model representations (both the *Early* and the *Late* models). In order to do this, minor changes were applied to the algorithms:

- Instead of operating on a fragment of the user model, the algorithms operate on the whole user model[1].
- The interest terms maintained in the Combined model are all in the preferred language. So, if the query to be expanded is not in the preferred language then all the terms of the user model are translated to the query's language *a priori*[2].

4.4 Result Adaptation Algorithms for PMIR

4.4.1 Result Adaptation based on the Fragmented User Model

Following the retrieval of multiple result lists by the Multilingual Information Retrieval Component (e.g. English, French, and German result lists), the results in each list are assigned scores based on the cosine similarity between the snippets and the interest vectors of the corresponding language-fragment in the user model (i.e. each results in the French list is scored against the vectors of the French fragment of the user model, the German results against the German fragment, and so on). The scoring process for each result list takes place as follows (using *F:cs* as defined earlier).

[1] The combined user model can be regarded as a special case of the Fragmented User Model: a model that is made up of a single fragment.

[2] The application of a translation step to the whole user model might negatively affect the process due to translation inaccuracies. However, in the PMIR experiment reported in Chapter 5, the source queries were already in the users' preferred languages, so no translation was needed.

Let $lang_i$ be the language of the result list (which is the language of each result in the list).

Let R be the set of results in the list such that $R = \{\vec{r}_1, \vec{r}_2, \ldots \vec{r}_x\}$, where $\vec{r}_j$ is a vector of TF-weighted terms that represents the snippet of the result.

Let V be the set of vectors maintained in the user model under language $lang_i$, such that $V = \{\vec{v}_1, \vec{v}_2, \ldots \vec{v}_y\}$, where $\vec{v}_k$ is a vector of weighted terms that represents a cluster of interests within $lang_i$, and w_k is the overall weight of $\vec{v}_k$.

Algorithm#6: Result Scoring based on Fragmented User Model.
Input: R, V, $lang_i$
Output: Scored results
Steps: **1: For** each $\vec{r}_j \in R$ **2: For** each $\vec{v}_k \in V$ **3: Let** $SimR_j = \frac{\sum_{k=1}^{y}(cs(\vec{r}_j, \vec{v}_k) \cdot w_k)}{y}$

Narrative: for **each result**, the **cosine similarity** is computed between the result's snippet and **each user model vector** in the designated fragment, where each similarity score is multiplied by the overall weight of the user model vector. This produces multiple scores per result. The computed scores for a result are then summed up and **averaged over the number of vectors** in the user model. Figure 15 shows an illustration of the algorithm.

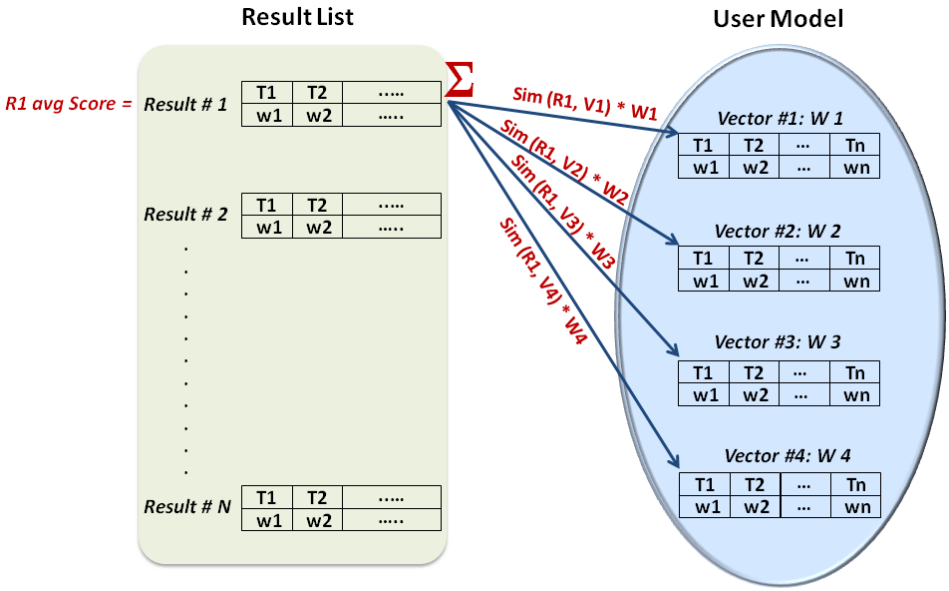

Figure 15: scoring a result against the vectors of the user model

After the scoring algorithm is applied to each result list, the lists are **processed to be merged** into a single list in either one of the following ways:

1. *Score-based merging:* the results are **merged solely based on the similarity score** that each result was assigned by the algorithm above. This essentially means that all the results are **put together** in one list and then sorted in descending order of scores. This process can be deemed as **"altogether result re-ranking"**.
2. *Separate re-ranking with round robin merging:* the results **within each list** are **re-ranked on their own** by sorting them amongst each other in descending order of the assigned similarity scores. Then, the lists are **merged** into a single list using the **round robin scheme**.

An argument that might be raised against the first way (i.e. the notion of altogether re-ranking) is that the process does not take the following actuality into consideration: the scores assigned to results of a certain language are not directly comparable to the scores assigned to the results of another language because the results are not scored against the same vectors in the user model; for example, if a French result received a score of "0.8" and a German result received a score of "0.4" then this does not mean that the French result is, quantitatively, twice as relevant as the German result. However, the process should actually be viewed from a different perspective: it can be regarded as a confidence-based voting process where each fragment in the user model has a voting power depending on how much evidence it has about the user's degree of interest in the current search topic. To this effect, the produced similarity scores can be partially regarded as values that indicate how far a fragment is confident that the current search topic (thereby its associated results) belongs to its "jurisdiction". So, based on this perspective, a French result receiving a high score of "0.8" is an indication that the topic exemplified by that result was exhibited in the French side of the user's interests and therefore this French result should appear at a higher rank in the final result list presented to the user (i.e. the specific score value is not transferable between lists but the relative rank in the list is). This perspective is in line with the underpinning assumption of the Fragmented User Model stated earlier: *Users exhibit different search interests in different languages*.

Concerning the round robin merging operation, before the operation is applied, the lists themselves are sorted according to a pre-determined language priority arrangement. In other words, each whole list is treated as one block where a certain priority scheme determines which block should be the first block to start with when carrying out the round robin operation. In the experiments reported in Section 5.4 this language-priority scheme was as follows:

1. The language of the query.
2. The Preferred language, if it is different from the language of the query.
3. English, if it was not one of the above.
4. Any remaining language(s); the order therein is enforced by the order of their appearance in the configuration file (in the property that specifies the list of languages on which the framework operates).

After the initial block-sorting takes place, the merged list is created by carrying out a standard round robin operation: inserting the first result from the first list, followed by the first result from the second list, then the first result from the third list, then back to the second result from the first list, and so on.

4.4.2 Result Adaptation based on the Combined User Models

The result scoring algorithm (Algorithm#6) discussed in the previous subsection is also used in conjunction with the Combined User Model representations (both the *early* and the *late* models). In order for that to take place, the following changes are applied:

- Result snippets that are not in the preferred language are translated to the preferred language *a priori*; where available, the translated version of a snippet was obtained from the search logs instead of translating it.
- Instead of operating on a fragment of the user model, the algorithm operates on the whole user model. Therefore, all results are scored against the same vectors in the user model.

The **score-based merging** approach discussed in the previous subsection is used to interleave the results into a single list. This essentially means that the results were all re-ranked based on their similarity scores with the user model. The associated argument presented in the previous subsection is no longer applicable to this situation because the results were all scored against the same user model vectors. Thus the similarity scores are directly comparable.

4.5 Implementation Summary

This chapter discussed the implementation details of the various components that make up the PMIR system. Furthermore the chapter provided the implementation of the proposed algorithms for: user modelling, query adaptation, selective query adaptation, and result adaptation.

The user modelling approaches discussed in this chapter partially addressed challenge#3 of this thesis. The evaluation carried out in Chapter 5 will serve to show that these approaches successfully lead to improvements in PMIR.

Challenge #3: *Can the use of user models that encompass the aspect of multilinguality improve retrieval effectiveness in PMIR?*

The adaptation algorithms discussed in this chapter contributed to addressing challenge#4 of the thesis:

Challenge #4: *How should query adaptation and result adaptation algorithms be extended in order to incorporate the aspect of multilinguality?*

Chapter 5: Evaluation

This chapter discusses the quantitative and qualitative evaluation carried out for the thesis. The discussions include: objectives, experimental setup, results, analysis of the findings, limitations and caveats, and conclusions. Given that, in PMIR, either the users or the content can be multilingual, the goal of the evaluation is to study the effect of multilinguality on: (1) how users interact with the content; (2) how multilingual user models and adaptation algorithms can improve the way they search for and gain access to that content; and (3) how they perceive a multilingual search service.

This chapter is organised as follows:

- Section 5.1 discusses an initial experiment which involves an industry case-study for multilingual search. The objective of the experiment is to demonstrate the need for and the usefulness of MIR in realistic customer support scenarios.
- Section 5.2 discusses an investigation carried out on a dataset of multilingual search logs. The objective of the investigation is to analyse the search behaviour of users from different linguistic or cultural backgrounds and gain insight into patterns or differences in search behaviour.
- Following from the investigation, Section 5.3 discusses an exploratory experiment that demonstrates the efficacy of incorporating the attribute of language (language of the user and language of the corpus) in the process of personalising search results.
- Based on the lessons learnt from the exploratory experiments and investigation, Section 5.4 discusses a set of experiments for PMIR evaluation. The experiments are conducted using the PMIR framework proposed in this thesis. The objective of this set of experiments is twofold: (a) to evaluate the effectiveness of the multilingual personalisation algorithms proposed in this thesis (in conjunction with the proposed multilingual representations of user models); and (b) to demonstrate how the framework enables the individual and combined evaluation of PMIR components.
- Finally, Section 5.5 discusses qualitative evaluation of the usability of the PMIR system and of the users' perception of the translation quality of the multilingual search results presented to them.

5.1 Evaluating the Effectiveness of Multilingual Search in a Realistic Scenario: an Industry Case Study

Enterprises which provide solutions or online services to international customers often have to provide their customer support content in multiple languages. The most common way to make this content available is through a search interface on their websites. It requires a great deal of time, effort, and resources to prepare, localise, publish, and manage this content. This is a highly costly process that sets itself as an important item in the budget of major enterprises (Ryan et al., 2009, van Genabith, 2009). This cost, in turn, may be reflected in the actual price of the product or service that the enterprise offers to its customers.

Due to the global popularity of the English language, and the wealth of content authored in English on the Web, it is often the case that enterprises originally author their customer support documents in English and then carry out the translation process to make the documents available in other languages. With the advent of machine translation, multilingual search, and digital content management, the cost of this process can be greatly reduced. The success of MIR has been demonstrated in many research studies, yet it is still the case that many major enterprises have not employed it in their customer support websites. Therefore, it is important to carry out studies that demonstrate the usefulness of MIR in realistic customer support scenarios.

This experiment aims to evaluate the retrieval effectiveness of multilingual search in providing relevant results to a query in the absence of customer support content in the language of that query. This is the case when an enterprise wants to provide support in a language to which they have not translated any content (because of scarce resources, cost-benefit analysis, etc.). The experiment involves searching across content in English compared to searching across content in three languages: Polish, Turkish, and German.

The customer care scenario investigated in this experiment is concerned with a multilingual user; specifically, a user who understands English in addition to his/her native language. Therefore, the system is only required to translate the source query, and not the returned results. This means that the key factors under investigation in this experiment are the accuracy of query translation and the availability of content in different languages.

5.1.1 Objectives

The objectives of this experiment are:

- To investigate if it is possible for enterprises to extend their customer support to languages where they have not carried out any content translation[1].
- To investigate if this can be done at an acceptable level of quality (i.e. how effective this process is in terms of the relevance of search results).

The main question under investigation in this experiment is: *for a non-English query, is it possible to retrieve customer support content in English that is as relevant as customer support content retrieved in the query's language?*

5.1.2 Experimental Setup

This section discusses the dataset, the setup, and the evaluation metric used in the experiment.

5.1.2.1 Data

This experiment was carried out using query logs from *Office.com*[2], which is a customer support website providing a range of services concerning the Microsoft Office suite of products. These services include help articles, multimedia content, and downloadable plug-ins. The support is provided in multiple languages that cater for 67 markets[3] (world regions).

The experiments involved obtaining the top 20 queries from the search logs of each of the following three markets: Polish (PL), Turkish (TR), and German (DE). These queries were then used as test queries[4] to carry out search across the help articles provided in various languages.

[1] The examination of content quality or appropriateness from a linguistic perspective is out of the scope of this experiment.

[2] http://office.microsoft.com

[3] http://office.microsoft.com/worldwide.aspx

[4] *test queries* are often called *topics* in the IR field. However, in this thesis the term *test queries* is used in order to avoid confusion in certain discussions where the actual topic (i.e. subject) of a query or a document is being discussed.

5.1.2.2 Setup

For each test query, two lists of results were retrieved as follows:

1. **Source-language Result List:** 20 results were retrieved by submitting the source query to the corpus that corresponds to the query's language.
2. **Target-language Result List:** 20 results were retrieved by translating the source query to English and then submitting it to the English corpus. The translation of the queries was carried out using the Bing Translator API.

The evaluation involved comparing the retrieval effectiveness of the two lists to each other.

It is worth noting here that for English and German retrieval on Office.com, the search engine performs federated search where the first 10 results are retrieved from the customer support corpus, and then any additional results requested after the 10th position are retrieved from the Web. Accordingly, the evaluation is reported separately for the 10th position of the result-list (i.e. involving documents from the help articles only) and the 20th position of the result-list (i.e. involving documents from both the help articles and from the open Web).

All the results were subject to manual relevance judgment where each result was assessed on a 4-point scale: "Bad", "Fair", "Good", or "Excellent". Each result was judged by three experts in the domain and then the median of the three judgments was taken as the final judgment for the result.

The Mean Average Precision (MAP) metric was used to evaluate the retrieval effectiveness of the result lists as it rewards lists where relevant documents appear at higher positions. Since MAP operates on binary relevance judgments, the 4-point-scale judgments were converted to 2-point-scale by taking the higher two judgments as *Relevant* and the lower two judgments as *Irrelevant*[1].

[1] The reason for using a 4-point-scale in the first place was to support other forms of evaluation in case they become viable in the future.

5.1.3 Experimental Results

The following figures show the retrieval effectiveness of the Target-language result lists in terms of how far they achieved of the corresponding Source-language result lists. The percentages are shown for MAP@10 (Figure 16) and MAP@20 (Figure 17).

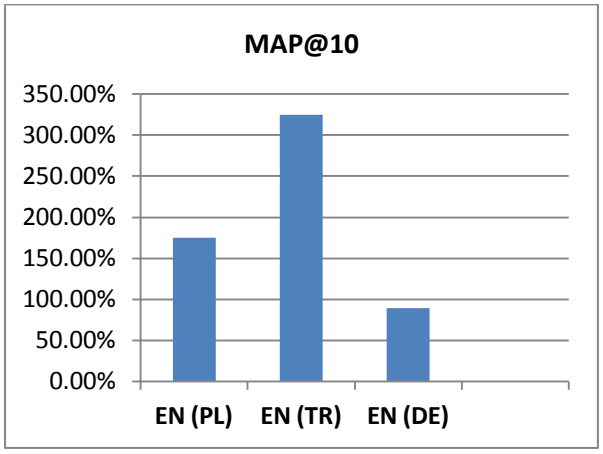

Figure 16: percentage of achievement of English lists over corresponding Polish, Turkish, and German lists for MAP@10

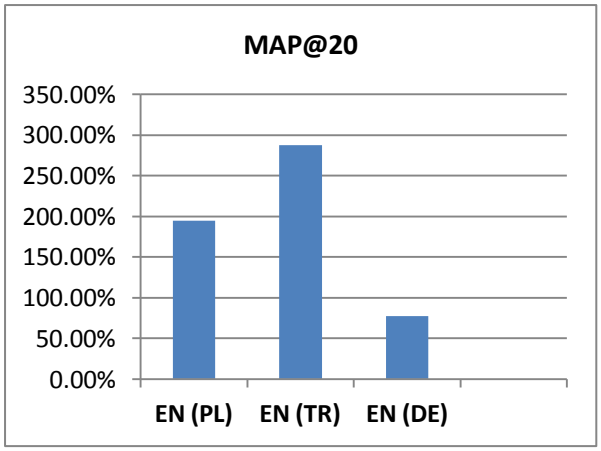

Figure 17: percentage of achievement of English lists over corresponding Polish, Turkish, and German lists for MAP@20

The evaluation shows that, for the German queries, the target list was able to achieve up to 89% of the effectiveness of the source list (MAP@10), indicating that the retrieved cross-language content was almost as good as the retrieved content in the same language of the query. For the Polish and Turkish queries, the target lists actually performed better[1] than the corresponding source lists.

[1] Statistically significant using the 2-tailed T-test with p=0.05

A manual examination of the translated queries revealed that the MT system performed better in translating the Polish and Turkish queries compared to translating the German queries. This may be the reason behind the observed difference in effectiveness between the source and target lists for the German queries.

5.1.4 Analysis of Findings

An important factor to consider regarding the experimental results –one that potentially stands out as another reason for the observed differences between languages– is the number of help articles available in, or has been translated to, each language. In this experiment, the number of help articles available in each language was not the same. The Microsoft Office help articles are originally authored in English; thus, are fully available in the English search. A large fraction, but not all, of the articles are selected to be manually translated to other languages beforehand; thus, the number of available help articles varies between languages. It might therefore seem that the comparison between the MIR system (represented by the target list) and the baseline system (represented by the source list) is not a fair comparison. However, that being said, the aim of the comparison is not to show improvements over the baseline, but rather to answer the question of: if an enterprise does not make content available in a language, would they still be able to provide adequate customer support using documents from another language? The experimental results suggest that this is indeed possible.

Another factor to consider regarding the experiment is whether or not the retrieval of customer support content from a target language partially involves retrieving open Web content in that language (as is the case with the English and German searches, where the results from the first position in the list till the tenth position are made up of in-house help articles and the remaining results are retrieved from the Web). The choice of a certain target language for MIR scenarios may depend on the amount of useful user-generated content available in that language on the Web, which in turn, affects the effectiveness of MIR systems.

5.1.5 Limitations and Caveats

The number of queries to test each system was chosen to be 20 because the experiment required manual relevance judgments. It was not feasible (in terms of availability of human resources) to experiment with a larger number of test queries per language. Therefore, the experimental results might not be regarded as very reliable. However, the role of this experiment in the overall objectives of the thesis is to serve as an exploratory study to investigate MIR in a

realistic scenario. To this end, the experiment succeeded in showing that there is a need for MIR in the industry and that it can be of benefit to multilingual users.

As discussed earlier, a limitation in the experimental setup is that there was no control over the number of documents in the corpus of each language. However, considering this situation from a different perspective, this situation is actually more similar to the case of conducting multilingual **Web search**: the number of relevant pages in various languages to a user's search is not exactly known. Therefore, the findings of this exploratory experiment were considered to be in line with the investigation plan of the PMIR framework proposed in this thesis (which involved configuring the framework to deliver a **Web search** service).

5.1.6 Conclusions

This experiment contributed to emphasise the need for, and the usefulness of, multilingual search in an industry case study. For enterprises, an important aspect to bear in mind about the customer support scenario addressed in this MIR experiment is that it caters for a multilingual user; thus the system only needed to translate the queries. The success demonstrated by the experiment for this scenario does not entirely eliminate the need to carry out content translation because a subset of the users can be monolingual. That having been said, what this experiment demonstrates is: if due to lack of resources or time an enterprise is not able to fully provide support to all customers in a certain market, it can fall back on MIR techniques to cater for a subset of the customers in that market.

5.2 Studying Users' Search Behaviour in light of Multilinguality

The next logical step, after the MIR side of PMIR had been investigated, was to explore users' search behaviour and analyse it with respect to multilinguality. This helped to gain insight into the aspects of content multilinguality and user multilinguality and how their relationship affects the way users behave in search. The outcome of this investigation served in guiding the personalisation approaches proposed in this thesis.

This investigation was carried out on search logs of *The European Library*[1], which is a website that is a prominent portal for searching across the content of many European national libraries in various languages. The logs were obtained as part of participating[2] in the LADS task (*Log Analysis for Digital Societies*) of the LogCLEF[3] track (Mandl et al., 2010) at the CLEF 2009 campaign[4] (formerly known as *Cross Language Evaluation Forum*, and now known as *Conference and Labs of the Evaluation Forum*).

5.2.1 Objectives

The analysis of the search logs was carried out with the following objectives in mind:

- To investigate how users from different linguistic or cultural backgrounds behave in search.
- To identify patterns of language-dependent search behaviour:
    - that may serve as directives for the personalisation strategy for each language or group of languages.
    - that may help in stereotypical grouping of users.
- To elicit user-specific attributes of multilingual-search users.

5.2.2 Description of Dataset and Pre-processing Operations

The logs of The European Library (TEL) comprised entries for different types of user interactions with the portal (hereafter, *actions*). The logs were collected between January 2007 and June 2008. Figure 18 shows a screen capture of the TEL website[5].

[1] TEL: http://theeuropeanlibrary.org

[2] This was part of a collaborative participation of researchers from Trinity College Dublin and Dublin City University, within CNGL (Centre for Next Generation Localisation).

[3] http://www.uni-hildesheim.de/logclef/LogCLEF2009.html

[4] http://www.clef-initiative.eu (used to be: http://www.clef-campaign.org/).

[5] This screen capture was taken in the year 2009. The website has undergone many changes since then.

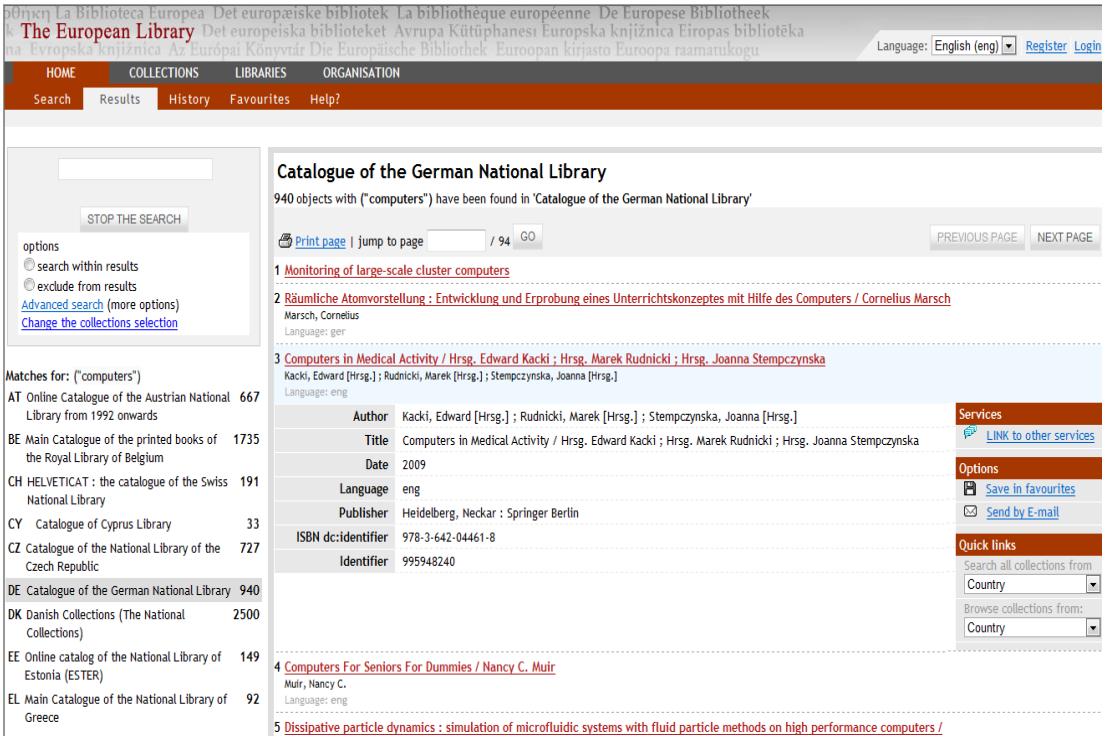

Figure 18: screen capture of TEL website (in 2009)

A log entry is created corresponding to every user action. The log entry contains the type of action performed, together with attributes such as the interface language, query, and timestamp. The experiments focused on the following attributes: *lang* (interface language selected by the user), *action*, and *query*. For the study of actions, the following six actions were considered, as they exhibited a high frequency in the logs:

1. **search_sim:** simple search, using a single text box.
2. **search_adv:** advanced search by the specific fields of title, creator (i.e. author, composer, etc.), subject, type (e.g. text, image, etc.), language, ISBN, or ISSN.
3. **view_brief:** clicking on a collection (i.e. the corpus of a library) to view its list of results in brief (where only the title of the result is displayed, and is sometimes accompanied by the authors and the language of the result).
4. **view_full:** clicking on a title link in the list of brief results to expand it.
5. **col_set_theme:** specifying a certain collection to search within.
6. **col_set_theme_country:** specifying multiple collections for searching or browsing.

Note: the dataset did not provide records of the results that the users viewed.

The initial examination of the dataset revealed that the data had to be pre-processed to address a number of issues including character encodings, syntactically malformed queries (missing quotation marks, additional parentheses, etc.), and actions/attribute-values that were not described in the LADS task guidelines. For this analysis, the following were deleted from the dataset:

- Entries with unrecorded session IDs (empty or null value).
- Search attempts having empty queries.
- Sessions with missing actions.
- Sessions with unrecorded or malformed language acronyms.

The original number of records in the dataset was 1,866,330 records, which was reduced to 1,632,044 after the cleaning process (i.e. approximately 12.6% of the records were deleted). Furthermore, inconsistencies in the format of the stored queries were dealt with, such as trimming unnecessary brackets, quotations, and white-space. In addition, query terms were extracted and stored in a separate table for performing term-based statistical analysis.

A key part of the pre-processing operation was the reconstruction of user sessions. The log entries contained anonymised user IDs and abbreviated IP addresses of the computers used to access the TEL website as well as session IDs. In addition, there was a timestamp attached to each logged action. Given that an IP address is not sufficient to distinguish between single users and a user ID may be associated with a guest account, session reconstruction was solely based on the session IDs. The session ID was used to reconstruct the actions in single sessions and then the timestamp was used to sort the actions in chronological order. Session duration was calculated as the time interval between the timestamp of the first action and the timestamp of last action in the session.

