A Semi-Automated FAQ Retrieval System for HIV/AIDS

Edwin Thuma

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

School of Computing Science

College of Science and Engineering
University of Glasgow

April 15, 2015

Abstract

This thesis describes a semi-automated FAQ retrieval system that can be queried by users through short text messages on low-end mobile phones to provide answers on HIV/AIDS related queries. First we address the issue of result presentation on low-end mobile phones by proposing an iterative interaction retrieval strategy where the user engages with the FAQ retrieval system in the question answering process. At each iteration, the system returns only one question-answer pair to the user and the iterative process terminates after the user's information need has been satisfied. Since the proposed system is iterative, this thesis attempts to reduce the number of iterations (search length) between the users and the system so that users do not abandon the search process before their information need has been satisfied. Moreover, we conducted a user study to determine the number of iterations that users are willing to tolerate before abandoning the iterative search process. We subsequently used the bad abandonment statistics from this study to develop an evaluation measure for estimating the probability that any random user will be satisfied when using our FAQ retrieval system.

In addition, we used a query log and its click-through data to address three main FAQ document collection deficiency problems in order to improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system. Conclusions are derived concerning whether we can reduce the rate at which users abandon their search before their information need has been satisfied by using information from previous searches to: Address the term mismatch problem between the users' SMS queries and the relevant FAQ documents in the collection; to selectively rank the FAQ document according to how often they have been previously identified as relevant by users for a particular query term; and to identify those queries that do not have a relevant FAQ document in the collection.

In particular, we proposed a novel template-based approach that uses queries from a query log for which the true relevant FAQ documents are known to enrich the FAQ documents with additional terms in order to alleviate the term mismatch problem. These terms are added as a separate field in a field-based model using two different proposed enrichment strategies,

namely the Term Frequency and the Term Occurrence strategies. This thesis thoroughly investigates the effectiveness of the aforementioned FAQ document enrichment strategies using three different field-based models. Our findings suggest that we can improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries in our query log. Moreover, our investigation suggests that it is important to use an FAQ document enrichment strategy that takes into consideration the number of times a term occurs in the query when enriching the FAQ documents. We subsequently show that our proposed enrichment approach for alleviating the term mismatch problem generalise well on other datasets.

Through the evaluation of our proposed approach for selectively ranking the FAQ documents, we show that we can improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system by incorporating the click popularity score of a query term t on an FAQ document d into the scoring and ranking process. Our results generalised well on a new dataset. However, when we deploy the click popularity score of a query term t on an FAQ document d on an enriched FAQ document collection, we saw a decrease in the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system.

Furthermore, we used our query log to build a binary classifier for detecting those queries that do not have a relevant FAQ document in the collection (Missing Content Queries (MCQs))). Before building such a classifier, we empirically evaluated several feature sets in order to determine the best combination of features for building a model that yields the best classification accuracy in identifying the MCQs and the non-MCQs. Using a different dataset, we show that we can improve the overall retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system by deploying a MCQs detection subsystem in our FAQ retrieval system to filter out the MCQs.

Finally, this thesis demonstrates that correcting spelling errors can help improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system. We tested our FAQ retrieval system with two different testing sets, one containing the original SMS queries and the other containing the SMS queries which were manually corrected for spelling errors. Our results show a significant improvement in the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system.

Acknowledgements

The completion of this thesis would not have been possible without the support of my family, friends and colleagues. First and foremost, I would like to thank my wife Kagiso Thuma whose love, support and encouragement made it possible for me to complete this work.

I am indebted to my supervisors Simon Rogers and Iadh Ounis for patiently steering me and guiding me throughout the course of this work. I am grateful for their constructive comments they provided on various drafts of this thesis.

Also, I would like to thank Dimane Mpoeleng for commenting on various drafts of this thesis and for the help and support he provided me while conducting various user studies in Botswana.

Lastly, I would like to thank the trustees of the Eleanor Emery Scholarship for supporting this research.