5.2.3 Analysis: Descriptive Statistics

Table 8 and Table 9 present descriptive statistics of the dataset. The logs exhibited outliers, such as the existence of sessions with either a very large number of actions or a single action (max: 1,093; min: 1) and sessions with very long or short durations (max: 116 days; min: 1 second). This affected the averages reported in Table 9.

Table 8: frequencies

Item	Frequency
Actions by guests	1,619,587
Actions by signed-in users	12,457
Queries by guests	456,816
Queries by signed-in users	2,973
Sessions	194,627
User IDs	690

Table 9: central tendencies

Item	Average	Median
Actions per session	8.39	4
Queries per session	2.81	2
Session duration (hh:mm:ss)	00:17:20	00:01:35

It was observed that relatively few actions were performed by signed-in users (0.76%) compared to guests (99.34%). Moreover, the dataset was found to contain only 690 distinct user IDs. This may indicate that the system did not motivate users to sign-in or that users found it easier, and/or perhaps more secure, not to register with an online search system when they did not perceive any added benefits of signing in. This hinders user-focused personalisation when session reconstruction is coarse-grained.

In the analysis for this investigation, user actions were classified into four broad categories:

1. **Search:** query actions.
2. **Browse:** browsing/navigating result pages of TEL, excluding the following of links leading to external websites.
3. **Collection:** actions involving limiting the search scope by the selection of a collection, theme, or subject.
4. **Other:** any actions other than the above.

Figure 19 shows the distribution of actions along the categories. A noticeable amount of user actions (11%) was performed before attempting the search; for example, specifying certain collections or subjects to search within. This indicates a diversity of user preferences where users seek to customise their search environment/session according to their needs. Taking these preferences into consideration when modelling the users may help in improving personalisation, particularly if they are repeated, which seems likely where different settings give different language results.

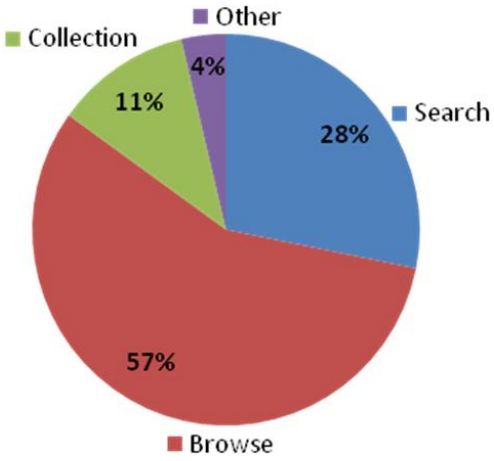

Figure 19: broad classification of TEL actions

Another tendency in user actions was noted regarding the choice of collections for search; the pre-selection of a single collection occurred considerably more frequently than the pre-selection of multiple collections (col_set_theme represented 7.13% of the total actions while col_set_theme_country represented 2.72%). This suggested that users who sought to limit their search tended to be very specific in selecting a designated collection. Two reasons may be behind this kind of behaviour:

1. Either: users' choice of collection was based on previous experience with searching in the TEL website, where they may have found that certain collections were more likely to satisfy their information needs. Because the majority of logged actions were not associated with any user identifiers, it was not possible to investigate this possibility any further.
2. Or: users' choice of collection was based specifically on the language or country associated with that collection. This called for further investigation in order to verify this assumption. This formed the basis of the experiment reported in Section 5.3 where the collections were re-ranked based on language.

5.2.4 Analysis: Interface Language and Actions

This part of the investigation was concerned with the relation between language and search behaviour. A number of variables were studied across the interface languages selected by users of the TEL website. The actions were distributed across 30 languages. The investigation focuses on the top five languages in terms of the number of associated actions. The top language was English (86.47% of the actions), followed by French (3.44%), Polish (2.17%), German (1.48%), and Italian (1.39%). It is worth noting here that the interface language does not necessarily imply the language of the query. A possible cause for the bias towards English, aside from its inherent popularity, is that it is the default interface language of the website.

Therefore, it may be the case that many non-native English speakers were not aware of the existence of the interface-language selection feature, and were sufficiently familiar with English to use it for browsing and navigating the website. This assumption was supported by the observation that non-English queries existed in association with actions that were logged under the English language[1]. Due to this strong bias towards English in the logs, it was decided to not include English (as an interface language, not as a query language) in further comparative discussions against other interface languages. Nevertheless, for the sake of completeness, subsequent tables/figures will report its associated frequencies and percentages.

An important usability lesson to learn from this observation is that such situation can be avoided by having the system automatically set the interface and/or querying language according to a language attribute in the user model or according to the client's IP address. This is an initial step in the process of adapting a service (or parts of a service) to the user's language preference.

Table 10 states the average and median for the number of actions and queries per session, and Figure 20 shows the frequency distribution of the six main actions across each of the five interface languages. It was observed that the distribution and the number (averages/medians) of actions varied across languages. More notable differences were exhibited for Italian in comparison with other languages. For example, the percentage of actions of viewing full records for Italian (28.39%) was higher than French, Polish, and German (23.55%, 21.95%, and 23.53% respectively). Furthermore, for Italian, it was observed that the ratio between the number of queries submitted through simple search and those submitted through advanced search was 2.34, while the ratio for French, Polish, and German were notably higher (3.2, 3.59, and 3.42 respectively). It was also observed that Polish users seemed to have a higher rate than others in using the feature of specifying a single collection before attempting the search (13.58% of the actions), compared to French, German, and Italian (10.86%, 9.46%, and 9.35% respectively)[2].

These kinds of observations of the querying and browsing behaviour of users from different linguistic backgrounds stand out as a lesson for search systems wishing to operate in the multilingual dimension: studying these differences and taking this kind of knowledge on board

[1] *Caveat:* however, a possibility that cannot be ruled out entirely is a native English user searching for a document using its original non-English title

[2] *Caveat:* a possibility that cannot be ruled out is that such observations may have been governed by the amount of available document collections in each language.

when developing a system may provide directives to the personalisation strategies that should be undertaken for a certain language or for a group of languages.

Table 10: interface language statistics

Interface Language	Number of actions per session		Number of queries per session	
	Average	Median	Average	Median
English	7.97	4	2.7	2
French	9.2	5	3.01	2
Polish	8.63	5	3.14	2
German	9.37	5	3.03	2
Italian	11.3	6	3.73	2

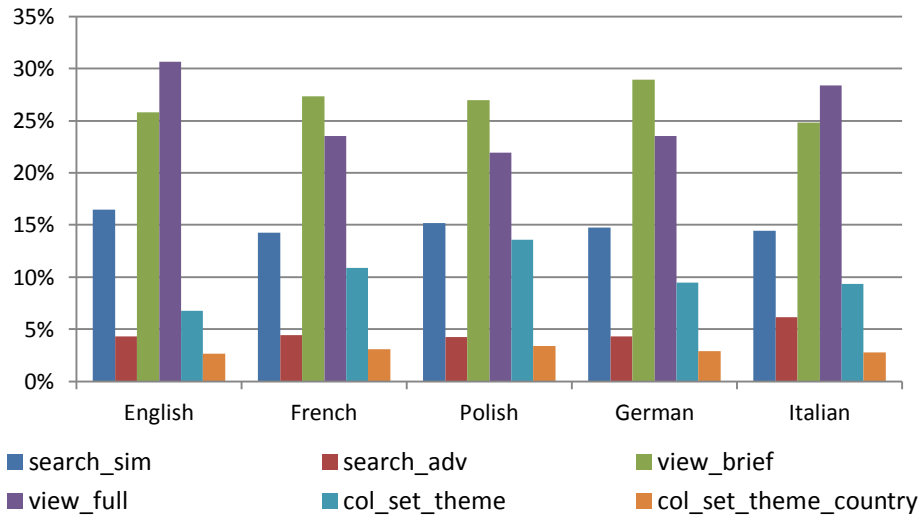

Figure 20: distributions of actions across languages

5.2.5 Analysis: Frequencies and Categories of Search Terms

This part of the investigation involved studying search_sim actions (simple search) with respect to the number of search terms per query and the most queried terms. It was found that the percentage of queries made up of three terms or less in the whole dataset was 83.12%. Table 11 shows the average and median of the number of terms per query across interface languages. The trend that can be seen in the table is that users only use a small number of search terms when querying the system. This contributes to higher query ambiguity, and therefore allows room for using multilingual query expansion algorithms to help disambiguate the queries.

It was observed that German users exhibited the lowest average with respect to the number of terms per query. Moreover, part of the analysis revealed that German exhibited the largest distribution of queries made up of just one term. This may be because the German language

allows noun compounds without spaces. This observation suggests that language-specific characteristics ought to be taken into consideration when developing multilingual query adaptation algorithms.

Table 11: number of search terms per query

Interface Language	Terms per query	
	Average	Median
English	2.38	2
French	2.09	1
Polish	1.89	1
German	1.77	1
Italian	2.09	2

As part of the analysis, the top 20 occurring search terms for each interface language were identified (excluding stopwords). Furthermore, the terms were manually divided into five categories:

1. ***Creator*:** author, composer, artist, and so on.
2. ***Location*:** cities, countries, and so on.
3. ***Subject*:** topics, as per the *Dewey Decimal Classification*[1].
4. ***Title*:** names of books or other types of artwork, including proper nouns and common nouns.
5. ***Type*:** the kind of document, such as: text, image, sound, and so on.

These categories were based on the fields of the advanced search in the TEL website, except for *location* which it made sense to add when several proper nouns denoting places were encountered in the top terms.

For simple search, it was observed that most of the search terms came under the *creator* and *title* categories (30% and 28% respectively). The same trend was exhibited for advanced search, though with a higher inclination towards the *creator* category (45% and 21%). For library search systems, where a document's author is of particular significance, this information could be useful to the query adaptation process.

Figure 21 shows the category distribution of the top 20 search terms across interface languages in simple search. Considerable behavioural difference was observed between languages. For example: for French, 20% of the terms were *subjects* and 25% were *creators*, while for Italian only 5% of the terms were *subjects* and 40% were *creators*. These findings emphasise that there are behavioural differences between users from different linguistic backgrounds. Therefore, for

[1] DDC is a widely used library classification system: http://www.oclc.org/dewey.en.html

a system to provide a successful multilingual service, these differences have to be taken into consideration if users are to be grouped under stereotypes. This information could become useful when designing adaptation strategies for the different languages that the system intends to support.

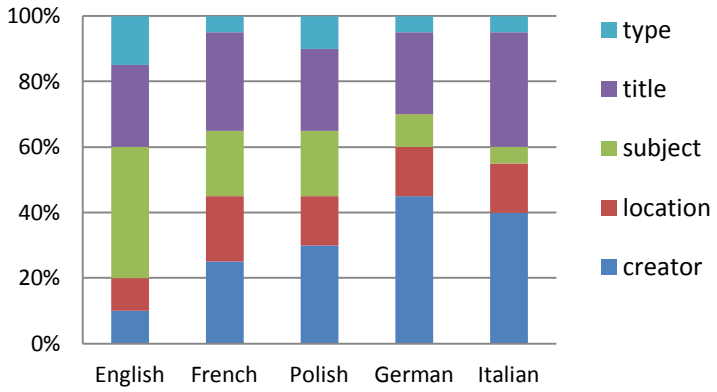

Figure 21: distribution of categories across languages in simple search

5.2.6 Analysis: Sequential Patterns in Users Actions

This part of the investigation involved studying sequential patterns in user actions. A number of selected patterns that commonly appeared in the logs are shown in the tables below. Table 12 presents common patterns of sequences of two actions and Table 13 presents common patterns of sequences of three actions. The first row in each table shows the most frequently occurring pattern; the remaining rows show a selected set of frequently occurring patterns which exhibited interesting behaviour. Related patterns are grouped together in the tables.

An intuitive sequence of user actions after submitting a query would be a *view_brief* action followed by a *view_full* action (i.e. it would be expected that a user would first click on one of the collections that appear on the left side of the screen to display its associated list of brief results on the right side of the screen, and then to click on one of the result links on the right to expand it and view its full details). However, a counter-intuitive behaviour was observed based on the pattern analysis: after performing a search action, users showed a relatively higher tendency to directly perform a *view_full* action (i.e. jump to clicking on a result from the right side of the screen to expand it). This indicates that the more users were satisfied with the collection that was already highlighted on the left[1] and went directly to its associated results displayed on the right (see screen capture in Figure 18). This may be attributed to one of two

[1] When multiple collections were being searched, the TEL website displayed them on the left in alphabetical order of country acronyms (see screen-capture figure) and automatically highlighted the top most collection. For the highlighted collection, the associated list of brief results was displayed on the right.

reasons: (1) either those users always found what they were searching for in the first collection (which is not likely); or (2) more users pre-specified their desired collection before attempting the search (maybe voluntarily out of preference, or because they needed to in order to avoid browsing a long list of collections). Based on the observations discussed in the previous subsections, the second reason is more likely to be the cause of the observed pattern.

Although studying this kind of behavioural pattern may seem specific to the TEL website, or perhaps to library search in general, it may be of benefit to search engines which operate in a similar method. For example, some search engines, such as *Yippy*[1] (formerly known as *Clusty*), do not just display a single list of results to their users (as is typically done by many search engines), but rather organise the results under several clusters or categories based on the different word senses or domains exhibited in the results; in which case, the clusters are displayed at a certain part of the screen and then when the user clicks on them, their associated list of results are displayed. This kind of search interface may benefit a lot from re-ranking the clusters/collections, as well as the results within, according to the user's preferences.

Another interesting observed pattern suggested that users got confused between two interface features (both provided as drop down menus) that were provided on TEL's main page:

1. col_set_theme: specifying a single collection to search.
2. col_set_theme_country: specifying multiple collections to search or browse, which redirected the user to another page that listed all the available collections in many European countries.

This was observed as user actions subsequently alternated between the two features. The observation of such patterns reflects how the study of search logs can contribute to enhancements in the search system's interface in a way that specifically contributes towards the satisfaction of multilingual users.

Table 12: selected sequential action patterns for two subsequent actions

Action 1	Action 2	Frequency
view_full	view_full	153,952
search_sim	view_full	112,562
search_sim	view_brief	86,625
search_adv	view_full	32,356
search_adv	view_brief	28,732
col_set_theme	search_sim	40,044
col_set_theme_country	search_sim	12,397

[1] http://www.yippy.com/

Table 13: selected sequential action patterns for three subsequent actions

Action 1	Action 2	Action 3	Frequency
view_full	view_full	view_full	79,346
col_set_theme	search_sim	view_full	18,562
col_set_theme	search_sim	view_brief	16,446
col_set_theme_country	search_sim	view_brief	2,530
col_set_theme_country	search_sim	view_full	8,458
col_set_theme	col_set_theme_country	col_set_theme	4,735
col_set_theme_country	col_set_theme	search_sim	3,159

5.2.7 Limitations and Caveats

Due to the unavailability of similar search logs from general Web search engines at the time of conducting this investigation, it was decided to make use of library search logs. It is understood that some of the findings of this investigation may be regarded as specific to the library search environment. However, the analysis aimed to validate the idea that multilingual search logs may exhibit behavioural patterns for multilingual search users which can be detected, studied, and employed for personalisation. This was evident, even in this limited case.

5.2.8 Conclusions

This preliminary investigation of users' search behaviour revealed that different behavioural patterns are exhibited for users from different linguistic backgrounds. The analysis argued that the identification of these patterns could be useful for the stereotypical grouping of users. The analysis also made a case that search personalisation can benefit from taking the attribute of language, and perhaps country as well, into consideration when designing the adaptation algorithms; one application of this is re-ranking document collections in multilingual search based on language and/or country, which is investigated in the experiment reported in Section 5.3.

The findings of this investigation address the second challenge of this thesis: ***Challenge #2:*** *Are there certain behavioural patterns or differences that can be observed for users in multilingual search?* To which the answer is positive. This provides a helpful indication that language can drive personalisation.

5.3 Evaluating the Effectiveness of Re-ranking Collections in Digital Library Search based on Language and Country

This exploratory experiment follows on the analysis of TEL search logs. In TEL, and generally in portals of multilingual digital archives (such as *Europeana*[1]), the user is presented with search results that are grouped under various target collections (corpora). A collection is either a library catalogue or an online resource that is associated with a certain country. At the time of conducting this experiment, TEL presented the collections to the user on the left side of the page in alphabetical order of country acronyms (two-letter acronyms). The idea of this experiment is to explore the efficacy of re-ranking the document collections based on the language and country. This involves studying three attributes that are associated with the users and the queries they submit:

1. The language of the submitted query.
2. The language of the interface (specified by the user).
3. The current country of the user (i.e. the location from which the query was submitted).

5.3.1 Objectives

The main objective of this experiment is to investigate if the user's choice of collections is influenced by country and language. A secondary objective is to investigate the relation between languages in which users query the system and the languages that are spoken in the countries in which the users reside.

5.3.2 Experimental Setup

This subsection discusses the dataset and setup of the experiment, and also discusses the re-ranking function used in the evaluation.

5.3.2.1 Data

This experiment was conducted as part of participating[2] in LogCLEF 2010[3] (Di Nunzio et al., 2011). The dataset used in the experiment was compiled from two resources:

1. Query logs of TEL during the month of February 2007 (this is a subset of the dataset used in the investigation reported in Section 5.2).

[1] http://www.europeana.eu/
[2] This was also part of collaborative participation within CNGL.
[3] http://www.uni-hildesheim.de/logclef/

2. A TEL resource, called the *HTTP logs*, which contained additional details of the search sessions, including the list of collections searched for each query and the list of collections that the users clicked for each query[1].

The final dataset contained 566 queries from various languages (the number was initially 1800 queries, but was then reduced to 566 after the processing operations explained below).

The user's country was determined from the IP address recorded in the dataset. The query language was detected using the *Google AJAX Language API* (now known as *Google Translate API*[2]), which returned a confidence level associated with the detected language of a query. Only queries with 10% confidence or higher were used in the experiments; this reduced the number of queries from 1800 to 566 queries. Figure 22 shows the distribution of the queries across languages and countries.

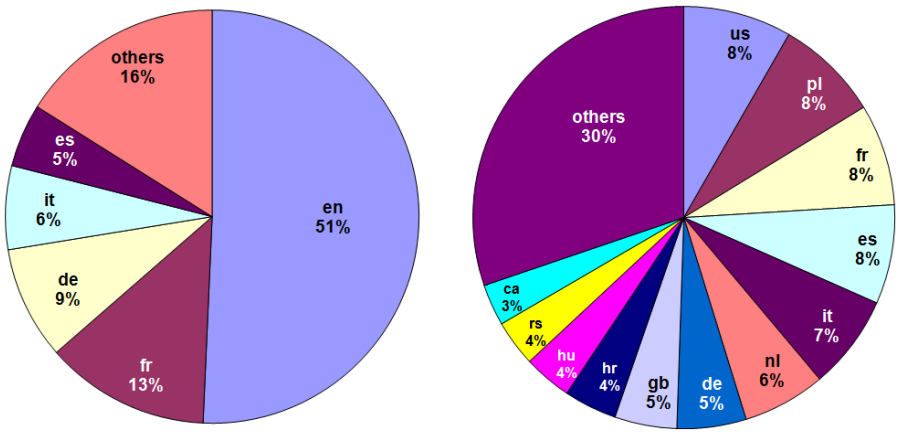

Figure 22: distributions of queries over languages (left) and countries (right)

It was noted that 50% of the queries were in English although the total percentage of queries coming from English-speaking countries (United States, United Kingdom, and Canada) was only 16%. Thus, a large fraction of the English queries came from countries that were identified as non-English-speaking countries. This was considered as an indication that the country attribute might not be as effective as the language attributes in the process of re-ranking library collections. This was further confirmed by the results of the experiment.

5.3.2.2 Setup and Re-ranking Function

The experiments involved studying the list of collections that were searched for each query, and the collections that the user clicked on. As mentioned earlier, each collection is associated with

[1] The reason for operating only on the subset of the logs corresponding to February 2007 is technical problems associated with the entries of the HTTP logs.

[2] http://developers.google.com/translate/

a country. The experiment investigated re-ranking the list of collections displayed to the user with the aim of bringing collections that match certain language and country criteria to higher ranks in the list (e.g. matching the user's country to the collection's country). In order to be able to match a collection with a query on the basis of language, a collection was mapped to one or more languages based on the official languages that are spoken in the country associated with the collection[1].

In order to evaluate retrieval effectiveness over the list of collections presented to the user, the collections that the user clicked on were used as an implicit assessment of relevance (i.e. binary relevance judgments, where the clicked collections are assumed to be the relevant ones, and non-clicked collections are assumed to be irrelevant). This follows on the method adopted in a number of studies in the literature such as (Speretta and Gauch, 2005). The MAP metric was used for evaluating retrieval effectiveness. The original ranked list of collections (i.e. the one that TEL presented to the user) was used as the baseline for evaluation. Several alternative re-ranked lists were investigated and compared to the baseline as discussed below.

A collection scoring function (i.e. a result scoring function) was used to re-rank the list of collections. Collection scores were computed as a weighted linear combination of the following three attributes (where M_x = 1 if the two items match, and M_x = 0 otherwise):

1. M_c : matching the user's country with the collection's country.
2. M_q: matching the query's language with the collection's language (i.e. matching with any of the official languages spoken in the collection's country).
3. M_i : matching the interface language with the collection's language.

Each one of the attributes is multiplied by a scalar weight (W_c , W_q , W_i respectively) to control (and test) the degree of contribution of each attribute in the function. Thus the collection scoring function is as follows:

$$Score = (M_c \cdot W_c) + (M_q \cdot W_q) + (M_i \cdot W_i)$$

After each of the collections that were displayed to the user has been assigned a score, the collections were re-ranked in descending order of the scores.

[1] *Caveat*: although a library may have documents (e.g. books) from various languages, the underlying assumption of the experiment was that a user is inclined to select a collection based on the perception that the majority of its documents will be in the language that is dominant in its country (e.g. the perception that a German library will mostly have German documents).

5.3.3 Results

Table 14 shows the MAP improvements[1] for some selected re-ranking runs with alternative combinations of weights that ranged from 0.0 to 1.0. The collection re-ranking process achieved up to 27.4% improvement in retrieval effectiveness over the baseline ranking (with weights: W_c=0.1 , W_q=0.3 , W_i=0.6 as reported in the first row of the table). The last three rows in the table report the improvements achieved by each one of the three attributes on its own (by assigning the weight of 1 to the attribute in question and assigning zero to the other two); the results show that re-ranking based on the interface language selected by the user or based on the language of the user's query achieves higher improvement than re-ranking based on the country from which the query was submitted.

Table 14: MAP improvements for alternative weight combinations

Weight			MAP Improvement Percentage
W_c	W_q	W_i	
0.1	0.3	0.6	27.4%
0.1	0.6	0.3	25.3%
0.6	0.1	0.3	23.9%
0.6	0.3	0.1	22.2%
0	*0*	**1.0**	22.2%
0	**1.0**	*0*	18.1%
1.0	*0*	*0*	3.4%

5.3.4 Analysis of Findings

The experimental results showed that the user's choice of collections is more influenced by the attributes of interface language and query language than the attribute of country. This suggests that there is opportunity for improving users' satisfaction with multilingual search if their language capabilities and preferences are taken into consideration when presenting search results to them.

The findings of this experiment were confirmed when further analysis was conducted on the dataset. The analysis revealed that only 24% of the queries coming from a country were in languages associated with that country. This is not necessarily a bias towards using English in search; when non-English queries in the dataset were studied on their own, it was observed that

[1] The improvements reported in the table are statistically significant as per the 2-tailed T-test, with p=0.05

less than 30% of those queries were in a language that matched the country from which the query was submitted.

Based on the abovementioned findings, it was decided to only operate on language attributes, and not the country attribute, in the PMIR experiments reported in Section 5.4.

5.3.5 Limitations and Caveats

Typically, experiments in the IR field involve performing the re-ranking process on the list of results itself rather than the list of collections. However, a limitation in the TEL dataset was that it only included information about the clicked collections, not the clicked results. Nevertheless, this experiment was useful to this study because one of the main characteristics of multilingual search is that results are retrieved from multiple document collections (multiple languages); the experiment offered insights into how users engage with these collections.

5.3.6 Conclusions

This experiment explored the effect of incorporating the attributes of language and country in the process of adapting search results in multilingual search. The findings of the experiment suggest that there is opportunity for improving users' satisfaction with multilingual search if their language capabilities and preferences are taken into consideration when designing the personalisation algorithms.

This experiment and the investigations carried out in Section 5.2 address the following challenge of the thesis:
Challenge #2: *Are there certain behavioural patterns or differences that can be observed for users in multilingual search?*
To this end, it was shown that such patterns and differences exist and that they can be successfully harnessed for the benefit of the users in multilingual search.

Furthermore, the outcome of this experiment (together with the outcomes of the set of experiments that are discussed in Section 5.4) contribute towards addressing the following challenges:
Challenge #3: *Can the use of user models that encompass the aspect of multilinguality improve retrieval effectiveness in PMIR?*

Challenge #4: *How should query adaptation and result adaptation algorithms be extended in order to incorporate the aspect of multilinguality?*
To this end, it was shown that extending user models and adaptation algorithms to include the attribute of language leads to improvements in PMIR.

5.4 Evaluating Various Approaches to PMIR

This section presents a set of experiments for evaluating various approaches to PMIR. An approach can be regarded as a solution suite (package) that comprises a certain user model representation along with a set of personalisation algorithms that operate in conjunction with that model. The PMIR approaches evaluated in this section can be divided into two dimensions:

1. The kind of user model on which personalisation is based: this includes the three types of user models proposed in Section 4.2:
    a. The Fragmented User Model.
    b. The Early-Combined User Model.
    c. The Late-Combined User Model.
2. How personalisation is performed:
    a. By adapting the result list(s).
    b. By adapting the query.
    c. By adapting both.

The experimental results are presented below, showing the effects of each user model approach grouped by personalisation method; thus, a separate subsection is provided for: result adaptation (subsection 5.4.3), query adaptation (subsection 5.4.4), and the combination of the two (subsection 5.4.5). Moreover, for the sake of clarity and logical progression of some of the ideas, the result adaptation subsection is presented before the query adaptation subsection; this is because the latter contains more discussions and more details to cover.

The experiments reported in this Section were conducted using the PMIR framework proposed in this thesis; the implementation of which was discussed in Section 4.1. The framework facilitated the provision and the evaluation of a multilingual Web-search service.

5.4.1 Objectives

The objectives of this set of experiments are:

- To compare the three multilingual user-model representations proposed in this thesis.
- To quantitatively evaluate the effectiveness of the proposed query adaptation and result adaptation algorithms.
- To determine how the adaptation components contribute to improving the personalisation process on their own and when combined together.
- To demonstrate how the PMIR framework facilitates conducting experiments that entail these kinds of comparative evaluations.

The underlying experimental question that drives the set of experiments is: *can retrieval effectiveness in multilingual Web search be improved by employing user models and adaptation algorithms that cater for multilinguality?*

5.4.2 Experimental Setup

The set of experiments was conducted online using the PMIR framework, which was configured to provide a multilingual Web-search service. It took place over three phases. This section starts by giving a brief outline of the three phases, and then the details for each phase are given in the following subsections.

In the first phase, users (participants) were asked to use the multilingual Web-search system to complete a number of search tasks. This was a baseline system that provided textual, non-personalised search results from three languages: English, French, and German. The system logged the submitted queries and the clicked results.

The second phase took place without user participation. In this phase, the last query submitted by the user in each task was reserved for testing. The remaining queries, along with their associated clicked results, were used to construct the user models. A pool of results was then automatically generated for each test query by submitting the query to the search system multiple times using various personalisation algorithms that adapt the query and the results.

The third phase involved the participation of the same users of the first phase. Each user was shown his/her test queries along with the associated pool of results. The users were asked to judge the degree of relevance of each result in the pool. Finally, the retrieval effectiveness of

each personalisation algorithm was evaluated according to the relevance judgments provided by the users. The evaluation shows the success of the multilingual approach to search personalisation and provides a comparison between different levels of improvement achieved by the algorithms for different groups of users.

The following subsections provide the details of the three phases of the experiment.

5.4.2.1 Phase 1: User Participation and Search Tasks

The experiment involved the participation[1] of 76 users from different countries[2] and linguistic backgrounds. The participants belonged to different age groups and had different educational backgrounds and professions. This contributed to establishing a diverse group of participants who are not biased to a particular linguistic or cultural background.

The participants were asked to specify the following information upon registering with the system:

- Their native language.
- Their preferred language; one of three options: English, French, or German.
- Their familiar languages: any additional languages they speak or understand with moderate proficiency or higher. This information was used to determine which languages the participant was not familiar with so that results in those languages get translated to the preferred language.

Participants were also asked to specify their countries of origin and their countries of residence. Although this data was not used in the personalisation process, it was useful in detecting the diversity of the participants.

At the beginning of the experiment, each user was asked to choose two search tasks (search subjects) from a list of tasks. The tasks were presented in the user's preferred language. The experiment had 11 tasks altogether. However, in order to reduce the selection burden on users, 7 out of the 11 tasks were randomly selected and displayed in the task selection screen. The 7 tasks were displayed in a random order so as to avoid bias towards tasks. The list of tasks is given in Appendix-B.

[1] This user trial conforms to ethical research conducts and was approved by the Research Ethics Committee of the School of Computer Science and Statistics at Trinity College Dublin.

[2] Because the experiment was offered online, participation was not limited to people who live in Ireland. Participants came from various countries and were resident in a variety of locations.

An important point to highlight here is that it is sometimes argued that task-based IR experiments may lack a genuine information need on the user's part; this is because information needs are provided to the users rather than the users coming up with those needs themselves. However, the task-based approach to simulating information needs was studied in (Borlund, 2000) and has shown to be a successful substitute to experimentation using user-instigated search sessions.

Moreover, in order for this experiment to facilitate (bring about) a realistic search session for the users that is as close as possible to a genuine information need, the tasks were designed so that they are neither too specific nor too vague; a too-specific task "dictates" the information need to the user and therefore leaves no room for users to search for things that are of particular interest to them; a too-vague task may confuse the users of the system and may therefore result in unrealistic or random search behaviour. The aim was to create tasks that were tangible yet allowed the users to interpret them in different ways and come up with different query intents. For example, the following is a sample of one of the tasks (titled: "*Political Event*"): "*Events, such as revolutions, protests, or military coups affect countries in many ways (politically, economically, socially, etc.). Write a few lines about such event that happened in a country in recent history and how it affected the country in which it took place (please select a country other than your country of origin or the country that you currently live in*[1]*)*".

An important design consideration that was taken into account when designing the tasks and the task selection process was to allow for personalisation by giving room for the user's personal preferences to emerge. This is demonstrated in:

1. Allowing the user to choose two tasks from a given task list; this gave users the opportunity to choose the preferred subjects from the ones offered.
2. The ability to interpret/approach the task in different ways; for example, in the sample task shown above, the users could:
    a. Decide which country to search about.
    b. Decide which type of event they would like to investigate.
    c. Decide which kind of impact they would prefer to look into (be it political, economical, social, etc.).

After selecting the tasks, the experiment asked the users to use the multilingual Web search system (shown in Section 4.1.1) to complete each task. The interface of the search system was

[1] The instruction to choose a country that is not the country of origin or the country of residence was there to drive users to actually carry out a search rather than answer the task with information that they already know.

displayed in the user's preferred language. The user was allowed to submit any number of queries s/he wished for each task. Upon submitting a query, the system returned a single merged result list containing 30 results obtained from the three languages (English, French, and German). The *round robin* merging scheme was used to merge the result lists (see details in Section 4.4.1).

Result snippets that came from a language that the user was not familiar with were translated to the preferred language. Furthermore, as discussed earlier, the documents themselves were translated on-the-fly where necessary (i.e. if a user clicked on a Web page coming from a language that they are not familiar with, the whole page was translated in a few seconds). Translation was carried out using the Bing Translator API. The system logged the queries and the clicked results for each task.

A button was displayed on the main search screen that allowed the user to indicate that s/he had finished gathering information about the task and was ready to submit the task solution. Upon clicking that button, the user was taken to a screen that asked him/her to enter a few sentences that summarised their findings with regards to the search task (i.e. what they were asked to enter was not a "solution" *per se*, but rather a *précis* of what they learned about the subject). Furthermore, the user was shown the last query s/he submitted for the search task and was asked to enter two pieces of information about it:

1. Query Description: a short description of the intent behind the query.
2. Narrative: a few lines to generally indicate what kind of content they would consider relevant/irrelevant to the query.

As explained earlier, those two pieces of information, along with the query itself, represented the common metadata used for test topics in evaluation campaigns like TREC and CLEF.

An important matter to highlight here is why the experiment required that the users submit task solutions (the summaries). This was done for two reasons:

1. **To cause a realistic search session:** if users engage with the search task knowing that there is some form of assessment at the end, then this is expected to instigate an information need that is genuinely pursued throughout the search session (thus leading to the submission of meaningful queries and to the clicking of relevant results).
2. **To ensure reliability of the test dataset:** part of the cleaning/screening operations that were carried out on the dataset involved manually viewing the task solutions of each user, before transferring them to the next phase, in order to ensure that meaningful solutions were entered (thus reflecting that the users engaged in a meaningful session).

After the user completed the two tasks (as per the process explained above), s/he was asked to fill in a questionnaire about their experience with using the system. The details of this questionnaire and its associated qualitative evaluation are discussed later in Section 5.5.

Finally, the user was presented with a thank-you screen and was notified that s/he will be contacted shortly (within a duration of one to two days) about participating in the remaining phase of the experiment. Some users who started the experiment did not complete their participations till the end. Details about this are given later in subsection 5.4.2.4.

5.4.2.2 Phase 2: Creating a Pool of Results using Various Algorithms

This phase of the experiment was an intermediary phase that did not involve user participation. The last query submitted by the user in each task in the previous phase was reserved for testing the system in this phase. The remaining queries, along with their associated clicked results, were used to construct the three types of user models discussed earlier.

Each test query was automatically submitted to the search system multiple times using various combinations of adaptation algorithms that operate in conjunction with the different user models (as per explained in the discussion of the Evaluation Component in Section 4.1.9). A result list was generated for each algorithm as follows:

1. **Baseline list (i.e. non-personalised) (B):** the source query is submitted to the system. Three lists of results are retrieved: English, French, and German. The result lists are merged using the round robin scheme. No result re-ranking is applied. *This is the same way the search results were generated in Phase 1*.
2. **Result Adaptation based on Fragmented User Model with Score-based Merging (RA-FragUM-SCmerge):** the source query is submitted to the system and then each result in the three result lists is scored based on the corresponding language-fragment of the user model. The lists are then merged together and re-ranked based on the assigned scores. *This is the result adaptation algorithm proposed in Section 4.4.1 combined with the first merging approach discussed in the same section (item#1 on page 127).*
3. **Result Adaptation based on Fragmented User Model with Round Robin Merging (RA-FragUM-RRmerge):** the source query is submitted to the system and then each of the three result lists is scored and re-ranked on its own based on the corresponding language-fragment of the user model. The result lists are then merged using the round robin scheme. *This is the adaptation algorithm proposed in Section 4.4.1 combined with the second merging approach discussed in the same section (item#2 on page 127)*.

4. **Result Adaptation based on Early-Combined User Model[1] (RA-EcombUM):** the source query is submitted to the system and then the three result lists are merged and re-ranked based on the Early-Combined User Model. Since the user model is maintained in the preferred language, the results (i.e. the snippets) are translated to the preferred language where necessary. *This is the result adaptation algorithm proposed in Section 4.4.2.*
5. **Result Adaptation based on Late-Combined User Model (RA-LcombUM):** this is similar to the previous algorithm but applied on the Late-Combined User Model.
6. **Query Adaptation based on Fragmented User Model (QA-FragUM):** the source query is expanded by two terms[2] from the most relevant vector in the corresponding language-fragment of the user model (i.e. the vector that receives the highest total similarity score: *SimT*). The expanded query is submitted to the system. The three result lists are merged using the round robin scheme. No result re-ranking is applied. *This is the query adaptation algorithm proposed in Section 4.3.1.*
7. **Query Adaptation based on Early-Combined User Model (QA-EcombUM):** the source query is expanded by two terms from the most relevant vector in the user model. Since the user model is maintained in the preferred language, all the vectors are translated to the query's language *a priori* if the query's language does not match the preferred language. The expanded query is submitted to the system. The three result lists are merged using the round robin scheme. No result re-ranking is applied. *This is the query adaptation algorithm proposed in Section 4.3.3.*
8. **Selective Query Adaptation based on Fragmented User Model (SeQA-FragUM):** this is similar to QA-FragUM; the only difference is that the source query is only expanded if the vector's total similarity score (*SimT*) is greater than or equal to a threshold of 0.2 (out of 1.0). *This is the selective query adaptation algorithm proposed in Section 4.3.2.*
9. **Selective Query Adaptation based on Early-Combined User Model (SeQA-EcombUM):** this is similar to QA-EcombUM; the only difference is that the source query is only expanded if *SimT* >= 0.2. *This is the query adaptation algorithm proposed in Section 4.3.3.*
10. **QA-FragUM & RA-FragUM-SCmerge:** a combination of query adaptation and result adaptation as per algorithms 6 and 2 respectively.

[1] This is essentially score-based merging because all the results are scored against the same user model.

[2] The rationale behind the chosen values for the thresholds and the variables (e.g. why 2 expansion terms were used) will be discussed after listing all the algorithms.

11. **QA-EcombUM & RA-EcombUM:** a combination of query adaptation and result adaptation as per algorithms 7 and 4 respectively.

All the results generated for a test query were pooled together in preparation for the relevance judgments to take place in the next phase of the experiment.

The algorithms entailed a set of variables and thresholds (configuration parameters), the values of which were arbitrarily set before executing the algorithms. The chosen values are listed below and then a discussion follows about the rationale for choosing them.

The values related to the retrieval of results from the search engine API were set as follows:

- Operating languages: English, French, and German.
- Number of results to retrieve per language: 8. This yielded a final merged list that contained 24 results.

The values related to constructing the user models (Section 4.2) were set as follows:

- The number of interest vectors maintained per language-fragment: $M = 3$ (see page 115). Therefore:
    - each fragment in the Fragmented User Model contained three vectors.
    - the Combined User Model (either of the two subtypes) contained three vectors as it is made up of a single fragment.
- The number of interest terms maintained per vector: $N = 20$.

The values related to the query expansion algorithms (Section 4.3) were set as follows:

- The number of terms to use to expand the query: $n = 2$.
- The SimQ-to-SimD factor: $\alpha = 0.5$. This is the weight value that specifies the influence of query similarity over document similarity when determining the total normalised score (SimT) of a user model vector with respect to the source query (see page 123). The value of 0.5 means that both similarities are equally represented (50%) in the final score.
- The minimum similarity (SimT) threshold for expanding the query in the selective algorithm: $t = 0.2$ (see page 124). A value of 0.2 means that the user model vector has to be at least 20% similar to the topic of the query in order to attempt query expansion.

Regarding the choice of the values for the variables: the main objective of this experiment was to evaluate the three personalisation approaches, thus, the varying factor under investigation

was the type of underlying user model. Therefore, it made sense to fix all the other factors in order to be able to conduct a fair comparison between the three approaches.

Furthermore, the fact that the relevance judgments in this experiment were carried out by the users themselves set a constraint on the experiment: if too many results are placed in the pool then the users would be reluctant to complete the judgment task (which is a lengthy and tedious task). Therefore, this constraint limited the number of alternative values that could be tested for the variables (and combinations thereof). For example, if a range of alternative numbers of query expansion terms had been experimented with[1], then this would have created several queries, which would have yielded several different sets of results, causing the judgment pool to grow unreasonably large. The same effect would have also been caused if a range of values for user model construction was used (different user model vectors means different term distributions and weights, which eventually leads to different terms being used in query expansion). This is also why the result list retrieved in each language was limited to the size of 8 results.

That constraint was also the reason why the algorithm combinations stated above did not include "*Query Adaptation based on Late-Combined User Model*" (it was only tried with Result Adaptation); based on a similar experimental trial conducted earlier (but on a smaller scale in terms of the number of users and number of algorithms), the experimental results of the Late-Combined model were observed to not be promising. In addition to this observation, that early trial served to provide some lessons and to give some preliminary directions for this experiment, such as:

1. Noticing which query expansion algorithms (along with the underlying user model representations) were more promising than others. This helped in minimising the number of results that the users had to judge for relevance in this experiment.
2. Trying a range of values for some of the variables, which helped in deciding upon the values to use for this experiment (e.g. it was found that the selective query expansion algorithm performed relatively better with threshold values between 0.1 and 0.3).

[1] In a number of query expansion experiments in the literature the number of expansion terms ranged between 1 and 20 (and sometimes 50) (Yin et al., 2009). Those experiments operated on their own retrieval component (e.g. Lucene on a closed document corpus) and thus were free to perform any number of retrievals. As for the PMIR experiment discussed above, in addition to the limitation mentioned about users and judgments, the search engine API itself stood out as another limitation in terms of the number of times it can be called for retrieving results.

3. Realising the fact that many users would abandon the experiment when asked to judge a large number of results (for each task in that trial the user was asked to judge a pool, the size of which ranged between 100 and 250 results)[1].

5.4.2.3 Phase 3: Relevance Judgments

In this phase, the users who participated in the first phase of the experiment were asked to judge the relevance of the results in the pools associated with their queries. Each user was shown the last query s/he submitted in each search task along with the metadata s/he entered for the query (query description and narrative). The judgments were performed according to a 4-point scale (*not relevant*, *somewhat relevant*, *relevant*, or *very relevant*). The results were shown in a randomised order so as to avoid bias (i.e. the users were not aware of which result was generated by which algorithm). The users were allowed to carry out the judgments on multiple sessions (i.e. they had the opportunity to sign out of the system at anytime and then sign in again at another time to continue the judgments).

A viable alternative in this relevance judgment phase would have been to ask some other users (e.g. experts) to judge the relevance of the results on behalf of the participants (i.e. instead of the users who participated in Phase 1); an approach that is common in IR studies. However, in order to ensure that the judgments truly reflected the **opinion of the participants** in this **personalisation experiment**, it was more sensible to have the participants themselves describe the test queries and personally carry out the relevance judgments.

The MAP metric (at cut-off values) was used to evaluate the effectiveness of each algorithm (including the baseline) as it rewards relevant search results appearing at higher list positions. Moreover, MAP factors recall as well as precision into its computation; this is particularly useful when comparing query expansion algorithms as different sets of results are retrieved for every modified query.

Since MAP operates on binary relevance judgments, the 4-point-scale judgments were converted to 2-point-scale by taking the higher two judgments as *Relevant* and the lower two judgments as *Irrelevant*.

[1] The results of that early experimental trial are not reported in this thesis because only a few users completed their participation in it (rendering the experimental results not to be very conclusive) and because some problems were discovered afterwards concerning the experimental setup and the data. Therefore, the results of that trial mostly served as "hints" for certain matters and directions but not as conclusive evidence.

5.4.2.4 Additional Details about the Experimental Setup and Data Pre-processing

As can be deduced from the description of the experiment's phases, the experiment required a significant amount of user involvement (in terms of the time they had to spend on phase 1 and phase 3). A difficult challenge encountered in the execution of this study was to encourage users to complete their participation in the experiment (especially phase 3, which they were allowed to fulfil on multiple sessions). This required continuous follow up with them by emails in order to remind them to finish the experiment. However, about 41% of the users dropped out of the experiment at some stage or another (details given below).

Another challenge that faced the experiment was ensuring that the users engaged in meaningful search sessions, thus ensuring the reliability of the final dataset (the search logs). This entailed performing some "sanity checks" and cleaning operations between experimental phases to detect and discard participations that showed obvious signs of random or abnormal user behaviour (e.g. submitting empty task solutions). More details about this follow.

To demonstrate the abovementioned challenges, the following is a report of the number of users who started the experiment and down to the number of users who finished it:

- A total of 128 users started phase 1 of the experiment.
- The number of users who made it to phase 3 was only 94 (i.e. ~27% less). This is because: (a) many users did not finish phase 1; and (b) pre-processing operations involved cleaning out users who: submitted abnormal queries, submitted a query without clicking on any results at all, or submitted empty or insufficient task solutions.
- 15 users did not complete phase 3 (the relevance judgements). Therefore, the number of users who finished phase 3 was 79.
- When carrying out the final stage of evaluation (computing metrics of retrieval effectiveness) errors were found in the data associated with three users (malformed database entries of the URLs and the judgments).

As a result of all this, the final number of users in the dataset was: 76.

Pre-processing operations also involved dealing with task sessions that only involved a single search. This is explained as follows: as discussed earlier, the last query submitted by the user in each task was used as a test query; all remaining queries for the **two tasks**, along with their associated clicked results, were put together and used to train (i.e. construct) the user model. However, in some cases, the user only submitted a single query in one of the tasks and clicked on some of the results and then found that the results s/he viewed provided sufficient

information to solve the task so s/he went directly to the task solution screen. Since that single query cannot be used for both training and testing, such cases were dealt with as follows: **the query of this task was not used as a test query in phase 2**, but was nevertheless used in constructing the user model (this contributed towards a rather realistic setting for the other test query of the user). As a result of this pre-processing operation, not all users had two test queries associated with them in phase 2 onwards.

Table 15 reports a breakdown of the number of users and queries in the final dataset (the languages of the users refer to their preferred languages):

Table 15: final dataset description

Item	Number
Total Users	76
English	56
French	10
German	10
Total Test Queries	98
English	75
French	12
German	11
Total results judged	6,775

5.4.3 Evaluation of Result Adaptation

Table 16 and Figure 23 show percentages of MAP improvement/dis-improvement[1] over the baseline for the result adaptation algorithms:

- Result Adaptation based on Early-Combined User Model (RA-EcombUM).
- Result Adaptation based on Late-Combined User Model (RA-LcombUM).
- Result Adaptation based on Fragmented User Model with Score-based Merging (RA-FragUM-SCmerge).
- Result Adaptation based on Fragmented User Model with Round Robin Merging (RA-FragUM-RRmerge).

[1] In the tables, the asterisk symbol * denotes improvements/dis-improvements that are statistically significant as per 2-tailed T-test, with p=0.05

Table 16: Result Adaptation: MAP percentages (over the baseline)

List Position	RA-EcombUM	RA-LcombUM	RA-FragUM-SCmerge	RA-FragUM-RRmerge
MAP@5	7.01%	-24.98%*	**12.84%**	7.42%
MAP@10	2.91%	-22.18%*	**9.93%**	2.82%
MAP@15	5.73%	-8.81%*	**11.04%***	2.25%
MAP@20	2.82%	-6.70%*	**7.92%***	2.60%

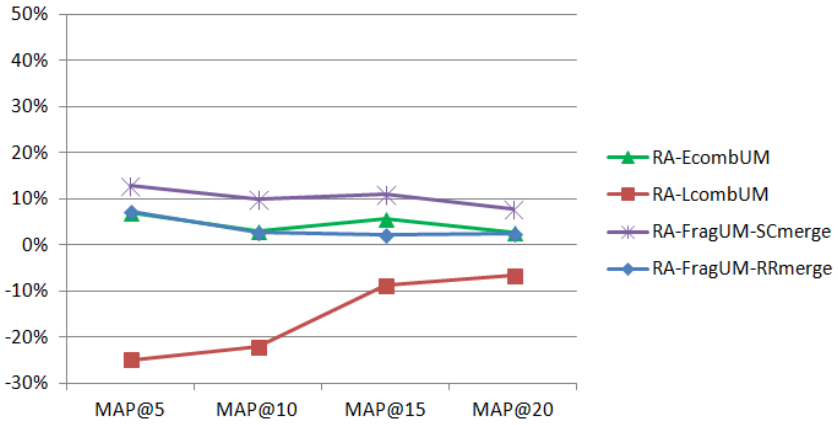

Figure 23: Result Adaptation

The evaluation of the different approaches to result adaptation shows that the most successful algorithm is the one where each search result is scored against the fragment that corresponds to its language in the Fragmented User Model and then the results are all merged and re-ranked based on their scores (RA-FragUM-SCmerge). Moreover, the evaluation shows that the improvements achieved by this algorithm are nearly double the improvements of RA-FragUM-RRmerge and RA-EcombUM. On the other hand, RA-LcombUM was shown to be the worst performing algorithm. This suggests that the approach of creating a Combined User Model by translating the Fragmented User Model (i.e. the Late-Combined approach) is not a successful approach. A possible reason for this is that the process of translating individual terms is subject to more translation inaccuracies because there is no context surrounding each term. This differs from the Early-Combined User Model where the interest terms are harvested from the translated versions of the results that the user clicked on; when translating portions of text, the MT system can make informed decisions regarding candidate translations for a term based on the context of the sentence or paragraph.

The RA-FragUM-RRmerge algorithm and the RA-EcombUM algorithm performed more or less equally although they are based on different user model representations (the Fragmented User Model vs. the Early-Combined User Model). Therefore, it is not possible at this stage in the evaluation to ascertain which user modelling approach leads to better personalisation of

search results. However, what can be deduced so far is that the problem with the Late-Combined User Model is not the fact that it is **combined** but rather its problem lies in the combination approach (i.e. the way it was combined).

In order to gain more insight into the experimental results, the users (i.e. the experimental results corresponding to them) were divided into the following two subsets:

1. English users: these are the users who selected the English language as their preferred language[1] when they signed-up with the system.
2. Non-English users: these are the users who selected French or German as their preferred language.

The objective of this separation was to investigate how the personalisation algorithms performed with respect to the different languages used in search. This investigation approach was based on lessons learnt from the experiments reported in the preceding sections of this chapter.

Table 17 and Table 18 report the MAP improvement/dis-improvement percentages for English users and Non-English users respectively. Figure 24 shows a side-by-side comparison of the two subsets.

Table 17: Result Adaptation: MAP percentages for English users only

List Position	RA-EcombUM	RA-LcombUM	RA-FragUM-SCmerge	RA-FragUM-RRmerge
MAP@5	4.06%	-29.75%*	**6.37%**	-4.60%
MAP@10	0.59%	-24.93%*	**7.77%**	-2.06%
MAP@15	4.11%	-12.10%*	**7.31%**	-1.32%
MAP@20	2.69%	-7.65%*	**6.87%***	0.18%

Table 18: Result Adaptation: MAP percentages for Non-English users only

List Position	RA-EcombUM	RA-LcombUM	RA-FragUM-SCmerge	RA-FragUM-RRmerge
MAP@5	19.61%	-4.63%	40.44%	**58.66%***
MAP@10	11.66%	-11.80%	18.08%	**21.26%***
MAP@15	11.96%	3.89%	**25.43%***	15.98%*
MAP@20	3.29%	-3.42%	**11.59%**	**11.03%***

[1] *Caveat:* the selection of a preferred language does not necessarily imply the native language (or the linguistic background) of the user. It is also worth reminding the reader here that the system only allowed users to choose a preferred language from: English, French, and German.

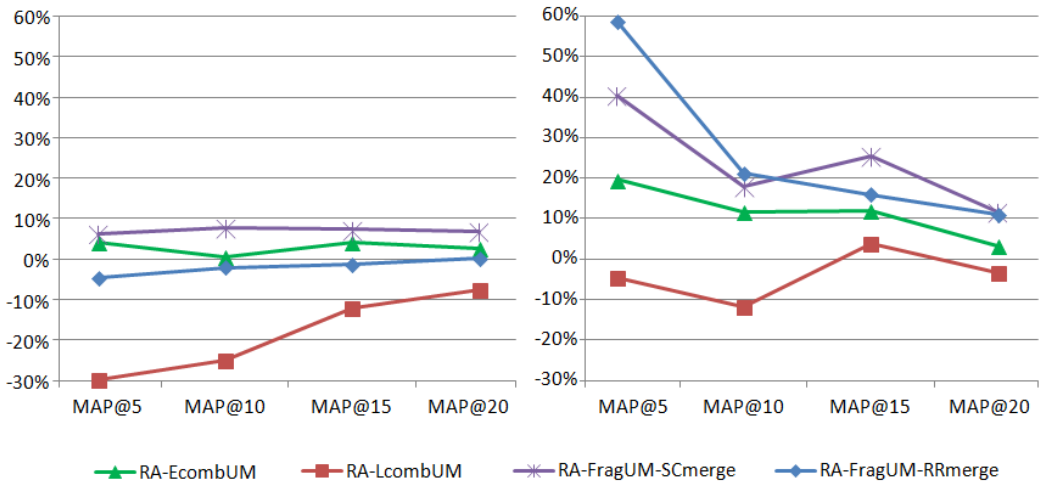

Figure 24: Result Adaptation: English (left) vs. Non-English (right)

The evaluation generally shows higher improvements for the result adaptation algorithms with non-English users. This indicates that the personalisation of search results benefits non-English users much more than it benefits English users. Furthermore, the evaluation shows that the algorithms did not outperform each other in a consistent manner with respect to English vs. Non-English users (e.g. for English users RA-EcombUM outperformed RA-FragUM-RRmerge, but *vice versa* for non-English users). These observations are in agreement with the notion (discussed earlier) that a personalised search system should adopt different personalisation strategies for certain languages or groups of languages. Further analysis of the performance of the personalisation algorithms with respect to English vs. non-English users will be given in Section 5.4.6.