I dedicate this thesis to:

- My son Loago Thuma, who has grown into a wonderful 5 year old in spite of his father spending so much time away from him working on this thesis.
- My mother Gladys Thuma, for instilling the importance of hard work in me and encouraging me to reach my dreams.

Table of Contents

List of Tables

List of Figures

Nomenclature

Acronyms

ARV	Antiretroviral
AQE	Automatic Query Expansion
AvICTF	Average Inverse Collection Term Frequency
AvSCQ	Average Collection Query Similarity
BM25	Okapi Best Match ranking function
CLEF	Conference and Labs of the Evaluation Forum
DFR	Divergence From Randomness
docid	Document Identifier
DPH	Hyper-Geometric DFR Model using Popper's Normalization
DLH	Hyper-Geometric DFR Model using the Laplace Normalization
FAQ	Frequently Asked Question
FIRE	Forum for Information Retrieval Evaluation
FRACT	FAQ Retrieval and Clustering Technique
GDP	Gross Domestic Product
Hiemstra_lm	Hiemstra Language Model
HIV/AIDS	Human Immunodeficiency Virus/Acquired Immunodeficiency Syndrome
HMM	Hidden Markov Model
IDF	Inverse Document Frequency
IHISM	Integrated Healthcare Information System through Mobile Telephony
IR	Information Retrieval
JM	Jelinek-Mercer
KL	Kullback-Leibler
MAP	Mean Average Precision
MaxSCQ	Maximum Collection Query Similarity
MCQs	Missing Content Queries

MDE	Minimal Differentiator Expression
MoH	Ministry of Health
MRD	Machine Readable Dictionary
MRR	Mean Reciprocal Rank
NLP	Natural Language Processing
NTCIR	NII Testbeds and Community for Information Access Research
PL2	Poisson Model with Laplace After-Effect and Normalisation 2
qid	Query Identifier
RSJ	Robertson/Sparck Jones
SCS	Simplified Clarity Score
SERP	Search Engine Results Pages
SMS	Short Message Service
SMT	Statistical Machine Translation
SumSCQ	Summed Collection Query Similarity
TF-IDF	Term Frequency-Inverse Document Frequency
TREC	Text REtrieval Conference
UDH	User Data Header
C-SVC	C-Support Vector Classification
QA	Question Answering
Q&A	Question and its Associated Answer

Symbols

Symbol	Description
Q	Is a user query
t	Is a query term
tf	Is the frequency of the query term in the document d
tfn	Represents the normalised within document term frequency
d	Is a document in the collection
C	Is the document collection
l	Denotes the length of the document d
avg_l	Is the average document length in the collection
N	Is the number of documents in the document collection
r	Denotes the rank at which the system would rank the relevant FAQ document
qtf	Is the number of occurrences of a given term in the query Q
$qtfn$	Represents the normalised query term frequency is given by
qtf_{max}	Is the maximum query term frequency among the query terms

dft	Is the number of documents in the collection that have a term t
b	Is the term frequency normalisation hyper-parameter
$w^{(1)}$	Denotes the IDF factor
w_f	Is the weight for the field f
tf_f	Is the frequency of the query term in the f^{th}
tfn_f	Represents the normalised term frequency of the query term in the f^{th} field
l_f	Represents the number of tokens in the f^{th} field
avg_l_f	Represents the average length of the f^{th} in the collection
b_f	Denotes the term frequency normalisation hyper-parameter of the f^{th} field
tfc	Is the frequency of the term t in the collection C
$token_c$	Represents the number of tokens in the collection C
ql	Is length of the query Q
k	top k retrieved FAQ documents