It can be clearly seen that adapting search results based on the Late-Combined User Model is unsuccessful for both English and non-English users. The evaluation reported in the following subsections will only focus on the Early-Combined User Model and the Fragmented User Model.

The evaluation for non-English users shows that the algorithms based on the Fragmented approach to user modelling outperform the algorithms based on the Combined approach to user modelling. In Web search, the results that are displayed to the user in the top 5 positions in the result list are of significant importance when determining the success of a search engine. These results make up the upper half of the list and are usually viewable to the user without the need to scroll down. To that matter, the evaluation shows that the algorithms that are based on the Fragmented User Model were able to achieve 40% to 58% improvement in MAP@5.

When comparing between RA-FragUM-SCmerge and RA-FragUM-RRmerge, the evaluation shows that RA-FragUM-RRmerge algorithm performed better for non-English users in MAP@5 and MAP@10. A closer look on the mode of operation of the two algorithms helps in explaining the implication of this observation: both algorithms score the results in the same way based on the corresponding language-fragment in the user model; however, the merging operation that produces the final ranked list is different. In RA-FragUM-SCmerge, the rank of each result in the final list is solely determined by the score it receives when it is compared to the corresponding language-fragment in the user model. If a fragment contains less information than the other fragments (e.g. in the case of a user model that is still not mature enough) then all the results that are scored against this fragment will eventually be assigned lower scores than the other results. This means that the whole result list of the corresponding language will be weakened and will not be properly represented in the final list although it may have contained results that are relevant to the query. On the other hand, what the RA-FragUM-RRmerge does is that it re-ranks each result list separately based on the scores but then uses the round robin scheme when merging the result lists into the final list. This means that the three result lists have balanced representations in the final list. As the scale of this experiment is relatively small (in terms of the number of users and the number of user interactions in the search logs), it is possible that the user models were not all mature enough; this would be in favour of the RA-FragUM-RRmerge algorithm.

5.4.4 Evaluation of Query Adaptation

Table 19 (all users), Table 20 (English users), and Table 21 (non-English users) show percentages of MAP improvement/dis-improvement[1] over the baseline for the query adaptation algorithms:

- Query Adaptation based on Early-Combined User Model (QA-EcombUM).
- Selective Query Adaptation based on Early-Combined User Model (SeQA-EcombUM).
- Query Adaptation based on Fragmented User Model (QA-FragUM).
- Selective Query Adaptation based on Fragmented User Model (SeQA-FragUM).

Figure 25 shows a side-by-side comparison of English users vs. non-English users.

[1] The asterisk symbol * denotes statistical significance with p=0.05, and the diamond symbol ◇ denotes weak statistical significance with p=0.1

Table 19: Query Adaptation and Selective Query Adaptation: MAP percentages for all users

List Position	QA-EcombUM	SeQA-EcombUM	QA-FragUM	SeQA-FragUM
MAP@5	-2.40%	3.68%	2.30%	**8.32%**
MAP@10	-13.59%$^{\diamond}$	-5.78%	-5.71%	**0.25%**
MAP@15	-14.56%*	-5.95%	-5.88%	-0.13%
MAP@20	-17.01%*	-8.13%	-6.71%	-0.21%

Table 20: Query Adaptation and Selective Query Adaptation: MAP percentages for English users only

List Position	QA-EcombUM	SeQA-EcombUM	QA-FragUM	SeQA-FragUM
MAP@5	-8.89%	-4.57%	-5.39%	-0.04%
MAP@10	-18.94%*	-11.80%	-8.97%	-3.94%
MAP@15	-20.59%*	-12.41%$^{\diamond}$	-9.11%	-4.06%
MAP@20	-20.63%*	-12.14%$^{\diamond}$	-8.26%	-2.85%

Table 21: Query Adaptation and Selective Query Adaptation: MAP percentages for Non-English users only

List Position	QA-EcombUM	SeQA-EcombUM	QA-FragUM	SeQA-FragUM
MAP@5	25.31%	38.86%*	35.12%$^{\diamond}$	**43.96%***
MAP@10	6.59%	**16.95%**	6.56%	**16.05%**
MAP@15	8.68%	**18.94%**	6.57%	15.02%
MAP@20	-4.37%	5.82%	-1.31%	**8.99%**

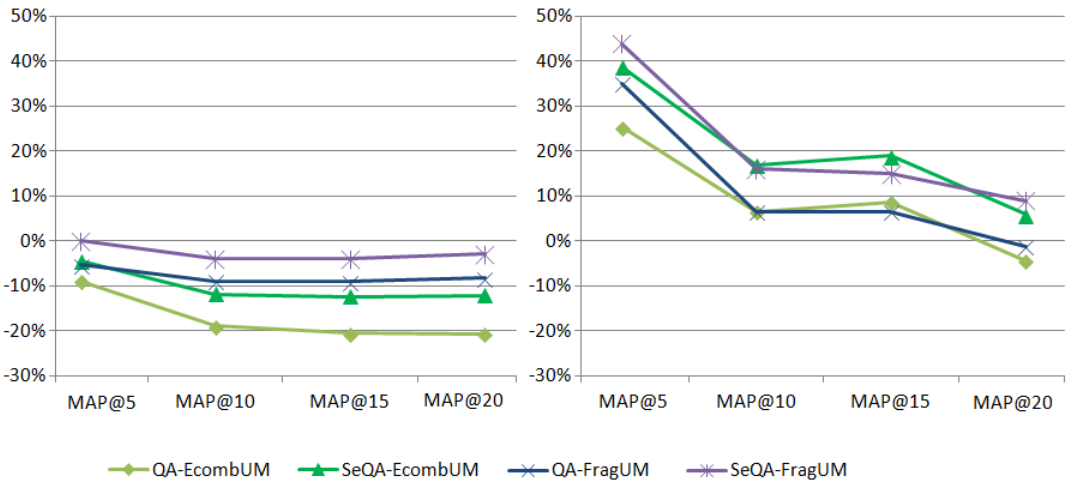

Figure 25: Query Adaptation and Selective Query Adaptation: English (left) vs. Non-English (right)

The evaluation shows that query adaptation in general seems to be more useful to non-English users. This observation is similar to what was found for result adaptation. This indicates that the way non-English users engaged with the search system and formulated their queries is different

from English users. A successful query is one that not only adequately describes the user's intent but also suits the document space representation (i.e. one that contains forms of terms, as in the correct synonyms, that match the content of the documents in the corpus). The observed difference between English and non-English users suggest that the non-English queries may have been weaker on any of the two characteristics, especially the latter, and therefore they had more room for improvement. To this effect, the interest terms in the user model have the advantage of encompassing both characteristics: they reflect what the user is interested in (i.e. their underlying intent) and the majority of the terms are picked from the document space itself (from clicked results).

The evaluation also shows that the algorithms based on the Early-Combined User Model mostly perform worse than their counterparts that are based on the Fragmented User Model (with the exception of MAP@15 for non-English users). This suggests that better expansion terms were obtained from the Fragmented User Model. This can be attributed to two potential factors: (1) the separation of the interest vectors by language stands out as a higher-level of clustering the interest terms and therefore acts as a filter that allows the query adaptation process to focus on the subset of interest terms that are of more relevance to the query; and (2) a subset of the interest terms in the Early-Combined User Model are obtained from the translated versions of the results, not the original, and therefore may be subject to translation inaccuracies. Further analysis (reported at the end of this subsection) suggests that the first factor (i.e. the way the user model is represented) is the reason behind the difference in performance.

When examining the process of selective query adaptation, the evaluation shows that it performs better than non-selective query adaptation. The SeQA-EcombUM algorithm was approximately 7% higher than QA-EcombUM on average for English queries and 11% higher for non-English queries. As for the SeQA-FragUM algorithm, it was 5% higher than QA-FragUM for English queries and 9% higher for non-English queries. These experimental results demonstrate the ability of the selective process to reduce the harm that query adaptation sometimes causes to retrieval effectiveness.

On the matter of benefitting/harming query performance when applying query adaptation, a deeper analysis is required in order to understand the implication of the experimental results. The following questions demonstrate the need for this analysis: What does a 0% MAP improvement mean? Does it mean that the query adaptation process has no effect on retrieval effectiveness? Or does it mean that the process benefits some queries as much as it harms other queries, yielding a net effect of 0% on retrieval effectiveness? Furthermore, how far does the

selective adaptation process increase the success rate when compared to the non-selective process?

In order to answer these questions, two steps of analysis were carried out:

1. Examining how many queries were improved and how many queries were harmed by the query adaptation process (QA-FragUM and QA-EcombUM).
2. Conducting a True/False Positive/Negative analysis concerning the selective query adaptation process (SeQA-FragUM and SeQA-EcombUM). Figure 26 explains the four possible cases for each query. The positive/negative dimension (the x-axis) represents the decision of whether to expand the query or not and the true/false dimension (the y-axis) represents whether the decision was correct or not.

The analysis concerning the Fragmented User Model will be presented first and then followed by the analysis of the Early-Combined User Model.

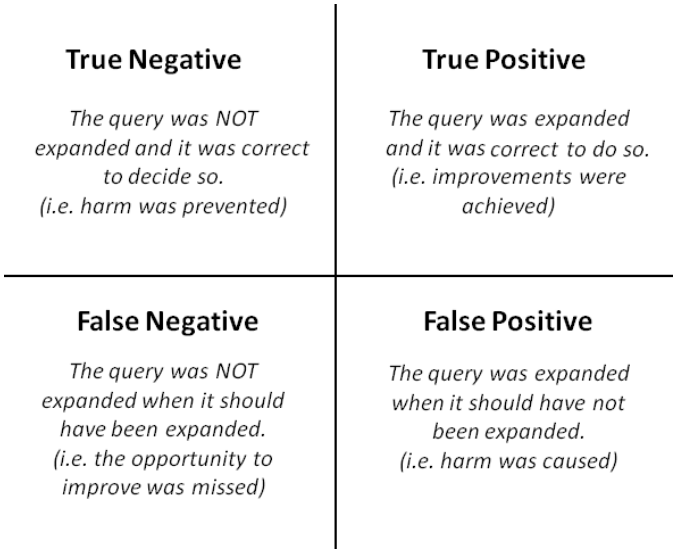

Figure 26: layout of T/F +/- analysis

Regarding QA-FragUM (the algorithm that attempts to expand all queries), the total number of English queries was 75, of which 36 were improved and 39 were harmed. As for the non-English queries, the total number was 23, of which 11 were improved and 12 were harmed. This means that QA-FragUM exhibited a ~48% chance of benefitting the queries (for both English and non-English queries).

Table 22 reports the True/False Positive/Negative (hereafter: TFPN) analysis of SeQA-FragUM (the algorithm that applies the selective process) for English queries and Non-English queries.

Table 22: TFPN of Selective Query Adaptation based on Fragmented UM

English		Non-English	
TN: 8	TP: 35	TN: 4	TP: 11
FN: 1	FP: 31	FN: 0	FP: 8

The TFPN analysis of the SeQA-FragUM algorithm reveals the following:

- It was able to reduce the number of harmed English queries from 39 to 31 (i.e. it was able to prevent the harm ~21% of the time).
- It was able to reduce the number of harmed non-English queries from 12 to 8 (i.e. it was able to prevent the harm ~33% of the time).
- The number of improved English queries was only reduced from 36 to 35 (i.e. by making a wrong decision, the opportunity to improve was missed ~3% of the time).
- The number of improved non-English queries remained the same (i.e. it did not miss any opportunities to improve).

The analysis shows that the selective process, applied in conjunction with the Fragmented User Model, raised the success rate from ~48% to ~53% for English queries and from ~48% to ~58% for non-English queries. Moreover it shows that the selective process prevents harm more than it prevents improvements by mistake (i.e. the True Negative rate is greater than the False Negative rate).

Regarding QA-EcombUM, 31 out of the 75 English queries were improved and 44 were harmed. As for the non-English queries, 11 out of 23 were improved and 12 were harmed. This means that QA-EcombUM exhibited a success rate of ~41% with English queries and ~48% with non-English queries. Table 23 reports the TFPN analysis of SeQA-EcombUM.

Table 23: TFPN of Selective Query Adaptation based on Early-Combined UM

English		Non-English	
TN: 8	TP: 29	TN: 2	TP: 11
FN: 2	FP: 36	FN: 0	FP: 10

The analysis reveals the following about SeQA-EcombUM:

- It reduced the number of harmed English queries from 44 to 36 (i.e. it prevented harm ~18% of the time).
- It reduced the number of harmed non-English queries from 12 to 10 (i.e. it prevented harm ~17% of the time).
- The number of improved English queries was reduced from 31 to 29 (i.e. the opportunity to improve was missed ~7% of the time).
- The number of improved non-English queries remained the same (i.e. it did not miss any opportunities to improve).

The analysis shows that the selective process, applied in conjunction with the Early-Combined User Model, raised the success rate from ~41% to ~45% for English queries and from ~48% to ~52% for non-English queries.

When comparing SeQA-FragUM to SeQA-EcombUM, the analysis shows that the former outperforms the latter in terms of increasing the success rate (i.e. pre-detecting harm and avoiding it) and also in terms of the True Negative rate vs. the False Negative rate (i.e. the rate of correct vs. incorrect decisions when deciding that a query should not be adapted). This suggests that the Fragmented User Model is more informative to the process than the Early-Combined User Model.

Samples of successful and unsuccessful query adaptations from the experimental dataset are given in Table 24. The table is organised in the following manner:

- Source Query: the input query –before adaptation.
- Lang: language of the source query which is either English (en), French (fr), or German (de).
- Adapted Query: what the query **was** adapted to (or **would have been** adapted to if the selective decision was to allow adaptation). This is split into two cells: the first cell shows the adapted query based on the Fragmented User Model and the second cell shows the adapted query based on the Early-Combined User Model.
- Improved: whether the query **is** (or **would have been**) improved by the adaptation process or not.
- TFPN: this indicates the decision of the selective process (P: carry out the adaptation, N: do not adapt) and whether the decision was successful or not (T: correct decision, F: wrong decision).

Table 24: samples of query adaptations

#	Source Query	Lang	Adapted Query: 1. based on FragUM 2. based on EcombUM	Improved?	TFPN
1	arizona wildlife	en	arizona wildlife **diversity nature**	Yes	TP
			arizona wildlife **diversity nature**	Yes	TP
2	syria news	en	syria news **breaking latest**	Yes	TP
			syria news **sport squash**	No	FP
3	apple iphone 6 US	en	apple iphone 6 US **mobile advances**	No	TN
			apple iphone 6 US **advances korea**	No	TN
4	India's culture and eating habits	en	India's culture and eating habits **indian healthy**	No	FP
			India's culture and eating habits **indian food**	No	FP
5	the relegious believes in Taiwan, ROC *[sic]*	en	the relegious believes in Taiwan, ROC **celebaration traditions**	No	TN
			the relegious believes in Taiwan, ROC **traditions marriage**	Yes	FN
6	les habitudes alimentaires en Afrique du Sud	fr	les habitudes alimentaires en Afrique du Sud **resto même**	Yes	TP
			les habitudes alimentaires en Afrique du Sud **régime mangez**	Yes	TP
7	Venezuela Natur	de	Venezuela Natur **wunder weltwunder**	Yes	TP
			Venezuela Natur **wunder everest**	No	FP
8	entdeckung von erdgas 2013 Ägypten	de	entdeckung von erdgas 2013 Ägypten **länder grafiken**	No	FP
			entdeckung von erdgas 2013 Ägypten **verbrauch exporte**	No	FP

The query samples, and the terms used to expand them, give additional insight into the query adaptation and user modelling processes. The following paragraphs highlight these insights.

In some cases, although the expansion terms are textually or semantically similar to the terms of the query (e.g. query#4: "India"-"indian" and "eating"-"food"), they might harm retrieval effectiveness because they do not reflect the specific intent of the user (e.g. seeking habits of eating and not food recipes or dishes).

The query adaptation algorithms are programmed to avoid duplicates when expanding the query (i.e. a query should be expanded with terms that are unique and that do not already appear in the query). However, in some cases a lexical or semantical variation of the term is used (e.g. query#7: "wunder" - "weltwunder"). This causes a sense of redundancy whereby a term does not add any information value to the query.

In the experiment, no spelling-correction feature was used. Thus, any mis-spelt words input by the user were propagated without change to the search logs (and also to the underlying search engine API). This caused some malformed terms to exist in the user model and consequently to be used in query adaptation (e.g. query#5).

Finally, an expected but noteworthy phenomenon was that the expansion terms obtained from the Fragmented user and the Early-Combined User Model were sometimes different from each other and sometimes the same. The existence of same terms can be attributed to the fact that the terms in each model are extracted from the same resources (queries and snippets of clicked results –whether the original or translated snippets). This may indicate that translation of snippets was generally of consistent quality. The existence of different terms can be attributed to the different representation within each model. Therefore, it may be deduced from the evaluation that the structure of the user models is the reason why the query adaptation algorithms that are based on the Fragmented User Model outperformed their counterparts that are based on the Early-Combined User Model.

These findings may provide specific guidance for future research in this area.

5.4.5 Evaluation of Combining Query Adaptation and Result Adaptation

One of the novel contributions of this thesis is that, in addition to providing an evaluation of the individual adaptation components, it also provides an evaluation of the combined outcome of these components. This offers answers to the following questions: which adaptation approach is more effective for PMIR: Query Adaptation or Result Adaptation? what is the effect of applying a combination of both adaptation approaches? does it cause a synergetic effect that leads to higher improvements or does it cause a noisy overlap that degrades effectiveness? which one of the two adaptation approaches has more influence on the combined (mixed)[1] output?

Figure 27 (English users) and Figure 28 (Non-English users) show MAP evaluation for the algorithms that mix both query adaptation and result adaptation (score-based merging) based on the Fragmented User Model and the Early-Combined User Model. In order to relate and compare the mixed algorithms with the individual algorithms, the figures also show the

[1] In order not to confuse the reader between the "combined" approach to adaptation and the "combined" user model, from that point on in this subsection the approach that combines query adaptation with result adaptation will be referred to as: "the mixed approach".

individual evaluation reported earlier for the query adaptation (QA-FragUM and QA-EcombUM) and result adaptation (RA-FragUM-SCmerge and RA-EcombUM).

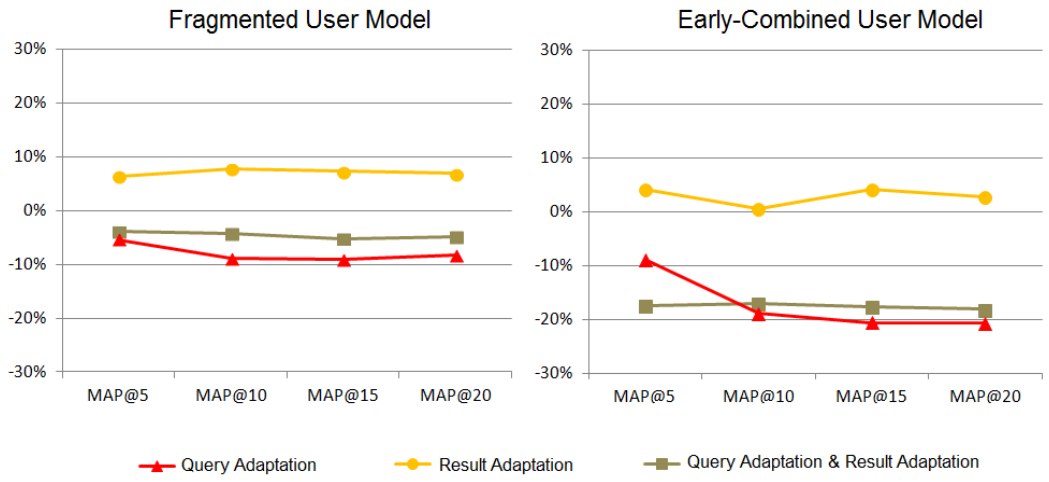

Figure 27: MAP percentages of QA, RA, QA&RA: English users

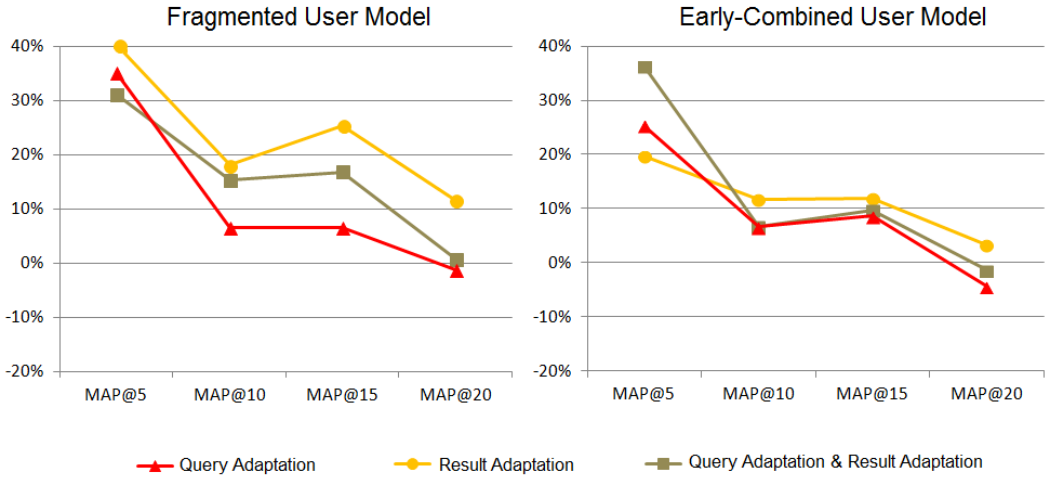

Figure 28: MAP percentages of QA, RA, QA&RA: Non-English users

The evaluation shows that the general trend when mixing the two adaptation approaches is that the desired synergy is usually not achieved. This can be explained in light of the analysis that was carried out in the previous subsection: personalisation in general, whether by adapting queries or results, entails the risk of making wrong assumptions in some cases and thereby harming retrieval effectiveness. What the evaluation indicates is that the performance of the mixed adaptation approach relies on two factors:

1. The **overlap** between the successful attempts and the failed attempts of its two components (query adaptation and result adaptation).
2. The degree of **influence** of each component on the overall output (how far each component contributes to the final outcome).

The **overlap** factor adds another layer of risk to the personalisation process, which is the risk of incorrect overlap. It could be the case that a query is successfully adapted but then the search is harmed by a failed result adaptation attempt or *vice versa*. This increases the chances of user dissatisfaction with the search results.

Regarding the **influence** factor, the evaluation indicates that query adaptation has more influence on the final score than result adaptation. This is an expected phenomenon because if a query was adapted unsuccessfully, this would yield a set of results that are not relevant to the user's information need, in which case the process of adapting the results may be rendered useless because all the results are not relevant in the first place. On the other hand, if a query was adapted successfully, this would yield a set of results that are even more relevant to the user's information need; if the result adaptation process goes wrong, then the harm will be that relevant results will switch places with less relevant results and be pushed further down the list, but can still be located within the search results nevertheless. Therefore, the outcome of the mixed adaptation approach is more affected by the query adaptation component than by the result adaptation component. This is further confirmed by observing the difference in performance of the mixed approach between Figure 27 and Figure 28: it can be seen that the line representing the mixed approach is low when the query adaptation line is low, and the line is higher up when query adaptation goes higher up.

Finally, the evaluation shows that result adaptation is a more successful approach than query adaptation and also than the combination of the two. This indicates that result adaptation is a less risky approach to personalisation in PMIR.

5.4.6 Further Analysis of Findings

One of the main findings of the experiment is that personalisation based on the Fragmented User Model was more effective than personalisation based on the Combined User Model. This finding supports the underlying assumption of the Fragmented User Model, which is that users exhibit different search interests across languages and that the user modelling approach should reflect this.

What makes this finding particularly interesting is that each user in this experiment submitted queries in one language only (the preferred language). This shows that because the content is multilingual, this leads to the existence of user interests in multiple languages, regardless of the language used to search the system. This supports the notion that, in multilingual search, the

users may choose to click on documents originating from certain languages depending on the type of information sought –whether the users themselves are monolingual or multilingual.

Therefore, a lesson to be learned from this finding is that users would welcome results from languages other than the one they used to query the system if those results offer a **better satisfaction** to their information need. In PMIR, this satisfaction mainly depends on three factors:

1. The ability of the system to retrieve results that are relevant to the query from other languages (languages other than that of the query).
2. The ability of the system to retrieve results that are relevant to the user.
3. The ability of the system to make the retrieved information accessible to the user in case they are not familiar with the language in which information is provided.

Another interesting finding is that different experimental results were exhibited for English users vs. non-English users when applying the personalisation algorithms. In order to gain more insight into this phenomenon, the retrieval effectiveness of the baseline algorithm was examined (i.e. the non-personalised algorithm –the first one mentioned in page 158). Table 25 reports the Precision scores of the baseline lists for English and non-English users at various list positions.

Table 25: baseline Precision scores

List Position	English	Non-English	Percentage of English over Non-English
P@5	0.58	0.45	29.15%
P@10	0.55	0.49	11.54%
P@15	0.51	0.45	14.46%
P@20	0.50	0.48	3.71%

The baseline Precision scores show that when non-English users used the system they were getting back results with lower relevance than for English users. This suggests that there was more room for the personalisation algorithms to improve over the baseline for non-English users. In other words, one of the reasons why lower improvement percentages were exhibited for English users when applying the personalisation algorithms is that the effectiveness of the baseline algorithm was relatively higher; this provided less opportunity for the personalisation algorithms to improve over the baseline.

An important matter to highlight here is that the higher Precision scores for English users is not a result of the order in which the multilingual results were presented (i.e. the order of languages

in the final result list which was obtained using round robin –see page 128). This is because the Precision metric does not take the order of results into consideration; it just reports the fraction of relevant results within the given result list. The significance of this is that it puts more emphasis on the notion that the exhibited differences are attributed to the way users query the system and less emphasis on the notion that the differences are attributed to the amount of relevant content available in each language.

An inference that can be made about non-English users, based on the observed lower baseline scores, is that they were relatively less satisfied with the search results they received when they used the baseline system in phase 1 of the experiment. This inference was later confirmed by the questionnaire that the users answered at the end of phase 1 (the outcomes of the post-system-usage questionnaire are fully discussed in Section 5.5).

One of the findings of the analysis of The European Library search logs reported earlier (Section 5.2.4) was that a single personalisation strategy may not fit all users. The findings from this PMIR experiment are in agreement with the earlier finding. This is demonstrated by the following:

- For English users, the approach of adapting results was more successful than the approach of adapting queries. On the other hand, both adaptation approaches were successful with Non-English users, as was the combination of the two approaches.
- On algorithm level, the evaluation of result adaptation showed that some algorithms performed better with English users and other algorithms performed better with non-English users.
- Regarding the TFPN analysis for selective query adaptation, the ratio between True Negatives and False Negatives (i.e. making correct vs. wrong decisions about not adapting a query) differed between English and non-English queries. This suggests that a certain threshold[1] value may be more suitable for English queries while another value may be suitable for non-English queries.

[1] This is the threshold that determines whether a query should be adapted or not based on its similarity (SimT) with the user model.

These findings emphasise that, in order for search systems to offer a better personalised service for users coming from different linguistic and cultural backgrounds, they should employ different personalisation strategies for each language or group of languages. A *personalisation strategy* comprises the following elements:

- The choice of user modelling approach (how the user model is structured).
- The choice of personalisation approach (whether to apply query adaptation, result adaptation, or both).
- The choice of specific algorithms within the selected personalisation approaches.
- The choice of thresholds for assertiveness when executing the personalisation algorithms (which controls whether the strategy is an aggressive one or a conservative one when making assumptions about the user and making personalisation decisions).

Finally, an important point to emphasise here is that this specific set of experiments demonstrated the need for the PMIR framework. The framework provided the necessary means for configuring, executing, and evaluating the elements of a personalisation strategy. It also provided the necessary means for comparing multiple strategies to each other. This validates the notion of a reproducible framework that guides experimentation in the field of PMIR.

5.4.7 Limitations and Caveats

The main limitation in this experiment is the search logs are small in scale (i.e. contains a few interactions for each user). As discussed in Chapter 2, some studies, especially the ones affiliated to major search engine companies, were conducted on large-scale search logs. For some researchers in the IR community, this difference in scale may cause doubts regarding the reliability of the findings. To that end, the role of the findings of this experiment can be regarded as a step towards a better understanding of the implications of multilinguality on search personalisation. It highlights novel research directions with regards to user modelling approaches and personalisation approaches, and shows that there is potential benefit behind exploring these research directions. In addition to this, the benefit of conducting experiments that are based on user trials is that qualitative evaluation can be carried out by administering questionnaires to the users after the trials, which is something that large-scale experiments lack. Furthermore, there are no such multilingual Web-search logs available in the research community to date.

Moreover, specifically regarding the finding that a user's interests are distributed across languages, if the experiment is able to demonstrate this on such a small-scale dataset, then it is expected that with more user activity more diverse interests across languages will be exhibited for the user in the logs. Therefore, in turn, it is expected that the Fragmented User Model will be an even better representation of the user. The key message here being that the more the user interacts with multilingual content, the more distributed his/her interests will be across languages, the more appropriate the Fragmented User Model will be in reflecting that phenomenon.

A viable alternative to the way this experiment was set up was perhaps to ask the users to freely use the system for their own searches over a period of time (e.g. 6 months) instead of assigning immediate search tasks to them. However, this kind of setup was not undertaken because of the following reasons:

1. The PMIR system is not a typical Web search engine, especially with regards to the multilingual search feature. Therefore users may be reluctant to use it on a periodic basis (as opposed to using their favourite search engine).
2. Users typically have fixed workflows and routines in mind and are reticent to changing them.
3. If the users get the feeling that they are forced to use the system, they might not come up with genuine information needs that they need to pursue by interacting with the system (i.e. they might end up submitting random queries and clicking on random results).
4. The current experimental setup allowed for directly administering the post-usage questionnaire while the experience is still fresh in their minds.

5.4.8 Conclusion

This set of experiments compared various personalisation approaches to each other through quantitative evaluation. This included the evaluation of the personalisation components individually and in combination with each other, which revealed how each component contributed to the personalisation process. This kind of comparative evaluation was directly supported by the use of the PMIR framework, which provided a platform for running the user trials and for automating various parts of the evaluation process.

The outcomes of these experiments address the following challenges of the thesis:
Challenge #3: *Can the use of user models that encompass the aspect of multilinguality improve retrieval effectiveness in PMIR?*

Challenge #4: *How should query adaptation and result adaptation algorithms be extended in order to incorporate the aspect of multilinguality?*
The outcomes also contribute to ***Challenge #2:*** *Are there certain behavioural patterns or differences that can be observed for users in multilingual search?*

The experiments addressed these challenges by showing that the effectiveness of multilingual search can be improved by employing personalisation approaches that cater for multilinguality. These approaches comprise user models that represent the user's multilingual search interests and comprise adaptation algorithms that are applied based on these models. The experiments also showed that different personalisation strategies may suit different groups of users, depending on the language attribute.

The following section reports qualitative evaluation for the PMIR experiment, which addresses challenge#5 of this thesis.

5.5 Qualitative Evaluation of System Usability, Multilingual Search Features, and Translation Quality

This section presents the qualitative evaluation of the PMIR system. This involves the online questionnaire that was administered to the users of the PMIR system right after they completed phase 1 of the PMIR experiment (Section 5.4.2.1), in which they used the baseline multilingual Web search system. The questionnaire is given in Appendix-C.

5.5.1 Objectives

The questionnaire served the following objectives:

- To evaluate system usability.
- To evaluate the users' perception of features that are specific to the multilingual search service.
- To evaluate the quality of translation.
- To gain understanding of what the users liked and disliked about the system.

Each objective was addressed by a subset of the questions in the questionnaire. The analysis reported in the following subsections is carried out on the responses of the 76 users altogether. Moreover, some parts of the analysis are reported for English users vs. non-English users when there are noteworthy differences. For consistency with the analysis reported in the previous

section, the breakdown of users by language is based on the language that the users specified as their *preferred language* when they signed up with the system; it does not necessarily reflect their native language.

5.5.2 Analysis: Usability

The usability of the search engine[1] was evaluated using the *System Usability Scale (SUS)* questionnaire (Brooke, 1996). The questionnaire is made up of 10 statements, 5 of which are positive-toned (the odd-numbered statements) and 5 are negative-toned (the even-numbered ones). Each statement (hereafter referred to as *question*) is answered on a 5-point *Likert Scale*: (1) Strongly Disagree, (2) Disagree, (3) Not Sure, (4) Agree, (5) Strongly Agree. The aim of having a mix of positive and negative questions is to minimise the effect of extreme response bias and to allow detecting invalid responses (e.g. a person agreeing to all positive and negative statements –which contradict each other).

Table 26 reports the median and mean of the users' answers to the positive-toned questions. The results indicate that the users were satisfied with the overall usability of the search engine.

Table 26: median and mean of answers to the positive-toned SUS questions

Question	Median (with Likert label)	Mean
I think I would like to use this search engine frequently	4 (Agree)	3.64
I thought the search engine was easy to use	4 (Agree)	4.3
I found the various functions in this search engine were well integrated	4 (Agree)	4
I would imagine that most people would learn to use this search engine very quickly	4 (Agree)	4.32
I felt very confident using the search engine	4 (Agree)	4.21

A noteworthy difference between English and non-English users was observed in the results of the first question in the table: *I think I would like to use this search engine frequently*. The median and mean for English users were 4 and 3.78 respectively while the median and mean

[1] The term "system" was replaced with the term "search engine" in the SUS questionnaire in order to make it clear to the participants that they were being asked about the search engine itself and not other parts of the experimental system that they interacted with (e.g. task selection screen, screen of entering task solutions, etc.).

for non-English users were 3.5[1] and 3.25. The lower results for non-English users, which fall between *Agree* and *Not Sure*, reflect that they were slightly less satisfied with the search engine. This finding is in agreement with the finding discussed in Section 5.4.6.

Table 27 reports the median and mean of the users' answers to the negative-toned questions. The results indicate that the users had no difficulty in using the search engine.

Table 27: median and mean of answers to the negative-toned SUS questions

Question	Median (with Likert label)	Mean
I found the search engine unnecessarily complex	2 (Disagree)	1.79
I think that I would need the support of a technical person to be able to use this search engine	1 (Strongly Disagree)	1.31
I thought there was too much inconsistency in this search engine	2 (Disagree)	1.97
I found the search engine very cumbersome (complicated) to use	1 (Strongly Disagree)	1.53
I needed to learn a lot of things before I could get going with this search engine	1 (Strongly Disagree)	1.39

An overall SUS score is calculated for each user across the 10 questions[2] in the following way:

1. For each positive-toned question, the value of 1 is subtracted from the user's response.
2. For each negative-toned question: the user's response is subtracted from 5.
3. For each user, the new unified response values (which have been converted to a scale from 0 to 4) are summed up and then multiplied by 2.5. This produces a final overall score for the user that is between 0 and 100.

The median and mean scores calculated across all the users were: 82.5 and 81.28 out of 100. This reflects that the system achieved a high usability score.

[1] The median being 3.5 is a result of having an even number of participants where the median fell between 3 and 4 (for the 38th and the 39th entries respectively).

[2] a few empty responses were encountered among the users; those were replaced with the value of 3 in order to be able to calculate the SUS score (i.e. as if the user responded with the neutral answer: "Not Sure").

A study conducted by Jeff Sauro[1] reported an analysis carried out over 500 SUS evaluations in the literature. The findings of the study can be summarised in the following points:

- The average SUS score across the 500 studies was found to be: 68. Therefore, scores above this value are considered as *above average* scores.
- The study devised six percentile ranks based on the analysis conducted on the 500 evaluations. The percentiles go from ***A*** to ***F***, where ***A*** is the highest percentile (scores of 80.3 or higher) and ***F*** is the lowest percentile (scores of 50 or lower).

Thus, according to that study, the SUS score of the PMIR system lies within percentile ***A***, which reflects that the usability of the PMIR system belongs to the top 10% of the 500 studies.

The outcome of the usability evaluation indicates that the users found the search interface easy to use. This suggests that the decision to design the GUI in a way that is as close as possible to major search engines was a correct decision, which was a design consideration discussed in Section 3.6.2.2; it helped users get used to the system quickly and it served to isolate the effect of the quality of HCI design when evaluating the multilingual-search-specific features.

5.5.3 Analysis: Multilingual Search Features

This subsection discusses the set of questions that were concerned with what the users thought of the features of the multilingual search service. As with the SUS questionnaire, the questions were designed to have a mix of positive-toned and negative-toned statements, to which the users indicated their agreement on a 5-point *Likert Scale*.

Table 28 reports the median and mean of the users' answers to the multilingual-search-specific questions (*negative-toned questions are marked in italics*). The table also states the aim of asking each question.

[1] http://www.measuringusability.com/sus.php

Table 28: answers of questions about multilingual search features

#	Question	Underlying Objective	Median (with Label)	Mean
1	I found the search system returned relevant results to my queries	to get a sense of how users generally felt about the relevance of the search results presented to them	4 (Agree)	4.28
2	*Many of the results were irrelevant to my query*	*same objective as Q#1, but asked in a negative-toned manner*	*2 (Disagree)*	*2.34*
3	The presentation of interleaved (mixed) results from different languages was useful	to specifically evaluate the users' perception of the notion of interleaved results, which is an essential element of multilingual search	4 (Agree)	3.78
4	The system returned results that were helpful in solving the search task	this question also asks about the general relevance of search results, but it goes a step further by addressing the matter within the context of the users' attempts to solve the search tasks	4 (Agree)	4.27
5	*I think the mixing of multilingual results was confusing*	*same objective as Q#3, but asked in a negative-toned manner. Additionally, this question aimed to explore whether the presentation of results from multiple languages added an element of confusion to the users.*	*2 (Disagree)*	*2.13*
6	The system encouraged me to explore information coming from languages other than my native/preferred language	a step further from Q#3 and Q#5, this question aimed to assess whether users would be happy to consume information that comes from other languages if a system provides such opportunity	4 (Agree)	4.07
7	*I found that a lot of information was redundant between languages*	*to evaluate whether the retrieval of results from other languages has an added information value to the user.*	*3 (Not Sure)*	*2.95*
8	*I had to search a lot before I was able to find useful content*	*to evaluate the users' satisfaction with the search engine by implicitly inquiring if they had to re-formulate their queries several times*	*2 (Disagree)*	*1.99*
9	I think I did well in solving the tasks	this question helps in detecting if the users' information needs were satisfied	4 (Agree)	3.89

The results of questions #1, #2, #4, and #8 indicate that users were generally satisfied with the search results. A noteworthy difference was exhibited for English vs. non-English users in question#2; the median and mean for English users were 2 and 2.21 respectively, compared to

2.5 and 2.7 for non-English users. This further confirms the assumption that English users were slightly more satisfied with the relevance of the search results.

The results observed for English and non-English users in question#4 are worth highlighting here: the median and mean for English users were 4 and 4.31 respectively, and for non-English users they were 4 and 4.15. The fact that the difference between the two means is rather small, and that the medians were equal, indicates that English and non-English users shared the opinion that the information supplied by the search results was helpful in solving the search tasks. This may seem to contradict with the finding discussed in the preceding paragraph regarding question#2. However, understanding the objectives of each question clarifies the contradiction; the idea that question#4 is tackling is: if the search engine presented fewer relevant results than the user expected, were those results nevertheless sufficient to solve the tasks? The result obtained for non-English users suggests that this was indeed the case. This was further confirmed by the results of question#9 which indicated that both English users and non-English users believed they did well on the search tasks; English results (median and mean) were 4 and 3.93 and non-English were 4 and 3.8. This indicated that their information needs were satisfied, despite some of the results not being very relevant.

The results of questions #3, #5, and #6 indicate that the users accepted the feature of mixed multilingual results and were not confused by the way the results were presented. This is a new feature that users are not used to in typical search engines, and therefore, the outcome of these questions reflects that introducing this feature to search engines would be potentially successful.

Finally, the result of question#7 indicates that there might have been a sense of redundancy between the results retrieved from the three languages in some cases. This was also reflected in the comments provided by some of the participants in the open text questions (discussed later in Section 5.5.5). For example, three users noted the fact that they received *Wikipedia* results in more than one of the three languages (e.g. getting a search result about a topic in English Wikipedia and getting another search result about the same topic but in German Wikipedia); however, an interesting fact was that other users commented that this was a favourable thing to them because it allowed them to view different points of view (from different cultures) about the same topic[1].

[1] Wikipedia articles in different languages, about a topic, are not necessarily translated versions of each other.

The studying of information redundancy, and the implications thereof, is out of the scope of this thesis. Nevertheless, a lesson to be learnt from these observations is that future research in this area should perhaps take the matter of information value into consideration.

5.5.4 Analysis: Translation Quality

This subsection is concerned with translation-related questions. The questions address matters of usefulness and adequacy of translated content, in addition to the actual translation quality; hence the notion of *translation-related* questions.

This part of the evaluation is reported only for 71 out of the 76 users. Five users were omitted because when they signed up with the system they indicated that they were familiar with the three languages (English, French, and German), thus, they were not exposed to any translated content. Table 29 reports the median and mean of the users' answers to the translation-related questions.;

Table 29: answers of translation-related questions

#	Question	Underlying Objective	Median (with Label)	Mean
1	*The translated results were less helpful in solving the tasks than other non-translated results*	*to indirectly evaluate the accuracy of translation by assessing the usefulness of the information provided by the translated results with respect to non-translated results.*	*3 (Not Sure)*	*2.65*
2	The quality of the translation was good	to directly evaluate the user's perception of the quality of translation	4 (Agree)	3.67
3	I was able to understand the information that came from languages other than my native/preferred language	to indirectly assess whether the translation quality was good enough to convey the information in the documents	4 (Agree)	3.99

The results of question#1 show that users neither agreed nor disagreed that the translated results were less helpful than the non-translated results (with a tendency towards disagreeing as suggested by the value of the mean). This indicates that the translated results were as useful to users as the non-translated results.

An important aspect to note about question#1 is that it is two-fold: it implicitly indicates the quality of **query translation** as well as the quality of **document (result) translation**. Regarding the former, if queries are not translated successfully then this would probably yield irrelevant search results which are not at all helpful in satisfying the user's information need – regardless of the quality of document translation. Regarding the latter, helpful documents indicate that the information within them was successfully conveyed to the user.

Another aspect to note about question#1 is that, by asking the user to conduct this relative comparison ("less helpful... **than**"), the question is essentially avoiding any bias that may be caused by the degree of relevance of the original version of the translated results. For example, it may be the case that all results are not of high relevance to the query, including the ones that were not subject to translation; in such case, if the user was just asked: "*The translated results were not helpful in solving the tasks*" then s/he would have agreed to that statement, which may have led to the incorrect assumption that translation was the reason for this, and not the content of the document itself (which was not relevant to the query in the first place). Avoiding this kind of bias means that question#1 isolates the characteristic of result relevance and focuses on the characteristic of translation quality.

A noticeable difference between English and non-English users was observed in question#1. The median and mean for English users were 2 and 2.51 respectively, while for non-English users they were 3 and 3.06. This indicates that English users were more satisfied with the information provided by translated documents than non-English users.

English and non-English users exhibited a similar trend in question#2. The median and mean for English users were 4 and 3.88, while for non-English users they were 3 and 3.06. This confirms that English users were more satisfied with the quality of translation.

The implication of these findings, with respect to English users, is that translation **from** English (translating English queries to French and German) and **to** English (translating French and/or German documents to English) was of good quality.

In order to understand the implication of these findings with respect to non-English users, the following has to be mentioned first: 16 out of 20 non-English users specified English as a language that they were familiar with, thus, the translation of English results was not required most of the time. Therefore, the findings reflect that translation between French and German (in both directions), may have been of lower quality than translations performed between the other

pairs (English-French and English-German). This may have been a contributing reason for why non-English users were less satisfied with the search engine in general.

Finally, question#3 focused more on the user's side: instead of making the translated results the subject of the question, the subject is actually switched to ask about the users themselves (whether **they** understood information that came from other languages or not). Median and mean for English users (4 and 4) and non-English users (4 and 3.94) reflected that they both were equally able to understand the information that came from other languages. This indicates that the quality of translation was at least sufficient to make the content of the translated documents comprehend-able.

5.5.5 Analysis: Open Questions and User Testimonies

The final section of the questionnaire involved open questions where users expressed their opinions in free text. The objective of administering these open questions was to capture additional details about the system and to allow users to express their opinions in a descriptive manner (as opposed to the Likert scale). This helps in gaining a deeper understanding of individual user impression of the system. The questions were as follows:

- What features or characteristics did you like most about the search engine?
- What features or characteristics did you like least about the search engine?
- The system currently provides the multilingual search service in English, French, and German. Are there any other languages that you would like the system to support in the future?
- If the search engine becomes available online for public use, when would you consider using it instead of your favourite search engine?
- Any additional comments or suggestions?

The following subsections present summary and analysis of the users' responses to these questions, and also quotes some chosen user testimonies. The discussions are grouped by theme/feature.

5.5.5.1 Multilingual Search and Related Features

The analysis of the users' comments revealed that 46 users indicated that they liked the feature of multilingual search and that they liked that results were translated to their preferred language. The users indicated that this enabled them to gain access to information beyond their

native/familiar languages. This demonstrates the usefulness of the notion of multilingual Web search and reflects the success of the PMIR system in delivering this kind of service.

On the other hand, three users indicated that the multilingual search feature became less useful when local information was being sought. This pertains to an interesting research direction, which is how to dynamically identify corpora that suit the nature of the query. This is further discussed in the future work reported in Section 6.4.

Regarding the notification that was displayed underneath each snippet in the result list to indicate its original language (i.e. the "*translated from*" notification), three users indicated that they liked being notified about the source language. On the other hand, it seems that some users did not notice the existence of this notification as four users indicated that they would have liked to be notified which results were translated. This suggests that this notification feature is useful and that it should perhaps be made bigger or displayed in a different place in order to be more noticeable to the users (e.g. to be placed beside the result's title instead of under the summary).

The following are some of the testimonies that are related to this discussion:
"*I like the ability to mash-up search results from different sources irrespective of the source content language and the language used in searching*".

"*I loved the delivery of multilingual results, it meant I was able to form opinions from content created in other languages. In my life, this is the first time I have ever searched for results in languages other than English and I think it's brilliant. I really found access to content from German and French pages brilliant. I would have loved more languages!*"

"*sometimes the results from other languages are irrelevant - for example if you are looking for information related to Ireland .. French language website did not seem to be providing good/useful information to the query*".

"*Liked the interleaving of results from different languages and the clear indication of which results were translated*".

"*It is not very clear whether the website proposed are translations or not. I think that, if a flag or something similar indicated the origin of translated information, it would have caught my attention*".

"*the fact that the languages were intertwined together was surprising at first but then enjoyable*".

5.5.5.2 Multiple Points of View vs. Redundancy

A degree of controversy was exhibited in the users' comments regarding what they thought about the existence of results from multiple languages that discussed more or less the same topic (i.e. rather similar content but not an exact duplicate in another language). This was considered a favourable feature when the additional content was perceived as complementary content. In contrast, this was considered as an unfavourable feature when the additional content was perceived as duplicated content. The following numbers demonstrate this controversy.

Twelve users indicated that they liked the fact that the multilingual search engine allowed them to compare opinions from different languages or different parts of the world about a certain topic, especially political topics. This suggests that a potentially useful implementation of the PMIR framework would be to provide a portal for opinionated results or news from different countries.

On the other hand, nine users pointed out that they considered the information redundant between languages. This suggests the following: (a) the need for a feature in the PMIR system that filters out multiple results from the same content provider or at least groups them together so that users can have the choice of whether to expand them or not; (b) the need for algorithms (and further research) in PMIR that determine the degree of similarity between results coming from different languages and filter out redundant information based on certain thresholds; and (c) the need to add information in the user model that indicates how far the user welcomes somewhat similar results, and use this information when adapting search results.

The following testimonies reflect the controversial opinions about this matter:
"*The transparent translation from multiple languages opens an amazing view on part of the information on the Web that I wouldn't have considered exploring before. It can be very useful for a researcher to see various views from various countries about political, cultural, business, general news among other things. For example, seeing how a piece of news is reported differently from one language to the other can be of interest to many people. This enlarged view can help a researcher to build a better picture of various global related topics*".

"*Multi-lingual results sometimes are redundant, e.g., Wikipedia pages*".

"*Nearly-repeated results from different languages is better to be removed or at least presented as a cluster together and a user can select the required language*".

"I liked that I could see some different interpretations of the same topics from the translated pages".

5.5.5.3 Relevance of Search Results

Regarding what the users thought about the degree of relevance of the search results, twelve users indicated that the relevance of search results should be improved. Two of those users specifically indicated that they were referring to the translated results. Since these comments were based on the usage of the baseline system, this reflects the need for personalisation algorithms to improve search results (the evaluation of which was presented in Section 5.4). It also reflects the need for accurate query translations. Those two factors can lead to better user satisfaction with the system.

In contrast to those opinions, three users commented that they were happy with the relevance of search results. This does not just reflect the natural phenomenon that users' opinions differ from each other, but may also generally reflect that part of the users' perception of a service depends on their expectations before engaging with it (i.e. subjectivity in opinions). For example, the search results may be perceived more relevant by a user who engages with the multilingual search engine with the pre-assumption that results that come from other languages are not likely to be useful. This also applies to translation; some users may perceive translation to be of higher or lower quality than it actually is, depending on their own expectations of translation quality before engaging with the system.

The following are some testimonies that are related to the relevance of search results:
"*I didn't like the ordering of the results. I found useful information in subsequent pages and less relevant results in the first page*".

"*relevant results and pages were generated, good translation, easy to use*".

"*I didn't like bringing information not related to the main search*".

"*very few results I faced that can be considered irrelevant to my search*".

"I suggest to increase the degree of topic relevance".

5.5.5.4 Translation

With respect to translation, nine users pointed out that translation quality needed improvement and seven users pointed out that some pages were not translated at all when they should have been (e.g. URLs that opened PDF documents). On the other hand, three users indicated that they were satisfied with the quality of translation and two users specifically commented that although translation was not good, it was of sufficient quality to convey the information in the page.

While these observations do not reflect a weakness in the PMIR system itself, they emphasise the crucial role of translation in the process and highlight that the success of a PMIR service is reliant on the accuracy of the chosen MT system.

The following are some testimonies about translation:
"*not accurate translations at some websites*".

"*most of the time, the translation wasn't good enough to easily understand the content*".

"*The quality of the translation is not great but it is more than adequate to understand the information*".

"*the translations were in some cases not accurate/understandable*".

"*I was able to interpret most of the translations but the overall quality could still be improved*".

5.5.5.5 Search Interface (GUI)

Regarding the interface of the search engine, 13 users indicated that the interface was intuitive and easy to use. This confirms the findings reported in earlier subsections and shows the success of the current implementation of the PMIR system in presenting the interleaved search results in a way that is familiar to the users.

On the other hand, two users indicated that the interface should be improved. This involved: (a) making the interface more informative; and (b) making it very clear to the users which results came from other languages.

The following are some testimonies concerning the interface:

"*It was easy to use and intuitive*".

"*clear and simple "Google-like" Interface*".

"*confusing interface because there was no feedback or information what was going on or which results had been translated and which had not been*".

"*It resembled my most commonly used search engine and output results in a very similar way and with similar fonts. Also it's functionality seemed very close to that engine and therefore it offered a very familiar interface and output to that engine. This made it very easy to use*".

5.5.5.6 Performance and Scaling Issues

Some user comments were related to performance issues where four users indicated the translation of pages was not fast enough and five users indicated that the responsiveness of the search engine should be improved. On the other hand, four users indicated that that they liked the responsiveness of the search engine. Furthermore, seven users indicated that they wanted the search engine to return more search results.

These comments reflect the importance of studying scaling issues if the PMIR system is to be released to the public as an online search tool (for Web search or other domains). Performance enhancements have to be considered for several components: (a) the controllers which manage the workflows; (b) the execution of the algorithms; (c) the underlying retrieval service; and (d) the underlying translation service. Moreover, performance can be significantly improved if adequate hardware is used to host the service[1]; hardware that is comparable to that of major search engines.

The following are some related testimonies:
"*the page loads in the native language first, and then after a while the page is suddenly translated. At first I thought it will remain in the native language and I was about to navigate away from the page*".

"*I would be happy to use this search engine, but some of the results were slow in returning*".