Chapter 1

Introduction

1.1 Motivation

This thesis describes a semi-automated Frequently Asked Question (FAQ) answering system that can be queried by users using short text messages to provide answers on Human Immunodeficiency Virus / Acquired Immunodeficiency Syndrome (HIV/AIDS) related queries. This study builds upon a project that I worked on with other members of the Department of Computer Science[1] at the University of Botswana[2] called Integrated Healthcare Information System through Mobile Telephony (IHISM) (Anderson et al., 2007a,b). A similar study was also conducted by Adesina and Nyongesa (2013) at the University of Western Cape in South Africa. The objective of the IHISM project is to develop a system that will provide access to a variety of HIV/AIDS related information for different users. Some of the proposed services to be offered by the IHISM system are the following (Anderson et al., 2007a,b):

- A semi-automated FAQ answering service for HIV and AIDS – This Short Message Service (SMS) will provide answers on HIV/AIDS related questions when queried by users.
- An automated reminder service – This service will automatically send reminders to patients. For example, the system will be able to send reminders about when to take HIV/AIDS drugs.
- A personal information service – This service will enable users to be able to query the IHISM portal in order to retrieve information about themselves. For example, users will be able to retrieve information about their next appointment to the doctor.

[1] http://www.cs.ub.bw/moodle/
[2] http://www.ub.bw/

According to the third Botswana AIDS Impact Survey (Kandala et al., 2012), the national HIV prevalence rate in 2012 was 17.6%. The rates of infection amongst females and males were reported to be 20.4% and 14.3%, respectively. This pandemic has a negative impact on the national development and socio-economic transformation of Botswana (Bollinger and Stover, 1999, Dixon et al., 2002). In particular, previous estimates have suggested that the average rate of growth of the Gross Domestic Product (GDP) for Botswana in 2000 – 2010 will be reduced by 1.5 % (Dixon et al., 2002). The government of Botswana has developed the second National Strategic Framework for HIV and AIDS (NSF II)[3] in order to outline national priorities for the response to HIV and AIDS for a seven year period from 2010 – 2016 (Molomo, 2009). The objective of this NFS II for HIV and AIDS is to:

- Devise strategies that can prevent the spread of HIV infection and reduce the socio-economic impact of this disease (e.g behavioural change interventions).
- Sustain high quality, cost effective HIV/AIDS services by strengthening human resources, improving infrastructure and procuring medical supplies and equipments.
- Develop a strategic plan for strengthening the information management systems in order to enhance data sharing and data utilisation in policy formulation and review.
- Provide treatment, care and support to those infected and affected by HIV/AIDS.

To support the initiatives by the Government, the Department of Computer Science at the University of Botswana has pledged to develop a semi-automated FAQ retrieval system that can be queried through SMS by both literate and semi-literate users to provide answers on HIV/AIDS related queries. Semi-literate users are those who have basic literacy but cannot read and write fluently (Findlater et al., 2009). On the other hand, literate users are those that can read and write fluently. The Department of Computer Science decided on mobile phone technology because of its low cost and high penetration in the market (Bornman, 2012, Donner, 2008).

One of the main advantages of using this technology is that it extends to rural settlements and maximizes coverage and access of information to a wide majority of the country's population. It is for this reason that mobile phones have emerged as the platform of choice for providing services such as banking (Medhi et al., 2009), payment of utility bills (Zhang and Dodgson, 2007) and learning (M – Learning) (Bornman, 2012) in the developing world. Currently, a majority of users access these services using low-end mobile phones (Figure 1.1). Low-end mobile phones provide limited capabilities such as voice calling, text messaging, multimedia services and internet capabilities. Less than 20% of the population in

[3]http://www.ilo.org/aids/legislation/WCMS_172465/lang–en/index.htm

Sub-Saharan Africa access these services using high-end mobile phones (smartphones) (Deloitte, 2012, GSMA-Intelligence, 2013). High-end mobile phones (Figure 1.2) have more capabilities and include all the features in low-end mobile phones plus additional features such as touch screen, web-browsing and WI-FI. The next section describes the HIV/AIDS FAQ question-answer booklet that the proposed system will use as an information source to provide answers on HIV/AIDS related queries.