"*Very promising. Just need to retrieve more results".*

[1] For this experiment, the PMIR service (website) was hosted on a desktop machine with the following hardware configuration: Intel Core2 Quad CPU – 2.66 GHz, 4 GB RAM, 32-bit operating system.

"*It seemed to me to provide less links then I am used to receive from other search engine*".

"*I am very impressed with the search engine, extremely impressed even. If you were to offer performance and scale to Google level of user experience I would possibly switch*".

5.5.5.7 Other Features

Regarding the type of search results returned, six users indicated that they did not like the fact that the search engine returned textual results only, and would have liked to see other types of results (e.g. images, videos, maps, etc.). This is a limitation that was specifically imposed for this experiment because the processing and translation of metadata of multimedia items was out of the scope of this study. Two lessons can be learnt from these comments: (a) from an experiment administration perspective, the users should have been made aware at the beginning that the system only returned textual search results; and (b) from a service-delivery perspective, the search engine should aggregate multimedia items in the search results[1] in order to meet the users' expectations when comparisons with major search engines take place. The following are two testimonies that are related to this matter:

"*I would use the engine if other search types added like search in Maps, Images, Books*".

"*As a visual learner I find it desirable to have images included in my searches to help me conceptualise my topic. The addition of image searching may prove useful*".

A feature pointed out by a user, one that actually concerns Bing's translation of Web pages and not the framework itself, was that it was good that the translator kept the layout and formatting of the pages in place and just translated the windows of text inside the pages: "*I liked the presentation of the translated page in its original format. For instance, the page layout was kept but only text was replaced with the English content*". From a usability perspective, this is an important feature that multilingual search systems should have. Such usability lessons should be taken on board if alternative/extension implementations of the PMIR system are to be developed.

[1] The current implementation of the PMIR system supports the retrieval of multimedia items, however, the current implementation is specific to the types returned by the Bing search API. Furthermore, this feature has not been subject to enough testing.

5.5.5.8 Recommendations of Other Languages to Support

When users were asked about what languages they would recommend for the system to support, the most requested languages were: Spanish, Arabic, Chinese, Italian, Portuguese, and Japanese. A noteworthy point to highlight here is that these languages were not only requested from users who spoke/understood these languages, but also from users who wished to be able to gain access to information in these languages. This is a realisation of the third usage scenario discussed in Section 3.1: A monolingual (or multilingual) user who may show interest in topics or content that comes from languages that s/he does not understand. This evaluation showed that PMIR system facilitates seeking and gaining access to multilingual information and that it supports the individual user by providing a comprehensive personalised service that caters for the user's interests across multiple languages.

The following are some responses to the question about recommending additional languages for the system (from users who do not speak the designated languages):
"*What about Spanish or Portuguese? For some topics that like soccer that might be very important*".

"*Chinese; I read that almost the same number of pages in internet are in English and in Chinese*".

"*Maybe Italian or Spanish*".

"*Spanish would be useful and quite easy to add. Chinese/Japanese would be more difficult but might be more interesting for users to be able to get Asian perspectives on topics*".

"*Arabic, Japanese, Chinese (Far eastern languages)*".

"*Ideally, all languages. But maybe the most spoken languages (Chinese, Spanish) since it gives access to many culturally-linked information*".

5.5.5.9 Envisaged Scenarios for Using the Multilingual Web Search System

Finally, 48 users expressed their willingness to use the multilingual search engine if it becomes available online for public use. The following is a summary of the cases/scenarios that they

indicated in their responses. These recommendations highlight potential application areas for the PMIR system:

1. When they think that more results for the search will be in a language that they don't speak (e.g. international news, tourism information, etc.).
2. When they are searching for cross-cultural topics or topics that are specific to certain countries.
3. When they want to gain insight into varying points of view (from multiple countries) about a certain topic.
4. When searching for job or studying opportunities abroad.

The following are some related testimonies:
"*It can be very useful to see opinion variations in different countries especially in political and news domains*".

"*I would use this search engine when searching for specific information which I believe might be more extensively covered in other languages. I believe it would be very useful. An example would be to get real time information from foreign newspapers regarding a particular topic. I would like to source this information myself rather than having it filtered to me through English language publications. I believe this would be very important for Geo-Political issues such as Franco-German relations to see how the press on both sides are treating an issue*".

"*I am very fond of Google. I wouldn't use this as my primary engine unless I was dealing specifically with a language related problem or perhaps some multi-cultural investigation*".

"*I think integrating multi-lingual results can be extremely useful in certain areas (e.g. tourism) to the point that it could eventually become the primary method of searching*".

5.5.6 Limitations and Caveats

An important *caveat* to highlight about the qualitative evaluation carried out for translation quality is: there are quantitative metrics in research literature for evaluating MT quality, such as the BLEU score (Papineni et al., 2002). However, these metrics were not used in this experiment because the objective was actually to evaluate the **perceived usefulness** of the translated results with respect to the users. As discussed earlier, the aim was to demonstrate that even though MT may sometimes not be of very good quality, it may nevertheless be of sufficient quality to convey the information residing in the original document and provide satisfaction to the user's information need.

A limitation with the translation quality questions in this experiment is that the *Likert Scale* does not provide additional information about what was good or bad in regards to translation. For example, if a user responds with the answer of "*Not Sure*" to the question about whether translation quality was good or not, then what does this indicate? Does it indicate that some results were of good translation quality while others were of poor translation quality? Or does it indicate that translation was of medium quality overall? Since the evaluation of MT quality is not the specific focus of this study, this matter was not investigated any further.

5.5.7 Conclusions

This qualitative evaluation examined the users' perception of various aspects of the PMIR service, including: usability, multilinguality in search, and adequacy and quality of translation. The evaluation addressed the following challenge:

Challenge #5: *What is the users' perception of using a system that offers personalised search across multiple languages?*

The overall outcome of the evaluation indicates the users' positive perception of a personalised multilingual search system that adapts to their language capabilities. The users would welcome such service in a number of domains, provided that the features and performance are provided in a way that scales-up to the level of commercial search engines.

5.6 Summary of Evaluation

This chapter presented the evaluation carried out for this study. First, the need for multilingual search was demonstrated in a customer support case study that was carried out on help articles of the *Microsoft Office* product. Second, user search behaviour was investigated in light of multilinguality by carrying out an analysis of search logs from The European Library. This helped to gain insight into the way users from various linguistic backgrounds behave in multilingual search. Third, the investigation was followed by an exploratory experiment that demonstrated the efficacy of incorporating the attribute of language in the process of re-ranking library collections. Fourth, a set of quantitative evaluations were carried out which showed the effectiveness of the multilingual personalisation approaches proposed in this thesis. The evaluations also demonstrated the capability of the PMIR framework to provide a platform for delivering and evaluating PMIR services. Fifth, the success of the PMIR system, in the area of multilingual Web search, was demonstrated through qualitative evaluation of the usability and the various features of the system.

The critical analysis of PIR approaches in the literature review and the Microsoft case study provided the foundational aspects for the rest of the evaluation. Figure 29 shows a layout of the experiments and how each one is linked to one or more of the research challenges stated in Section 1.2. Together, the analysis and the evaluations contributed towards answering the research question of the thesis. This is discussed in the conclusion chapter (Chapter 6).

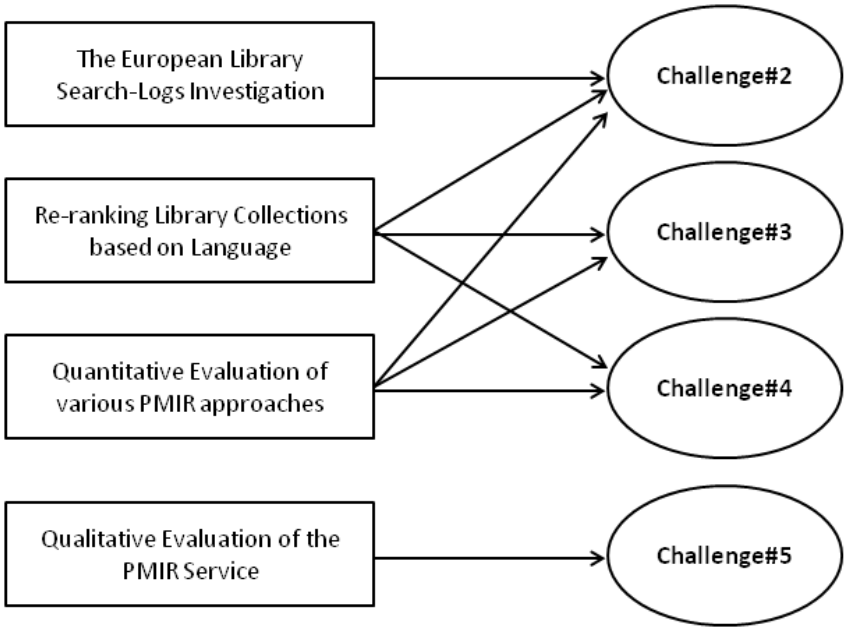

Figure 29: layout of experiments and challenges

Chapter 6: Conclusion and Future Work

This chapter concludes the thesis and discusses future research directions. The chapter is organised as follows. First, a summary of the work carried out for this thesis is given, along with a discussion of how this work met the objectives of the thesis. Second, a discussion of the contributions of this thesis and how the research question has been answered is provided, along with a discussion of the research impact of this study. Third, potential impact of this study on society and industry is highlighted, along with a discussion of potential application areas for the PMIR system. Finally, a discussion concerning research that can be carried forward from this thesis is provided.

6.1 Summary and Meeting the Objectives

Personalised Multilingual Information Retrieval entails **adapting search** to **users** of various linguistic profiles, given a **multilingual** environment. The study reported in this thesis addressed those four PMIR aspects: (1) multilinguality, (2) users, (3) search, and (4) personalisation (adapting to users). The aspects were addressed by considering them in relation to each other. Furthermore, a framework that unites these aspects was proposed in this thesis.

As a preliminary step, the study examined the usefulness of MIR in an industry case study (Section 5.1 [the Microsoft Office case study]). The case study demonstrated the need for MIR in a customer support scenario. It showed that enterprises can rely on MIR to extend their customer reach across international markets.

Regarding the first objective of the thesis (stated in Section 1.3), which is **gaining insight into users' search behaviour in light of multilinguality**, the study explored how users from different linguistic backgrounds and different language capabilities interact with multilingual search services (Section 5.2 [The European Library search-logs investigation] and Section 5.4 [the set of PMIR experiments]). The study showed that users exhibit language-dependent behavioural patterns and differences when interacting with a multilingual search service. Accordingly, the study showed evidence which support the recommendation that multilingual search providers devise multiple personalisation strategies that are tailored for certain languages or groups of languages.

Regarding the second objective, which is **investigating user modelling approaches that account for the aspect of multilinguality**, the study addressed this objective on two stages. First, with respect to incorporating the language attribute in the user model, the study conducted an experiment to investigate the efficacy of re-ranking document collections based on language (Section 5.3 [re-ranking collections of The European Library]). The experiment showed the benefit of taking into consideration the interface language selected by the user and the language of the user's query when displaying a list of document collections to the user. The experiment demonstrated the need for including language information in user models. Second, the study proposed and evaluated a set of user models that cater for multilinguality (Sections: 3.4., 4.2, and 5.4). This involved the inclusion of a set of language attributes in the user model: native language, preferred language, and a list of languages that the user is familiar with. Furthermore, this involved developing algorithms for mining search logs to infer the user's multilingual search interests. This also involved discussing three approaches to representing (structuring) the user model, namely: the Fragmented User Model, the Early-Combined User Model, and the Late-Combined User Model. The study showed that personalisation based on the Fragmented User Model was the most successful approach.

Regarding the third objective, which is **establishing a framework for evaluating the compartmentalisation and the combination of PMIR elements**, the study discussed the design and implementation of a framework for the delivery and evaluation of PMIR services (Sections: 3.6 and 4.1). As a prior step to designing the framework, a state-of-the-art survey of the literature was carried out. The survey featured an analysis of the stages and components of PIR and classifications of PIR approaches. It covered a variety of experimental systems and commercial systems in the fields of IR and MIR. The survey provided insights on how user and usage information is gathered, modelled, and employed for personalisation, and also provided insights on how PIR systems are evaluated. The findings from the survey formed the basis for this study in general and for the design and implementation of the PMIR framework in specific.

With regards to the framework's design, this thesis discussed the functional and non-functional requirements of the framework. This included a discussion of the framework's components, the inter-communication and workflow between components, the rationale behind each component, and the design considerations associated with each component. With regards to the framework's implementation, the thesis discussed the details of implementing the components of the framework, and how the functional and non-functional requirements were addressed. This also included a discussion of limitations and implementation issues associated with the components and with the use of external services for translation and retrieval of search results.

The study showed how the framework facilitated the delivery of a PMIR service in the context of multilingual Web search. It also showed how the framework facilitated both quantitative and qualitative evaluation of the features and components of the PMIR service (Sections: 5.4 and 5.5). The PMIR framework constituted the first deliverable of the thesis (stated in Section 1.4).

Regarding the final objective of the thesis, which is **improving retrieval effectiveness in MIR by means of personalised query adaptation and result adaptation algorithms**, the study proposed and evaluated a set of query adaptation algorithms and result adaptation algorithms for PMIR (Sections: 3.5, 4.3, 4.4, and 5.4 [the set of PMIR experiments]). These algorithms operated in conjunction with the proposed multilingual user models. The evaluation showed the improvements, in terms of retrieval effectiveness, that these algorithms achieved in isolation and in combination with each other. The evaluation also discussed the outcomes of applying these algorithms with users from different linguistic backgrounds. The set of proposed adaptation algorithms, together with the proposed user models, constitute the second deliverable of the thesis.

6.2 Contributions, Answering the Research Question, and Research Impact

The major contribution of this thesis is the PMIR framework; a platform that facilitates carrying out user trials and evaluations that concern a broad range of personalisation approaches in the field of PMIR. The study showed that the framework enabled the breaking up of the evaluation of PMIR into separate sections that are individually assessable as well as in combination with each other. In addition to PMIR, the framework can also be used to conduct experiments in the fields of IR, MIR, and PIR as it can be easily adjusted to deliver either a monolingual or a multilingual search service and can be easily configured to operate in either a personalised or a non-personalised mode. This flexibility facilitates a broad scope for reproducing and comparing work across IR subfields.

A second contribution of this study is a set of approaches for personalising multilingual search, the evaluation of which substantiates the notion of a PMIR framework. These approaches include multilingual representations of user models that cater for the user's multiple behavioural personas (facets) in search. They also include a family of algorithms for multilingual user-model construction, multilingual query adaptation based on user models, selective multilingual query adaptation based on user models, and multilingual result adaptation based on user models.

The research question posed in this thesis was (Section 1.2): ***What are the key considerations of evaluating the effect of a multilingual approach to search personalisation?***

The answer to question is as follows: in light of the abovementioned contributions, and in light of the PIR and PMIR stages discussed in Chapter 2 and Chapter 3, the key considerations are:

1. **On the process level:** the study has established that the consideration of multilinguality on the user's side and the content's side has a fundamental impact on the process of designing PIR systems and PIR experiments. This pertains to both the studying of *personalisation approaches to multilingual search* and the studying of *multilingual approaches to personalised search*. As a result of this impact, a need for a dedicated framework of operation for delivering and evaluating PMIR services is perceived; one that is aware of multilinguality from the ground-up and that orchestrates the flow of multilingual information throughout all the components of the process (especially with regards to user information and the information in search results).
2. **On the level of gathering and representing user and usage information:** the presence of multilingual search results, and accordingly multilingual search logs, affects how the user's search interests are mined from the logs and how they are represented. The study has shown how this alters the way user models are constructed and updated in order to capture the user's interests across languages.
3. **On the level of personalisation execution:** introducing a multilingual approach to search personalisation is perceived to have a profound effect on how query adaptation and result adaptation algorithms are designed, implemented, and evaluated. It also changes the way these algorithms interact with the underlying user models.
4. **On the technological level:** delivering and evaluating PMIR was confronted with scalability and quality issues, especially with regards to the performance of the translation and retrieval components. Concerning translation, PMIR involves extensive use of Machine Translation (translating queries, snippets, and whole Web pages) in various parts of the process (when retrieving/displaying search results, when constructing user models, and when performing the adaptation). Thus, the responsiveness of the translation service as well as the quality of translation become of significant importance, especially as the number of search results increases. Concerning retrieval, as both the number of search results and the number of languages that the system wishes to support increase, the responsiveness of the retrieval service becomes of significant importance, especially if the system wishes to provide a level of service that is comparable to modern-day search engines.
5. **On the strategic level:** research in the field of PMIR can make researchers and enterprises realise the relationship between the availability of content in different

languages and the way users from different linguistic or cultural backgrounds behave when searching for, and interacting with, that content. The outcome of the research reported in this thesis showed the need for devising different strategies for different groups of users depending on linguistic and cultural backgrounds/aspects. This adds a new dimension to *Localisation* in that it extends it to include adapting a process to different user groups in addition to adapting content to different user groups. This is perceived to have a significant effect on the way content providers design and implement their search services.

In conclusion, the investigation of search personalisation in a multilingual environment impacts research in the areas of User Modelling, Personalisation, Information Retrieval, and Digital Libraries. It introduces new research directions regarding how to model Web users, how to retrieve information that adequately satisfies their information needs, how to make this information accessible to them, and how to present it in the most convenient manner. The contribution of the research reported in this thesis to those research areas is demonstrated by the following selected publications (a full list of publications that are based on this study is given in Appendix-A):

- Ghorab, M. R., Zhou, D., O'Connor, A. & Wade, V. 2013. *Personalised Information Retrieval: Survey and Classification.* **User Modeling and User-Adapted Interaction (UMUAI)**, 23, 381-443.
- Ghorab, M. R., Zhou, D., Lawless, S. & Wade, V. 2012. *Multilingual User Modeling for Personalized Re-ranking of Multilingual Web Search Results*. **The 20th Conference on User Modeling, Adaptation, and Personalization (UMAP 2012)**. Montreal, Canada: Springer.
- Ghorab, M. R. 2011. *Improving Query and Result List Adaptation in Personalized Multilingual Information Retrieval***. The 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2011)**. Beijing, China: ACM, 1323-1324.
- Ghorab, M. R., Leveling, J., Lawless, S., O'Connor, A., Zhou, D., Jones, G. J. F. & Wade, V. 2011. *Multilingual Adaptive Search for Digital Libraries*. **International Conference on Theory and Practice of Digital Libraries (TPDL 2011)**. Berlin, Germany: Springer Berlin / Heidelberg, 244-251.

The following is a list of potential target venues for additional planned publications based on this thesis:

- The Journal of Information Retrieval (IR Journal).
- Information Processing and Management (IPM Journal).
- User Modeling and User-Adapted Interaction (UMUAI Journal).

6.3 Impact on Society and Industry and Potential Application Areas

In addition to the research impact, this study also has a potential impact on society and on industry. The potential impact of this study on society is in extending people's searches beyond familiar language, country, and culture. Not only does this provide a richer document base for them, but it may also help in bridging the gap between cultures. Increasing the penetration of information worldwide could help in broadening people's perspective on several world matters (e.g. Politics, Economy, Events, and so on). The potential societal impact also includes facilitating the process of seeking relevant information from within the plethora of information on the Web. This means that finding information on the Web becomes easier and faster, and that gaining access to it becomes more convenient.

The potential impact of this study on industry is enabling enterprises and content providers to reach a wider customer/audience base across the world at reduced costs. Furthermore, it allows content providers to focus more on the production of the content and less on how the content will be made accessible to the target audience. In addition to this, the study could serve as an initial description of a strategy for comparing models and approaches to PMIR as well as a set of guidelines for effective PMIR.

The PMIR system has the potential to be employed in diverse application areas and content domains. The following are examples of these areas/domains.

Enterprise Multilingual Customer Support. Following on the case study presented in this thesis, the PMIR system has the potential for commercialisation. It can be configured to operate on content within the technical customer support domain. In this case, the user model can be extended to include information about the user's technical knowledge and the products or services that s/he is interested in. Employing the PMIR system will help companies in gaining worldwide recognition by providing support that is personally tailored for each customer in

international markets. This can complement work that is carried out in this industrial domain within the CNGL[1] project (Steichen et al., 2011).

Digital Libraries. As demonstrated in this thesis, the PMIR system can be very useful to online digital libraries. In this case the user model can be extended to include information about items that the user has browsed/accessed (e.g. books, articles, multimedia items, etc.) and information about his/her favourite authors/artists.

Cultural Heritage. The PMIR system also lends itself well to the cultural heritage domain where users search and access information in historical documents (e.g. manuscripts, images, etc.) in multiple languages. The PMIR system can be configured to operate on the metadata of those documents, thus facilitating the access to cultural heritage online. This can contribute to projects like *tranScriptorium*[2] and *CULTURA*[3] (Hampson et al., 2012).

6.4 Future Research

This section discusses future work that can extend the work reported in this thesis.

Concept-based Multilingual User Models. An extension to the keyword-based multilingual user models proposed in this thesis could be to represent them in a concept-based manner. This involves using multilingual ontologies or multilingual Web taxonomies to represent the user's multilingual interests. This approach may lead to better matching between documents and the user's interests when re-ranking search results. This research direction generally falls under the area of semantic search (Fazzinga and Lukasiewicz, 2010).

Extending Selective Query Adaptation. This study proposed a selective query adaptation that makes adaptation decisions based on information from the user model. As discussed in Chapter 2, other approaches in the literature involve basing this selective decision on query/corpus features (Leveling and Jones, 2010a). A possible extension to this study is exploring the application of the two approaches together. The first stage may involve analysing the query itself to infer whether the query needs adaptation or not in the first place. The second stage may then involve consulting with the user model to determine if this is a recurring user interest and then select the expansion terms accordingly. This kind of study can be of concern to both monolingual search and multilingual search.

[1] http://www.cngl.ie/industry-commercialisation/cngl-for-industry/
[2] http://transcriptorium.eu/
[3] http://www.cultura-strep.eu/

Dynamic Selection of Target Corpora. With respect to the multilingual Web search service provided by the PMIR system, a useful research direction would be to study how to dynamically decide whether a query would benefit from multilingual search results or not. This has been investigated over one target language in (Hefny et al., 2011). An extension to this investigation would be identifying which language-corpora on the Web are likely to have pages that are more relevant to the given query.

6.5 Closing Statement

In conclusion, this thesis argues that the Web community has moved to a situation where global multilinguality is becoming a more important aspect of the users' daily interaction with information than ever before. Yet, research in the area of Personalised Multilingual Information Retrieval is still at an early stage. Research in this area should seek to enable users to achieve maximum benefit of information on the Web, beyond the barriers of language locale and country. Therefore, researchers ought to be looking at how personalised systems can be enhanced with two things in mind: the multilingual Web and the multilingual user. The consideration of this characteristic of multilinguality will have a profound effect on the way personalised systems gather, model, and employ 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.

References

Agichtein, E., Brill, E. & Dumais, S. 2006a. Improving Web Search Ranking by Incorporating User Behavior Information. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006). Seattle, Washington, USA: ACM, 19-26

Agichtein, E., Brill, E., Dumais, S. & Ragno, R. 2006b. Learning User Interaction Models for Predicting Web Search Result Preferences. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006). Seattle, Washington, USA: ACM, 3-10.

Agosti, M. 2011. Digital Libraries. *In:* Melucci, M. & Baeza-Yates, R. (eds.) *Advanced Topics in Information Retrieval.* Springer Berlin Heidelberg**,** 1-26.

Agosti, M. & Melucci, M. 2001. Information Retrieval on the Web. *In:* Agosti, M., Crestani, F. & Pasi, G. (eds.) *Lectures on Information Retrieval.* Springer Berlin Heidelberg**,** 242-285.

Amati, G., Carpineto, C. & Romano, G. 2004. Query Difficulty, Robustness, and Selective Application of Query Expansion. The 26th European Conference on Information Retrieval (ECIR 2004). Sunderland, U.K.: Springer, 127-137.

Ambati, V. & Uppuluri, R. 2006. Using Monolingual Clickthrough Data to Build Cross-lingual Search Systems. New Directions in Multilingual Information Access Workshop of SIGIR 2006. Seattle, Washington, USA: ACM.

Asnicar, F. A. & Tasso, C. 1997. ifWeb - a Prototype of User Model-Based Intelligent Agent for Document Filtering and Navigation in the World Wide Web. Adaptive Systems and User Modeling on the World Wide Web. Chia Laguna, Sardinia, 3-11.

Baeza-Yates, R. & Ribeiro-Neto, B. 2011. *Modern Information Retrieval: The Concepts and Technology behind Search (2nd Edition),* Addison-Wesley.

Bast, H., Majumdar, D. & Weber, I. 2007. Efficient Interactive Query Expansion with Complete Search. 16th ACM Conference on Information and Knowledge Management (CIKM 2007). Lisbon, Portugal: ACM, 857-860.

Belkin, N. J. & Croft, W. B. 1992. Information Filtering and Information Retrieval: Two Sides of the Same Coin? *Communications of the ACM,* 35**,** 29-38.

Billerbeck, B., Scholer, F., Williams, H. E. & Zobel, J. 2003. Query Expansion using Associated Queries. 12th International Conference on Information and Knowledge Management (CIKM 2003). New Orleans, LA, USA: ACM, 2-9.

Billsus, D. & Pazzani, M. 2007. Adaptive News Access. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer**,** 550-570.

Borlund, P. 2000. Experimental Components for the Evaluation of Interactive Information Retrieval Systems. *Journal of Documentation,* 56**,** 71-90.

Brajnik, G., Guida, G. & Tasso, C. 1987. User Modeling in Intelligent Information Retrieval. *Information Processing & Management,* 23**,** 305-320.

Brin, S. & Page, L. 1998. The Anatomy of a Large-scale Hypertextual Web Search Engine. 7th International World Wide Web Conference (WWW1998). Brisbane, Australia, 107-117.

Brooke, J. 1996. SUS-A Quick and Dirty Usability Scale. *In:* Jordan, P. W., Thomas, B., Weerdmeester, B. A. & McClelland, A. L. (eds.) *Usability Evaluation in Industry.* 189-194.

Brusilovsky, P. 2001. Adaptive Hypermedia. *User Modeling and User-Adapted Interaction,* 11**,** 87-110.

Brusilovsky, P. & Millán, E. 2007. User Models for Adaptive Hypermedia and Adaptive Educational Systems. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer**,** 3-53.

Brusilovsky, P. & Peylo, C. 2003. Adaptive and Intelligent Web-based Educational Systems. *International Journal of Artificial Intelligence in Education,* 13**,** 157-299.

Brusilovsky, P. & Tasso, C. 2004. Preface to Special Issue on User Modeling for Web Information Retrieval. *User Modeling and User-Adapted Interaction,* 14**,** 147-157.

Budzik, J. & Hammond, K. J. 2000. User Interactions With Everyday Applications as Context for Just-in-time Information Access. 5th International Conference on Intelligent User Interfaces (IUI 2000). New Orleans, Louisiana, USA: ACM, 44-51.

Callan, J. P., Croft, W. B. & Broglio, J. 1995. TREC and TIPSTER Experiments with INQUERY. *Information Processing & Management,* 31**,** 327-343.

Cao, G., Gao, J., Nie, J.-Y. & Bai, J. 2007. Extending Query Translation to Cross-Language Query Expansion with Markov Chain Models. 14th ACM International Conference on Information and Knowledge Management (CIKM 2007). Lisbon, Portugal: ACM, 351-360.

Cao, G., Nie, J.-Y., Gao, J. & Robertson, S. 2008. Selecting Good Expansion Terms for Pseudo-Relevance Feedback. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008). Singapore, Singapore: ACM, 243-250.

Carman, M. J., Baillie, M. & Crestani, F. 2008. Tag Data and Personalized Information Retrieval. Workshop on Search in Social Media (SSM at CIKM 2008). Napa Valley, California, USA: ACM, 27-34.

Carroll, J. M. & Rosson, M. B. 1987. The Paradox of the Active User. *In:* Carroll, J. M. (ed.) *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction.* Cambridge, MA: MIT Press**,** 80-111.

Chen, A. & Gey, F. C. 2004. Multilingual Information Retrieval Using Machine Translation, Relevance Feedback and Decompounding. *Information Retrieval,* 7**,** 149-182.

Chen, L. & Sycara, K. 1998. WebMate: A Personal Agent for Browsing and Searching. 2nd International Conference on Autonomous Agents. Minneapolis, Minnesota, United States: ACM, 132-139.

Chirita, P.-A., Firan, C., S. & Nejdl, W. 2007. Personalized Query Expansion for the Web. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007). Amsterdam, The Netherlands: ACM, 7-14.

Chowdhury, G. G. & Chowdhury, S. 1999. Digital Library Research: Major Issues and Trends. *Journal of Documentation,* 55**,** 409-448.

Conlan, O., Hockemeyer, C., Wade, V. & Albert, D. 2003. Metadata Driven Approaches to Facilitate Adaptivity in Personalized eLearning Systems. *Journal of the Japanese Society for Information and Systems in Education,* 1**,** 38-45.

Conlan, O. & Wade, V. 2004. Evaluation of APeLS – An Adaptive eLearning Service Based on the Multi-model, Metadata-Driven Approach. Lecture Notes in Computer Science. 3rd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2004). Eindhoven, The Netherlands: Springer Berlin / Heidelberg, 504-518.

Cui, H., Wen, J.-R., Nie, J.-Y. & Ma, W.-Y. 2003. Query Expansion by Mining User Logs. *IEEE Transactions on Knowledge and Data Engineering,* 15**,** 829-839.

De Bra, P., Aerts, A., Berden, B., De Lange, B., Rousseau, B., Santic, T., Smits, D. & Stash, N. 2003. AHA! The Adaptive Hypermedia Architecture. 14th ACM Conference on Hypertext and Hypermedia (Hypertext 2003). Nottingham, UK: ACM, 81-84.

de La Passardiere, B. & Dufresne, A. 1992. Adaptive Navigational Tools for Educational Hypermedia. *In:* Tomek, I. (ed.) *Computer Assisted Learning, Lecture Notes in Computer Science.* Springer Berlin / Heidelberg**,** 555-567.

De Luca, E. W. & Nürnberger, A. 2006. Adaptive Support for Cross-Language Text Retrieval. Lecture Notes in Computer Science. 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2006). Dublin, Ireland: Springer, 425-429.

Di Nunzio, G. M., Leveling, J. & Mandl, T. 2011. Multilingual Log Analysis: LogCLEF. *In:* Clough, P., Foley, C., Gurrin, C., Jones, G. F., Kraaij, W., Lee, H. & Mudoch, V. (eds.) *Advances in Information Retrieval.* Springer Berlin Heidelberg**,** 675-678.

Efthimiadis, E. N. 2000. Interactive Query Expansion: A User-based Evaluation in a Relevance Feedback Environment. *Journal of the American Society for Information Science,* 51**,** 989-1003.

Espinoza, F. & Höök, K. 1995. An Interactive Interface to an Adaptive Information System. User Modelling for Information Filtering on the World Wide Web Workshop. Hawaii, USA.

Fazzinga, B. & Lukasiewicz, T. 2010. Semantic Search on the Web. *Semantic Web,* 1**,** 89-96.

Ferro, N. & Peters, C. 2009. CLEF 2009 Ad Hoc Track Overview: TEL and Persian Tasks. *In:* Peters, C., Di Nunzio, G. M., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A. & Roda, G. (eds.) *Multilingual Information Access Evaluation I. Text Retrieval Experiments. LNCS.* Springer Berlin / Heidelberg**,** 13-35.

Furnas, G. W., Landauer, T. K., Gomez, L. M. & Dumais, S. T. 1987. The Vocabulary Problem in Human-System Communication. *Communications of the ACM,* 30**,** 964-971.

Ganguly, D., Leveling, J. & Jones, G. J. F. 2013. Overview of the Personalized and Collaborative Information Retrieval (PIR) Track at FIRE-2011. *In:* Majumder, P., Mitra, M., Bhattacharyya, P., Subramaniam, L. V., Contractor, D. & Rosso, P. (eds.) *Multilingual Information Access in South Asian Languages.* Springer Berlin Heidelberg**,** 227-240.

Gao, W., Niu, C., Nie, J.-Y., Zhou, M., Hu, J., Wong, K.-F. & Hon, H.-W. 2007. Cross-Lingual Query Suggestion Using Query Logs of Different Languages. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007). Amsterdam, The Netherlands: ACM, 463-470.

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 ed.: Springer**,** 54-89.

Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G. & Halatsis, C. 2007. Creating an Ontology for the User Profile: Method and Applications. Research Challenges in Information Science (RCIS 2007). Ouarzazate, Morocco, 407-412.

Gollapudi, S. & Sharma, A. 2009. An Axiomatic Approach for Result Diversification. 18th International Conference on World Wide Web (WWW 2009). Madrid, Spain: ACM, 381-390.

Hampson, C., Agosti, M., Orio, N., Bailey, E., Lawless, S., Conlan, O. & Wade, V. 2012. The CULTURA Project: Supporting Next Generation Interaction with Digital Cultural Heritage Collections. *In:* Ioannides, M., Fritsch, D., Leissner, J., Davies, R., Remondino, F. & Caffo, R. (eds.) *Progress in Cultural Heritage Preservation.* Springer Berlin Heidelberg**,** 668-675.

Hanani, U., Shapira, B. & Shoval, P. 2001. Information Filtering: Overview of Issues, Research and Systems. *User Modeling and User-Adapted Interaction,* 11**,** 203-259.

Harman, D. 1988. Towards Interactive Query Expansion. 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1988). Grenoble, France: ACM, 321-331.

Harman, D. 1992a. Relevance Feedback and Other Query Modification Techniques. *In:* Frakes, W. B. & Baeza-Yates, R. (eds.) *Information Retrieval.* Prentice-Hall, Inc.**,** 241-263.

Harman, D. 1992b. Relevance Feedback Revisited. 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1992). Copenhagen, Denmark: ACM, 1-10.

Harper, B. D., Slaughter, L. A. & Norman, K. L. 1997. Questionnaire Administration Via the WWW: A Validation & Reliability Study for a User Satisfaction Questionnaire. World Conference on the WWW and Internet. Toronto, Canada.

Haveliwala, T. H. 2002. Topic-sensitive PageRank. 11th International Conference on World Wide Web (WWW 2002). Honolulu, Hawaii, USA: ACM, 517-526.

Hefny, A., Darwish, K. & Alkahky, A. 2011. Is a Query Worth Translating: Ask the Users! 33rd European Conference on Information Retrieval (ECIR 2011). Dublin, Ireland: Springer Berlin / Heidelberg, 238-250.

Hothi, J. & Hall, W. 1998. An Evaluation of Adapted Hypermedia Techniques using Static User Modelling. 2nd Workshop on Adaptive Hypertext and Hypermedia. Pittsburgh, USA.

Hu, J. & Chan, P. K. 2008. Personalized Web Search by Using Learned User Profiles in Re-ranking. Workshop on Web Mining and Web Usage Analysis (WebKDD 2008). Las Vegas, Nevada, USA: ACM, 84-97.

Jain, S. C. (ed.) 2007. *Emerging Economies and the Transformation of International Buisness: Brazil, Russia, India And China,* Cheltenham, UK: Edward Elgar Publishing.

Jameson, A. 2008. Adaptive Interfaces and Agents. *In:* Sears, A. & Jacko, J. A. (eds.) *The Human-Computer Interaction Handbook: Fundamentals Evolving Technologies and Emerging Applications.* 2nd ed.: CRC Press.

Jansen, B. J., Spink, A. & Saracevic, T. 2000. Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *Information Processing and Management,* 36**,** 207-227.

Katakis, I., Tsoumakas, G., Banos, E., Bassiliades, N. & Vlahavas, I. 2009. An adaptive personalized news dissemination system. *Journal of Intelligent Information Systems,* 32**,** 191-212.

Kelly, D. & Teevan, J. 2003. Implicit Feedback for Inferring User Preference: A Bibliography. *SIGIR Forum,* 37**,** 18-28.

Kobayashi, M. & Takeda, K. 2000. Information Retrieval on the Web. *ACM Computing Surveys,* 32**,** 144-173.

Kobsa, A. 2007a. Generic User Modeling Systems. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer Berlin Heidelberg**,** 136-154.

Kobsa, A. 2007b. Privacy-Enhanced Web Personalization. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer**,** 628-670.

Koidl, K., Conlan, O. & Wade, V. 2011. Towards User-Centric Cross-Site Personalisation. *In:* Auer, S., Díaz, O. & Papadopoulos, G. (eds.) *Web Engineering.* Springer Berlin Heidelberg**,** 391-394.

Koutrika, G. & Ioannidis, Y. 2004. Rule-based Query Personalization in Digital Libraries. *International Journal on Digital Libraries,* 4**,** 60-63.

Lavrenko, V., Choquette, M. & Croft, W. B. 2002. Cross-lingual Relevance Models. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002). Tampere, Finland: ACM, 175-182.

Leff, A. & Rayfield, J. T. Year. Web-application development using the Model/View/Controller design pattern. *In:* 5th International Enterprise Distributed Object Computing Conference (EDOC 2001), 2001 2001 Seattle, WA. IEEE, 118-127.

Leveling, J. & Jones, G. J. F. 2010a. Classifying and Filtering Blind Feedback Terms to Improve Information Retrieval Effectiveness. Adaptivity, Personalization and Fusion of Heterogeneous Information (RIAO 2010). Paris, France: LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, 156-163.

Leveling, J. & Jones, G. J. F. 2010b. Query Recovery of Short User Queries: on Query Expansion With Stopwords. Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. Geneva, Switzerland: ACM, 733-734.

Liu, F., Yu, C. & Meng, W. 2004. Personalized Web Search for Improving Retrieval Effectiveness. *IEEE Transactions on Knowledge and Data Engineering,* 16**,** 28-40.

Magdy, W. & Jones, G. J. F. 2011. An Efficient Method for Using Machine Translation Technologies in Cross-Language Patent Search. 20th ACM International Conference on Information and Knowledge Management (CIKM 2011). Glasgow, Scotland, UK: ACM, 1925-1928.

Magennis, M. & van Rijsbergen, C. J. 1997. The Potential and Actual Effectiveness of Interactive Query Expansion. 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1997). Philadelphia, Pennsylvania, United States: ACM, 324-332.

Mandl, T., Agosti, M., Di Nunzio, G. M., Yeh, A., Mani, I., Doran, C. & Schulz, J. 2010. LogCLEF 2009: The CLEF 2009 Multilingual Logfile Analysis Track Overview. *In:* Peters, C., Di Nunzio, G. M., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A. & Roda, G. (eds.) *Multilingual Information Access Evaluation I. Text Retrieval Experiments.* Springer Berlin Heidelberg**,** 508-517.

Manning, C. D., Raghavan, P. & Schutze, H. 2008. *Introduction to Information Retrieval*, Cambridge University Press.

McCandless, M., Hatcher, E. & Gospodnetic, O. 2010. *Lucene in Action,* New York, NY, USA, Manning Publications.

McCarley, J. S. 1999. Should We Translate the Documents or the Queries in Cross-language Information Retrieval. 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics (ACL 1999). College Park, Maryland, USA: Association for Computational Linguistics, 208-214.

McNamee, P. & Mayfield, J. 2002. Comparing Cross-Language Query Expansion Techniques by Degrading Translation Resources. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002). Tampere, Finland: ACM, 159-166.

Mei, Q. & Church, K. 2008. Entropy of Search Logs: How Hard is Search? With Personalization? With Backoff? International Conference on Web Search and Web Data Mining (WSDM 2008). Palo Alto, California, USA: ACM, 45-54.

Micarelli, A., Gasparetti, F., Sciarrone, F. & Gauch, S. 2007. Personalized Search on the World Wide Web. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* 1 ed.: Springer**,** 195-230.

Micarelli, A. & Sciarrone, F. 2004. Anatomy and Empirical Evaluation of an Adaptive Web-Based Information Filtering System. *User Modeling and User-Adapted Interaction,* 14**,** 159-200.

Minack, E., Demartini, G. & Nejdl, W. 2009. Current Approaches to Search Result Diversification. 1st International Workshop on Living Web: Making Web Diversity a True Asset. Washington DC., USA.

Mulwa, C., Lawless, S., Ghorab, M. R., O'Donnell, E., Sharp, M. & Wade, V. 2011. A Framework for the Evaluation of Adaptive IR Systems through Implicit Recommendation. *In:* Andrews, S., Polovina, S., Hill, R. & Akhgar, B. (eds.) *Conceptual Structures for Discovering Knowledge. Lecture Notes in Computer Science.* Springer Berlin / Heidelberg**,** 366-374.

Nguyen, D., Overwijk, A., Hauff, C., Trieschnigg, D., Hiemstra, D. & de Jong, F. 2008. WikiTranslate: Query Translation for Cross-Lingual Information Retrieval Using Only Wikipedia. Lecture Notes in Computer Science. Cross-Language Evaluation Forum (CLEF 2008). Aarhus, Denmark: Springer, 58-65.

Nie, J.-Y. 2010. *Cross-Language Information Retrieval*, Morgan and Claypool Publishers.

Noll, M. & Meinel, C. 2007. Web Search Personalization Via Social Bookmarking and Tagging. 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC 2007). South Korea: Springer Berlin / Heidelberg, 367-380.

O'Connor, A., Lawless, S., Zhou, D., Jones, G. J. F. & Wade, V. 2009. Applying Digital Content Management to Support Localisation. *Localisation Focus,* 8**,** 39-52.

Oard, D. 1997. The State of the Art in Text Filtering. *User Modeling and User-Adapted Interaction,* 7**,** 141-178.

Oard, D. W. 1998. A Comparative Study of Query and Document Translation for Cross-Language Information Retrieval. 3rd Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup. Pennsylvania, USA: Springer-Verlag, 472-483.

Oard, D. W. 2010. Multilingual Information Access. *Encyclopedia of Library and Information Sciences, 3rd Edition***,** 3682-3687.

Oard, D. W. & Diekema, A. R. 1998. Cross-Language Information Retrieval. *In:* Williams, M. (ed.) *Annual Review of Information Science (ARIST).* 223-256.

Obrist, M., Geerts, D., Brandtz, P. B. & Tscheligi, M. 2008. Design for Creating, Uploading and Sharing User Generated Content. 26th ACM Conference on Human Factors in Computing Systems (CHI 2008). Florence, Italy: ACM, 2391-2394.

Ogilvie, P., Voorhees, E. & Callan, J. 2009. On the Number of Terms Used in Automatic Query Expansion. *Information Retrieval,* 12**,** 666-679.

Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C. & Johnson, D. 2005. Terrier Information Retrieval Platform. *In:* Losada, D. & Fernández-Luna, J. (eds.) *Advances in Information Retrieval.* Springer Berlin Heidelberg**,** 517-519.

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. 40th Annual Meeting on Association for Computational Linguistics (ACL 2002). Philadelphia, Pennsylvania: Association for Computational Linguistics, 311-318.

Pazzani, M. & Billsus, D. 2007. Content-Based Recommendation Systems. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer**,** 325-341.

Peters, C., Braschler, M. & Clough, P. 2012. *Multilingual Information Retrieval - from research to practice*, Springer Berlin Heidelberg.

Pitkow, J., Schutze, H., Cass, T., Cooley, R., Turnbull, D., Edmonds, A., Adar, E. & Breuel, T. 2002. Personalized Search. *Communications of the ACM,* 45**,** 50-55.

Pretschner, A. & Gauch, S. 1999. Ontology Based Personalized Search. 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 1999). Chicago, Illinois, USA: IEEE, 391-398.

Psarras, I. & Jose, J. 2006. A System for Adaptive Information Retrieval. Lecture Notes in Computer Science. 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2006). Dublin, Ireland: Springer Heidelberg, 313-317.

Qiu, F. & Cho, J. 2006. Automatic Identification of User Interest for Personalized Search. 15th International Conference on World Wide Web (WWW 2006). Edinburgh, Scotland: ACM, 727-736.

Rich, E. 1983. Users are Individuals: Individualizing User Models. *International Journal of Man-Machine Studies,* 18**,** 199-214.

Robertson, S. E., Walker, S., Jones, S., M.Hancock-Beaulieu, M. & Gatford, M. 1995. Okapi at TREC-3. 3rd Text REtrieval Conference (TREC-3). 109-126.

Ruthven, I. 2003. Re-examining the Potential Effectiveness of Interactive Query Expansion. 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2003). Toronto, Canada: ACM, 213-220.

Ruthven, I. & Lalmas, M. 2003. A Survey on the Use of Relevance Feedback for Information Access Systems. *The Knowledge Engineering Review,* 18**,** 95-145.

Ruvini, J.-D. 2003. Adapting to the User's Internet Search Strategy. Lecture Notes in Computer Science. 9th International Conference on User Modeling (UM 2003). Johnstown, Pennsylvania, USA: Springer, 55-64.

Ryan, L., Anastasiou, D. & Cleary, Y. 2009. Using Content Development Guidelines to Reduce the Cost of Localising Digital Content. *Localisation Focus,* 8**,** 11-28.

Salton, G. & Buckley, C. 1990. Improving Retrieval Performance by Relevance Feedback. *Journal of the American Society for Information Science,* 41**,** 288-297.

Santos, R. L. T., Macdonald, C. & Ounis, I. 2010. Exploiting Query Reformulations for Web Search Result Diversification. 19th International Conference on World Wide Web (WWW 2010). Raleigh, North Carolina, USA: ACM, 881-890.

Schafer, J. B., Frankowski, D., Herlocker, J. & Sen, S. 2007. Collaborative Filtering Recommender Systems. *In:* Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.) *The Adaptive Web.* Springer**,** 291-324.

Shen, X., Tan, B. & Zhai, C. 2005. Implicit User Modeling for Personalized Search. 14th ACM International Conference on Information and Knowledge Management (CIKM 2005). Bremen, Germany: ACM, 824-831.

Si, L. & Callan, J. 2005. CLEF 2005: Multilingual Retrieval by Combining Multiple Multilingual Ranked Lists. Cross-Language Evaluation Forum (CLEF 2005). Vienna, Austria: Springer Berlin / Heidelberg, 121-130.

Silvestri, F. 2010. Mining Query Logs: Turning Search Usage Data into Knowledge. *Foundations and Trends in Information Retrieval,* 4**,** 1-174.

Smyth, B. & Balfe, E. 2006. Anonymous Personalization in Collaborative Web Search. *Information Retrieval,* 9**,** 165-190.

Speretta, M. & Gauch, S. 2005. Personalized Search based on User Search Histories. IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005). Compiegne University of Technology, France, 622-628.

Stamou, S. & Ntoulas, A. 2009. Search Personalization Through Query and Page Topical Analysis. *User Modeling and User-Adapted Interaction,* 19**,** 5-33.

Stefani, A. & Strapparava, C. 1998. Personalizing Access to Web Sites: The SiteIF Project. 2nd Workshop on Adaptive Hypertext and Hypermedia Pittsburgh, Pennsylvania, USA.

Stefani, A. & Strapparava, C. 1999. Exploiting NLP Techniques to Build User Model for Web Sites: the Use of WordNet in SiteIF Project. 2nd Workshop on Adaptive Systems and User Modeling on the World Wide Web. Toronto, Canada.

Steichen, B., Lawless, S., O'Connor, A. & Wade, V. 2009. Dynamic Hypertext Generation for Reusing Open Corpus Content. 20th ACM Conference on Hypertext and Hypermedia (Hypertext 2009). Torino, Italy: ACM, 119-128.

Steichen, B., O'Connor, A. & Wade, V. 2011. Personalisation in the Wild: Providing Personalisation Across Semantic, Social and Open-Web Resources. 22nd ACM Conference on Hypertext and Hypermedia (Hypertext 2011). Eindhoven, The Netherlands: ACM, 73-82.

Stroppa, N. & Way, A. 2006. MaTrEx: the DCU Machine Translation System for IWSLT 2006. International Workshop on Spoken Language Translation (IWSLT 2006). Kyoto, Japan., 31-36.

Sugiyama, K., Hatano, K. & Yoshikawa, M. 2004. Adaptive Web Search Based on User Profile Constructed without Any Effort from Users. 13th International Conference on World Wide Web (WWW 2004). New York, USA: ACM, 675-684.

Sun, J.-T., Zeng, H.-J., Liu, H., Lu, Y. & Chen, Z. 2005. CubeSVD: A Novel Approach to Personalized Web Search. 14th International Conference on World Wide Web (WWW 2005). Chiba, Japan: ACM, 382-390.

Teevan, J., Dumais, S. T. & Horvitz, E. 2005. Personalizing Search via Automated Analysis of Interests and Activities. 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2005). Salvador, Brazil: ACM, 449-456.

Teevan, J., Dumais, S. T. & Horvitz, E. 2010. Potential for Personalization. *ACM Transactions on Computer-Human Interaction,* 17**,** 1-31.

Teevan, J., Dumais, S. T. & Liebling, D. J. 2008. To Personalize or Not to Personalize: Modeling Queries with Variation in User Intent. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008). Singapore, Singapore: ACM, 163-170.

Teevan, J., Morris, M. R. & Bush, S. 2009. Discovering and Using Groups to Improve Personalized Search. 2nd ACM International Conference on Web Search and Data Mining (WSDM 2009). Barcelona, Spain: ACM, 15-24.

Tsai, M.-F., Wang, Y.-T. & Chen, H.-H. 2008. A Study of Learning a Merge Model for Multilingual Information Retrieval. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008). Singapore, Singapore: ACM, 195-202.

Vallet, D., Cantador, I. & Jose, J. 2010. Personalizing Web Search with Folksonomy-Based User and Document Profiles. 32nd European Conference on Information Retrieval (ECIR 2010). Milton Keynes, UK: Springer Berlin / Heidelberg, 420-431.

van Genabith, J. 2009. Next Generation Localisation. *Localisation Focus,* 8**,** 4-10.

Vassiliou, C., Stamoulis, D., Spiliotopoulos, A. & Martakos, D. 2003. Creating Adaptive Web Sites using Personalization Techniques: a Unified, Integrated Approach and the Role of Evaluation. *In:* Patel, N. V. (ed.) *Adaptive evolutionary information systems.* IGI Publishing**,** 261-285.

Wade, V. 2009. Challenges for the Multi-dimensional Personalised Web. *In:* Houben, G.-J., McCalla, G., Pianesi, F. & Zancanaro, M. (eds.) *Proceedings of User Modeling, Adaptation, and Personalization Conference (UMAP 2009). Lecture Notes in Computer Science.* Springer Berlin / Heidelberg**,** 3-3.

Wainer, J., Xavier, E. C. & Bezerra, F. 2009. Scientific Production in Computer Science: A Comparative Study of Brazil and Other Countries. *Scientometrics Journal,* 81**,** 535-547.

White, R. W., Ruthven, I. & Jose, J. M. 2002. The Use of Implicit Evidence for Relevance Feedback in Web Retrieval. Lecture Notes in Computer Science. 4th BCS-IRSG European Colloquium on IR Research (ECIR 2002). Glasgow, UK: Springer, 449-479.

Witten, I. H., Frank, E. & Hall, M. A. 2011. *Data Mining: Practical Machine Learning Tools and Techniques (3rd Edition),* Morgan Kaufmann.

Xu, J. & Croft, W. B. 1996. Query Expansion Using Local and Global Document Analysis. 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1996). Zurich, Switzerland: ACM, 4-11.

Yin, Z., Shokouhi, M. & Craswell, N. 2009. Query Expansion Using External Evidence. Lecture Notes In Computer Science. 31st European Conference on Information Retrieval (ECIR 2009). Toulouse, France: Springer, 362-374.

Zhang, H., Song, Y. & Song, H.-t. 2007. Construction of Ontology-Based User Model for Web Personalization. Lecture Notes in Computer Science. 11th International Conference on User Modeling (UM 2007). Corfu, Greece, 67-76.

Appendix-A: List of Publications Based on this Study

1. Ghorab, M. R., Zhou, D., O'Connor, A. & Wade, V. 2013. **Personalised Information Retrieval: Survey and Classification**. *User Modeling and User-Adapted Interaction (UMUAI)*, 23, 381-443.
2. Ghorab, M. R., Lawless, S., O'Connor, A., Zhou, D. & Wade, V. 2013. **Multilingual vs. Monolingual User Models for Personalized Multilingual Information Retrieval**. *In: 21st Conference on User Modeling, Adaptation and Personalization (UMAP 2013)*. Rome, Italy: Springer Berlin Heidelberg, 356-358.
3. Ghorab, M. R., Zhou, D., Lawless, S. & Wade, V. 2012. **Multilingual User Modeling for Personalized Re-ranking of Multilingual Web Search Results**. *In: 20th Conference on User Modeling, Adaptation, and Personalization (UMAP 2012)*. Montreal, Canada: Springer.
4. Ghorab, M. R., Zhou, D., Steichen, B. & Wade, V. 2011. **Towards Multilingual User Models for Personalized Multilingual Information Retrieval.** *In: 1st Workshop on Personalised Multilingual Hypertext Retrieval (PMHR 2011)*. Eindhoven, The Netherlands: ACM, 42-49.
5. Ghorab, M. R., Leveling, J., Lawless, S., O'Connor, A., Zhou, D., Jones, G. J. F. & Wade, V. 2011. **Multilingual Adaptive Search for Digital Libraries**. *In: International Conference on Theory and Practice of Digital Libraries (TPDL 2011)*. Berlin, Germany: Springer Berlin / Heidelberg, 244-251.
6. Ghorab, M. R. 2011. **Improving Query and Result List Adaptation in Personalized Multilingual Information Retrieval**. *In: 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2011)*. Beijing, China: ACM, 1323-1324.
7. Leveling, J., Ghorab, M. R., Magdy, W., Jones, G. J. F. & Wade, V. 2010. **DCU-TCD@LogCLEF 2010: Re-ranking Document Collections and Query Performance Estimation.** *In: LogCLEF 2010 Workshop at Conference on Multilingual and Multimodal Information Access Evaluation (CLEF) 2010*. Padua, Italy.
8. 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. M., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A. & Roda, G. (eds.) Lecture Notes in Computer Science (6241/2010), Multilingual Information Access Evaluation I*. Text Retrieval Experiments. Springer, 518-525.