Figure 1.1: Low-end Mobile Phone (Nokia 6020)

Figure 1.2: High-end Mobile Phone (Nokia Asha 501)

1.2 Information Source for the Semi-Automated FAQ Retrieval System

There is an HIV/AIDS FAQ question-answer booklet provided by the Ministry of Health (MoH)[4] in Botswana. This HIV/AIDS FAQ question-answer booklet contains the most frequently asked questions about HIV/AIDS and Antiretroviral (ARV) therapy. The MoH in Botswana has set up a call centre called IPOLETSE[5] so that people can call and ask any questions related to HIV/AIDS. At this call centre, operators use the aforementioned HIV/AIDS FAQ question-answer booklet to find and reply to HIV/AIDS related questions posed by the information seekers. For a larger population, it is possible that some users may have to wait longer on the phone line while the call centre operators are responding to other user queries. Our main objective is to automate this process for users to ensure that they receive information about HIV/AIDS in timely manner. We provide the table of contents for the HIV/AIDS FAQ question-answer booklet in Table 1.1.

Each chapter in the HIV/AIDS FAQ question-answer booklet is made up of entries of question-answer pairs. Table 1.2 provides examples of question-answer pairs that can be found in various chapters of the IPOLETSE HIV/AIDS FAQ question-answer booklet.

[4]http://www.moh.gov.bw/
[5]http://www.hiv.gov.bw/content/ipoletse

Table 1.1: The chapters and the number of question-answer pairs in each chapter in the IPOLETSE HIV/AIDS question-answer booklet.

Chapters	**Number of Question/Chapter**
Understanding HIV and AIDS	23
Protecting Yourself (Condoms)	28
Understanding Tuberculosis (TB)	11
Taking the test	12
Routine HIV testing	9
I have found out that I have HIV - what should I do?	13
Nutrition, Vitamins and HIV/AIDS	8
Introduction to ARV Therapy	50
The Government's ARV Therapy Programme - Masa	18
Women and Children and HIV/AIDS	28
Men and HIV/AIDS	5
Total Question and Answer Pairs	**205**

Table 1.2: Examples of question-answer pairs found in the IPOLETSE HIV/AIDS question-answer booklet.

QUESTION	**ANSWER**
Does HIV / AIDS affect women differently from men?	No, the virus affects both men and women in exactly the same way i.e. by making the immune system weak, so that it cannot fight off other illnesses.
How does HIV weaken the immune system?	The immune system is made up of "soldiers", which fight off diseases. These "soldiers" are called CD4 cells, which are white blood cells. HIV attacks and kills the CD4 cells in your body.
Is it true that a man can remain negative even if he sleeps with an HIV – positive woman because men have stronger blood?	Men do not have stronger blood than women. If you sleep with someone who is HIV – positive without using a condom, you will contract HIV.
What is IPT and how does it work?	IPT stands for Isoniazid Preventive Therapy. It is used to protect people living with HIV/AIDS from contracting TB. It does this by killing the silent TB infection before it makes the person ill. IPT is available in government clinics, free of charge, and the course takes six months to complete.

The IPOLETSE HIV/AIDS FAQ question-answer booklet will be used to build the semi-automated FAQ retrieval system for HIV/AIDS in this thesis. For any query posed by the information seekers, the proposed system will respond automatically with the correct question-answer pair from the aforementioned question-answer booklet. Like many search engines (Silvestri, 2010, Zhang and Nasraoui, 2006), our semi-automated FAQ retrieval system will record previous searches of the information seekers. This information will be used to improve the retrieval performance of our FAQ retrieval system in the future.

For the remainder of this thesis, the complete set of 205 question-answer pairs in the HIV/AIDS FAQ question-answer booklet will be referred to as the FAQ document collection. Each question-answer pair in the HIV/AIDS FAQ question-answer booklet will be referred to as an FAQ document. Information seekers will be referred to as users and the users' SMS messages (questions) will be referred to as queries.