9. Ghorab, M. R., Zhou, D., O'Connor, A. & Wade, V. 2009. **A Framework for Cross-Language Search Personalization**. *In: 4th International Workshop on Semantic Media Adaptation and Personalization (SMAP 2009)*. San Sebastian, Spain, 15-20.
10. Ghorab, M. R., Leveling, J., Zhou, D., Jones, G. J. F. & Wade, V. 2009. **TCD-DCU at LogCLEF 2009: An Analysis of Queries, Actions, and Interface Languages**. *In: Working Notes of the Cross-Language Evaluation Forum (CLEF 2009) Workshop*. Corfu, Greece.

Appendix-B: Search Tasks of the PMIR Experiment

Tasks in English

(displayed to users who selected English as their preferred language)

Task 1: Art:

Artists all over the world have created great works of art throughout the years like paintings, music, novels, films, etc. Write a few lines about a famous work of art and the person who created it, and the story behind its creation (e.g. how the person was inspired to create that work).

Task 2: Science and Technology:

The last few centuries have witnessed a lot of scientific achievements and inventions in many fields. Write a few lines about a scientific discovery, an invention, or a technological advancement that was achieved by a certain country of your choice.

Task 3: Nature:

Write a few lines about the wonders of nature in a certain part of the world. For example, you can write about any of the following: seas, oceans, mountains, deserts, forests, etc. in any country or continent.

Task 4: Political Dispute:

Sometimes disputes happen between two countries for various reasons (e.g. dispute over land, resources, etc.). People in those two countries, and also people from other countries, may have different opinions about the nature of the dispute and how it should be solved. Write a few lines about a dispute that took place between two countries/nations in recent or ancient history, and how the peoples' points of view were different from each other.

Task 5: Environment:

Different countries have different natural resources (e.g. diamonds, gold, iron, gas, etc.). Countries utilize their resources in a number of ways, for example by developing certain industries around them or by exporting them to other countries. Write a few lines about a country that is famous of a certain resource and how it utilizes it.

Task 6: Health:

Sometimes a disease (e.g. virus) may break out in a certain region or country and it affects the people in many ways. Governments implement different mechanisms to deal with and control such events. Moreover, sometimes the World Health Organization (W.H.O.) gets involved in preventing the disease from spreading out further. Write a few lines about how people in a certain region or country were affected by a disease outbreak and how the government and/or the World Health Organization dealt with it.

Task 7: Sports:

People who are very passionate about international sports events (e.g. Football, Basketball, Tennis, The Olympics, etc.) may sometimes do extraordinary or bizarre things when their national teams/contestants win or lose a game/competition. Write a few lines about such incident that was associated with an international sports event that took place in a country other than your country of origin or the country you live in at the moment.

Task 8: Culture:

Different countries have different cultural habits and traditions. These may be general habits/traditions or ones that are associated with certain things (e.g. eating or drinking habits, greeting customs, marriage customs, etc.). Write a few lines about special traditions in a country of your choice (a country other than your country of origin or the country that you live in at the moment).

Task 9: Economy:

Governments allocate different percentages of their resources and income to different sectors in a country (e.g. budget for health sector, budget for education, budget for military, etc.). This can be met with agreement or disagreement from the people and politicians of the country. Select a country (other than your country of origin or the country that you currently live in) and write a few lines about their allocated budget in one or more sectors, and how the people or critics feel about this allocation.

Task 10: Political Event:

Events, such as revolutions, protests, or military coups affect countries in many ways (politically, economically, socially, etc.). Write a few lines about such event that happened in a country in recent history and how it affected the country in which it took place (please select a country other than your country of origin or the country that you currently live in).

Task 11: Tourism:

Many people like to travel to different countries for tourism. This may be for site-seeing tourism or therapeutic (medical) tourism. Write a few lines about any kind of tourism in a country of your choice (a country other than your country of origin or the country that you live in at the moment).

Tasks in French

(displayed to users who selected French as their preferred language)

Tâche 1: Art:

Des artistes du monde entier ont créé de grandes œuvres d'art au fil des années comme des peintures, de la musique, des romans, des films, etc. Écrivez quelques lignes sur une célèbre œuvre d'art et sur la personne qui l'a créée ainsi que l'histoire de sa création (par exemple comment la personne était inspirée pour créer cette œuvre).

Tâche 2: Sciences et Technologie:

Les derniers siècles ont donné lieu à beaucoup de réalisations scientifiques et inventions dans de nombreux domaines. Écrivez quelques lignes sur une découverte scientifique, une invention ou un progrès technologique qui a été réalisé par un pays de votre choix.

Tâche 3: Nature:

Écrivez quelques lignes sur les merveilles de la nature dans une certaine partie du monde. Par exemple, vous pouvez écrire sur n'importe lequel des éléments suivants : mers, océans, montagnes, déserts, forêts, etc., dans un pays ou un continent de votre choix.

Tâche 4: Différend Politique:

De temps en temps, certains litiges peuvent se produire entre deux pays pour des raisons diverses (par exemple les différends conernant les terres, les ressources, etc.). Les citoyens de ces deux pays ainsi que ceux d'autres pays, peuvent avoir des opinions différentes sur la nature du différend et comment il devrait être résolu. Écrivez quelques lignes sur un différend qui a eu lieu entre deux pays/nations au cours de l'histoire (recente ou ancienne), et comment les points de vue de citoyens étaient différents les uns des autres.

Tâche 5: Environnement:

Différents pays ont différentes ressources naturelles (ex:, diamants, or, fer, gaz, etc.). Chaque pays utilise ses ressources de differentes manieres, par exemple en développant certaines industries autour de ces resources ou en les exportant vers d'autres pays. Écrivez quelques lignes sur un pays en particulier, connu pour une certaine ressource, et comment cette resource est utilise.

Tâche 6: Santé:

Parfois une maladie (p. ex. virus) peut-être apparaitre dans une région ou un pays, et cela affecte les gens de differentes manieres. Les gouvernements mettent en place différents mécanismes pour gérer et contrôler de tels événements. Quelque fois, l'Organisation mondiale de la santé (O.M.S.) s'implique dans la prévention de la maladie afin qu'elle ne se propage pas d'avantage. Écrivez quelques lignes concernant des personnes dans une région ou un pays touchés par une épidémie et comment le gouvernement et/ou l'Organisation mondiale de la santé a gérer cette situation.

Tâche 7: Événement sportif:

Les gens très passionnés par les manifestations sportives internationales (football par exemple, basket, tennis, les Jeux olympiques, etc) peuvent parfois faire des choses extraordinaires ou bizarre quand leurs equipes nationales / participants triomphent ou perdent un match / une compétition. Écrivez quelques lignes sur ce genre d'incident qui a été associée à un événement sportif international qui a eu lieu dans un pays autre que votre pays d'origine ou le pays dans lequel vous vivez à l'heure actuelle.

Tâche 8: Culture:

Différents pays possedent différentes habitudes culturelles et traditions. Ces habitudes peuvent être générales ou associées à certaines choses (par exemple des habitudes alimentaires, des coutumes de salutation, des coutumes de mariage, etc.) Ecrivez quelques lignes sur des traditions particulières dans un pays de votre choix (un pays autre que votre pays ou le pays où vous vivez en ce moment).

Tâche 9: Economie:

Les gouvernements allouent des pourcentages différents de leurs ressources et revenus dans différents secteurs d'un pays (par exemple pour le budget du secteur de la santé, le budget de l'éducation, le budget militaire, etc.) Ces decisions peut être rencontré avec l'accord ou le désaccord de la population et des politiciens du pays. Sélectionnez un pays (autre que votre pays ou le pays dans lequel vous vivez en ce moment) et ecrivez quelques lignes sur les chiffres ou les pourcentages budgetaires dans les pays sélectionnés, et comment le peuple / les critiques / les politiciens de ces pays ont percu ces allocations budgétaires.

Tâche 10: Événement politique:

Des événements, comme des révolutions, des manifestations, ou des coups d'etats militaires affectent des pays de différente manieres (politiquement, économiquement, socialement, etc.) Ecrivez quelques lignes sur un tel événement survenu dans un pays dans l'histoire récente et comment cet evenement a touché le pays dans lequel il a eu lieu (un pays autre que le votre ou le pays où vous vivez en ce moment).

Tâche 11: Tourisme:

Beaucoup de gens aiment voyager vers différents pays pour le tourisme. Cela peut être pour admirer le paysage touristique ou pour le tourisme thérapeutique (médical). Ecrivez quelques lignes sur un type de tourisme dans un pays de votre choix (un pays autre que votre pays ou le pays où vous vivez en ce moment).

Tasks in German

(displayed to users who selected German as their preferred language)

Aufgabe 1:Kunst:

Künstler auf der ganzen Welt haben große Kunstwerke im Laufe der Jahre wie Malerei, Musik, Romane, Filme, etc. erstellt. Schreiben Sie einige Sätze über einen berühmtes Kunstwerk und die Person, die es erstellt hat, und die Geschichte hinter seiner Erschaffung (z.B. was hat die Person dazu inspiriert, dieses Kunstwerk zu erschaffen).

Aufgabe 2: Wissenschaft und Technologie:

Im letzten Jahrhunderte kamen viele wissenschaftliche Leistungen und Erfindungen in vielen Bereichen zum Vorschein. Schreiben Sie einige Sätze über eine wissenschaftliche Entdeckung, Erfindung oder eine technologische Weiterentwicklung, die von einem bestimmten Land Ihrer Wahl erreicht wurde.

Aufgabe 3: Natur:

Schreiben Sie einige Sätze über die Wunder der Natur in einem bestimmten Teil der Welt. Sie können z. B. über eines der folgenden Themen schreiben: Meere, Ozeane, Berge, Wüsten, Wälder, etc. in einem beliebigen Land oder Kontinent.

Aufgabe 4: politischen Auseinandersetzung:

Manchmal passieren Streitigkeiten zwischen beiden Ländern aus verschiedenen Gründen (z. B. Streit um Land, Ressourcen, etc..). Menschen in diesen beiden Ländern und auch Leute aus anderen Ländern haben möglicherweise unterschiedliche Meinungen über die Natur des Problems und wie es gelöst werden sollte. Schreiben Sie ein paar Zeilen über einen Disput, der fand zwischen zwei Länder/Nationen in den letzten oder alte Geschichte und wie die Völker Gesichtspunkten voneinander unterscheiden.

Aufgabe 5: Umwelt:

Unterschiedliche Länder haben unterschiedliche natürliche Ressourcen (z.B. Diamanten, Gold, Eisen, Gas, etc.). Länder nutzen Sie ihre Ressourcen durch eine Reihe von Möglichkeiten, z. B. durch die Entwicklung bestimmter Industries um sie herum oder indem Sie sie in andere Länder exportieren. Schreiben Sie ein paar Zeilen über ein Land, das berühmt ist für eine bestimmte Ressource und wie es diese nutzt.

Aufgabe 6: Gesundheit:

Manchmal bricht ein Krankheit (z.B. Virus) in einer bestimmten Region oder einem Land aus und beinflusst die Menschen in vielerlei Hinsicht. Regierungen implementieren unterschiedliche Mechanismen um solche Ereignisse zu steuern. Darüber hinaus involviert sich machmals die Welt Gesundheit Organisation (W.H.O) um eine Ausbreitung der Krankheit zu verhindern. Schreiben Sie ein paar Zeilen über wie Menschen in einer bestimmten Region oder eines Landes durch einen Ausbruch der

Krankheit betroffen waren und wie die Regierung bzw. die Weltgesundheitsorganisation damit umgegingen.

Aufgabe 7: Sport-Ereignis:

Menschen, die sehr leidenschaftlich über international Sportveranstaltungen (zB Fußball, Basketball, Tennis, Olympische Spiele, etc.) sind, machen manchmal außergewöhnliche oder bizarre Dinge, wenn ihre nationalen Teams / Teilnehmer ein Spiel oder Wettkampf gewinnen oder verlieren. Schreiben Sie ein paar Zeilen über solche Vorfälle, die einem internationalem Sportereignis zugeordnet sind, und in einem anderen Land als Ihrem Herkunftsland oder das Land, das Sie im Augenblick zu Leben , passierten.

Aufgabe 8: Kultur:

Verschiedene Länder weisen unterschiedliche kulturelle Gewohnheiten und Traditionen auf. Diese können allgemeine Gewohnheiten / Traditionen betreffen, oder mit bestimmten Dingen (z.B. Essen oder Trinkgewohnheiten, Grußbräuche, Hochzeitsbräuche, etc.) zugeordnet sein. Schreiben Sie einige Sätze über besondere Traditionen in einem Land Ihrer Wahl (einem anderen Land als Ihrem eigenen Land/das Land, in dem Sie im Moment leben).

Aufgabe 9: Wirtschaft:

Der Staat allokiert unterschiedliche Prozentsätze seiner Ressourcen und Einkommen zu den verschiedenen Sektoren in einem Land (zB Budget für Gesundheitswesen, Budget für Bildung, Budget für das Militär, etc.). Dies kann mit Zustimmung oder Ablehnung von den Menschen und Politiker des Landes aufgenommen werden. Wählen Sie ein Land , in der Sie derzeit nicht und das nicht Ihr Ursprungsland ist, und schreiben Sie bitte ein paar Zeilen über das zugewiesene Budget (Haushalt) in einem oder mehreren Sektoren, und wie sich die Menschen oder die Kritiker über diese Aufteilung fühlen.

Aufgabe 10: Politischen Ereignisses:

Veranstaltungen, wie Revolutionen, Proteste oder militärische Staatsstreiche betreffen Länder in vielerlei Hinsicht (politisch, wirtschaftlich, sozial, etc.). Schreiben Sie einige Sätze zu einem solchen Fall, welcher in einem Land in der jüngeren Geschichte passierte und wie dies das betroffene Land beeinflusst hat (Bitte wählen Sie einem anderen Land als Ihrem Heimatland oder dem Land, in der Sie derzeit Leben).

Aufgabe 11: Tourismus:

Viele Menschen reisen gerne in verschiedene Länder als Touristen. Dies ist oft entweder für Site-Seeing-Tourismus oder therapeutischen (medizinischen) Tourismus. Schreiben Sie einige Sätze über jede Art von Tourismus in einem Land Ihrer Wahl (einem anderen Land als Ihrem eigenen Land/das Land, in dem Sie im Moment leben).

Appendix-C: Questionnaire of the PMIR Experiment

#	Question	Strongly Dis-agree	Dis-agree	Not Sure	Agree	Strongly Agree
1	I found the search system returned relevant results to my queries					
2	Many of the results were irrelevant to my query					
3	The presentation of interleaved (mixed) results from different languages was useful					
4	The system returned results that were helpful in solving the search task					
5	I think the mixing of multilingual results was confusing					
6	The system encouraged me to explore information coming from languages other than my native/preferred language					
7	I found that a lot of information was redundant between languages					
8	I had to search a lot before I was able to find useful content					
9	I think I did well in solving the tasks					
10	The translated results were less helpful in solving the tasks than other non-translated results.					
11	The quality of the translation was good					
12	I was able to understand the information that came from languages other than my native/preferred language.					
13	I think that I would like to use this search engine frequently					
14	I found the search engine unnecessarily complex					
15	I thought the search engine was easy to use					
16	I think that I would need the support of a technical person to be able to use this search engine					
17	I found the various functions in this search engine were well integrated					
18	I thought there was too much inconsistency in this search engine					
19	I would imagine that most people would learn to use this search engine very quickly					
20	I found the search engine very cumbersome (awkward) to use					
21	I felt very confident using the search engine					
22	I needed to learn a lot of things before I could get going with this search engine					
23	What features or characteristics did you like most about the search engine?	*open text question*				
24	What features/characteristics did you like least about the search engine?	*open text question*				
25	The search engine currently provides the multilingual search service in English, French, and German. Are there any other languages that you would like the system to support in the future?	*open text question*				
26	If the search engine becomes available online for public use, when would you consider using it instead of your favourite search engine?	*open text question*				
27	Any additional comments or suggestions?	*open text question*				

Appendix-D: Configuration File of the PMIR Framework

```
# This file holds main configuration parameters for the PMIR
framework. It is divided into sections (marked with '#') and
subsections underneath each section (marked with '##' or '###'). The
values of some properties are omitted for privacy/security reasons

#Flags (to enable/disable various parts in the PMIR workflow)

##Specifies if the result page should open in a new window when the
user clicks on it.
Default-Open-Results-In-New-Window = false
##Generally specifies whether to perform query adaptation or not
Query-Adaptation-Enabled = true
##Specifies whether to perform pre-translation query adaptation or not
Pre-Translation-Query-Adaptation-Enabled = true
##Specifies whether to perform post-translation query adaptation or
not
Post-Translation-Query-Adaptation-Enabled = false
##Generally specifies whether to perform result list adaptation or not
Result-List-Adaptation-Enabled = true
##Specifies whether to perform pre-merging result list adaptation or
not
Pre-Merging-Result-List-Adaptation-Enabled = false
##Specifies whether to perform post-merging result list adaptation or
not
Post-Merging-Result-List-Adaptation-Enabled = true
##Specifies whether to translate results or not (the Multilingual Mode
in general has to be enabled for this flag to be of use)
Result-List-Translation-Enabled = true
##Specifies whether to perform search logging or not
Search-Logging-Enabled = true
##Specifies whether to log IP addresses when performing search logging
IP-Address-Logging-Enabled = true
##Specifies whether to log session IDs when performing search logging
Session-ID-Logging-Enabled = true
##Specifies whether the UserModelingController should receive
notifications whenever a query is submitted or a result is clicked
UserModelingController-Should-Be-Notified-Of-Search-Events = true


#Class names for dynamic injection into the framework. Class names
have to be fully qualified names (i.e. packagename.classname).
Retrieval-Service-Class-Name = components.retrievers.BingSearchAccess
Language-Detection-Service-Class-Name = components.languagedetectors
                  .MicrosoftLanguageDetectionAccess
Translation-Service-Class-Name = components.translators
                  .MicrosoftTranslationAccess
Document-Localization-Service-Class-Name = components
            .documentlocalizers.MicrosoftDocumentLocalizationAccess
Personalized-Pre-Translation-Query-Adaptor-Class-Name = components.
                  queryadaptors.DefaultQueryAdaptor
Personalized-Post-Translation-Query-Adaptor-Class-Name = components.
                  queryadaptors.DefaultQueryAdaptor
Personalized-Pre-Merging-Result-List-Adaptor-Class-Name = components.
                  resultlistadaptors.DefaultResultListAdaptor
Personalized-Post-Merging-Result-List-Adaptor-Class-Name = components.
                  resultlistadaptors.DefaultResultListAdaptor
```

```
Personalized-Result-List-Merger-Class-Name = components
          resultlistmergers.RoundRobinRemoveDuplicatesResultListMerger
Non-Personalized-Pre-Translation-Query-Adaptor-Class-Name =
                   components.queryadaptors.DefaultQueryAdaptor
Non-Personalized-Post-Translation-Query-Adaptor-Class-Name =
                   components.queryadaptors.DefaultQueryAdaptor
Non-Personalized-Pre-Merging-Result-List-Adaptor-Class-Name =
                 components.resultlistadaptors.DefaultResultListAdaptor
Non-Personalized-Post-Merging-Result-List-Adaptor-Class-Name =
                 components.resultlistadaptors.DefaultResultListAdaptor
Non-Personalized-Result-List-Merger-Class-Name = components.
         resultlistmergers.RoundRobinRemoveDuplicatesResultListMerger
Search-Logger-Class-Name = components.
                   searchloggers.PostgreSQLSearchLogger
Authentication-Class-Name = components.
          authenticatorsandscrutinizers.PostgreSQLAuthenticationAccess
Demographic-Model-Access-Class-Name = dataaccess.
                   PostgreSQLDemographicModelAccess
Interests-Model-Access-Class-Name = dataaccess.
                   PostgreSQLInterestsModelAccess
Users-Access-Class-Name = dataaccess.PostgreSQLUsersAccess
Search-Logs-Analyzer-Class-Name = components.
                   searchlogsanalyzers.DefaultAnalyzer


#Multilinguality Settings

##Specifies whether the framework operates in multilingual mode or
monolingual mode
Multilingual-Mode-Enabled = true
##The default list of languages on which the framework should operate
Default-Operating-Languages = en,fr,de
##Default Interface Language
Default-Interface-Language = en
##Specifies, when running a personalized mode search when the
framework is operating in Multilingual mode, whether all the results
should be translated (usually to the user's preferred language), or
only results that the user is not familiar with (i.e. only translate
results that are not in the list of languages that the user is
familiar with)
### 1: RESULT_TRANSLATION_MODE_ALL
### 2: RESULT_TRANSLATION_MODE_ONLY_NON_FAMILIAR
Default-Result-Translation-Mode = 2
##Length of language acronyms, currently two-letter acronyms are used.
Language-Acronym-Length = 2


#Information Retrieval and Results Display Settings

##The default number of results that the retrieval component should
retrieve for each language (i.e. in each list)
Default-Required-Number-Of-Results = 10
##The maximum number of results that can be specified(programmatically
or by the user) for retrieval for each language (i.e. in each list)
Maximum-Required-Number-Of-Results = 50
## The default number of results to display per page
Default-Number-Of-Results-Per-Page = 10
##The default type of lists to retrieve
### 1: REQUIRED_MERGED_RESULT_LIST_ONLY
### 2: REQUIRED_SEPARATE_RESULT_LISTS_ONLY
### 3: REQUIRED_SEPARATE_AND_MERGED_RESULT_LISTS
```

```
Default-Lists-Type = 1
##The default sorting mode of multiple result lists
### 1: MULTIPLE_RESULT_LISTS_SORTING_MODE_NO_PRIORITY
### 2: MULTIPLE_RESULT_LISTS_SORTING_MODE_LANGUAGE_PRIORITY
Default-Multiple-Result-Lists-Sorting-Mode = 2
#Default Internal ResultList Parsing (e.g. json or xml)
Default-Internal-ResultList-Parsing = xml


#Query Adaptation Settings

##Number of terms to use for query expansion
Default-Number-Of-Terms-For-Query-Expansion = 2
##SimQ to SimD factor (a value between 0.0 and 1.0 that indicates the
importance of Query Similarity over Document Similarity).
SimQ-SimD-Factor = 0.5
##Similarity Threshold above which to attempt to expand query (a value
between 0.0 and 1.0).
Minimum-Similarity-Threshold-For-Query-Expansion = 0.2


#User Model Settings

##maximum number of interest vectors per language
Maximum-Interest-Vectors-Per-Language = 3
##maximum number of terms per interest vector
Maximum-Terms-Per-Interest-Vector = 20


#Experiment-specific configurations

##Path of the folder to output spread sheets (or csv files) of
quantitative evaluation
Evaluation-Results-Folder =
##A temporary flag to determine whether to run the framework in
experiment mode or not (This affects a number of things, including:
screens, menu, search logging, links, etc.)
Run-In-Experiment-Mode = true


#Proxy Settings
Proxy-Name =
Proxy-Port = 8080
Use-Proxy = true


#Database Settings (JDBC connection settings)
Database-Connection-String = jdbc:postgresql://127.0.0.1:5432/pmir
Database-Username =
Database-Password =


#API IDs/Keys (for accessing Web Services)

##Microsoft IDs/Keys
### Microsoft (Bing) Application ID
Microsoft-Bing-Application-ID =
###Microsoft Azure Marketplace Account Key
Microsoft-Azure-Bing-Search-API-Key =
```

```
##Google IDs/Keys
### Google API Key
Google-API-Key =

##Yahoo IDs/Keys
### Yahoo Application ID
Yahoo-Application-ID =


#URLs of Web Services (Search APIs, Transalation APIs, etc.)

##Microsoft Services
### Bing Search API Base URL
Bing-Search-Base-URL = https://api.datamarket.azure.com/Data.ashx
                  /Bing/Search/v1/Web?
### Microsoft Translator Base URL
Microsoft-Translation-Base-URL = http://api.microsofttranslator.com/V2
                  /Http.svc/Translate?
### Microsoft Translator for an Array of Strings Base URL
Microsoft-Translation-Multiple-Base-URL = http://api
                  .microsofttranslator.com/V2/Ajax.svc/TranslateArray?
### Microsoft Language Detection Base URL
Microsoft-Language-Detection-Base-URL = http://api.microsofttranslator
                  .com/V2/Http.svc/Detect?
### Microsoft (Bing) Web Page Translation Base URL
Microsoft-Web-Page-Translation-Base-URL = http://www
                  .microsofttranslator.com/bv.aspx?

##Google Services (requires update because some URLs have changed
recently)
### Google Search Base URL
Google-Web-Search-Base-URL = http://ajax.googleapis.com/
                  ajax/services/search/web?v=1.0
### Google University Research Program - Search Base URL
Google-University-Web-Search-Base-URL = https://research.google.com
                  /university/search/service?
### Google Translation Base URL
Google-Translation-Base-URL = http://ajax.googleapis.com/ajax
                  /services/language/translate?v=1.0
### Google Language Detection Base URL
Google-Language-Detection-Base-URL = http://ajax.googleapis.com
                  /ajax/services/language/detect?v=1.0
### Google Web Page Translation Base URL
Google-Web-Page-Translation-Base-URL = http://translate.google.com
                  /translate?

##Yahoo Services (requires update because some URLs have changed
recently)
### Yahoo Search Base URL
Yahoo-BOSS-Base-URL =  http://api.search.yahoo.com/WebSearchService
                  /V1/webSearch?
```