1.3 Aspects to Consider when Developing a Semi-Automated FAQ Retrieval System

In this section, the main aspects to consider when developing a semi-automated FAQ retrieval system are discussed. Four main aspects are identified and briefly discussed. In particular, some of these aspects originate from the Forum for Information Retrieval Evaluation (FIRE)[6] SMS-Based FAQ retrieval tasks. For example, handling noisy text, which is inherent in SMS queries and the FAQ document collection deficiency problems (no relevant information and term mismatch problem). In addition, the other aspects to consider, which are less documented in the literature are: search result presentation, which is more specific to SMS-Based FAQ systems and handling cross-lingual and bilingual queries.

1.3.1 Handling Noisy Text

Noisy text refers to any kind of text that has errors. Examples include: misspellings, non-standard abbreviations, transliterations, phonetic substitutions and omissions (Kothari et al., 2009). Noisy text is very common in SMS messages because of the size of the keypad used for entering text and the illiteracy of users. Correcting these errors in a semi-automated FAQ retrieval system that rely on keyword matching in their weighting models is very important for effective ranking and retrieval of the relevant FAQ documents (Kothari et al., 2009).

1.3.2 FAQ Document Collection Deficiencies

Another aspect to consider when developing a semi-automated FAQ retrieval system is that the information supplier has to create the FAQ document collection by answering typical questions that users may have. According to Sneiders (1999), FAQ document collections consist of ordinary FAQ documents which often have the following deficiencies:

- The information supplier does not know the users' actual questions. Rather, the information supplier constructs question candidates in advance using their own knowl-

[6] http://www.isical.ac.in/fire/2011/index.html

edge. They answer the question candidates, however, these question candidates will not always satisfy the users' information needs. The HIV/AIDS FAQ question-answer booklet used in this thesis as an information source to provide answers on HIV/AIDS related queries was created in a similar manner.

- Each FAQ document consists of a small number of words, unlike ordinary documents. The information supplier chooses words in each FAQ document according to their knowledge. However, the FAQ documents and the user queries may still use different words or phrases to refer to the same thing.

These deficiencies may result in the lexical disagreement (term mismatch) problem when keyword matching systems are deployed to match user queries to the relevant FAQ document in the collection (Kim and Seo, 2006, Sneiders, 2009).

For example, the user's query: *"Is HIV/AIDS gender based to some extent?"* and the FAQ document: *"Does HIV/AIDS affect women differently from men? No, the virus affects both men and women in exactly the same way i.e. by making the immune system weak, so that it cannot fight off other illnesses"* are semantically similar but lexically different. This term mismatch between the user's query and the relevant FAQ document may result in a less effective ranking by a retrieval system that relies on keywords matching in its weighting model (Fang, 2008). Such a system may return FAQ documents that are not related to the user's query (Kim and Seo, 2006). Giving non-relevant information about diseases such as HIV/AIDS to users can have serious consequences as they might abandon their search before their information need has been satisfied. It is possible that some users may query the FAQ retrieval system while in a distressed state. For example, a user might want to know whether he/she has contracted the HIV/AIDS virus of not. Therefore, it is crucial that we return the relevant FAQ documents to users before they abandon their search as this could further increase their stress levels.

1.3.3 Presentation of Results

Another aspect to consider when developing a semi-automated FAQ retrieval system that can be queried through low-end mobile phones is the presentation of results. In low-end mobile phones, the maximum number of characters per SMS is 160. If an SMS exceeds that limit, it is split into multiple SMS messages that are delivered to the recipients' mobile phones as separate messages. This limitation can pose a challenge to users if for each SMS query a ranked list of FAQ documents are returned. For example, we can see in Figure 1.3 that a majority of the FAQ documents need between 2 and 4 SMS messages to be sent to the user. Therefore, if the proposed FAQ retrieval system for HIV/AIDS returns 5 ranked FAQ documents to the user for each SMS query, the user will receive approximately 15 SMS

messages, which may be addressing different information needs. Some users, especially the semi-literate, may find it difficult to navigate through this long list of SMS messages to find the relevant FAQ document. The lack of clear boundaries can have serious consequences as users may believe that some of the non-relevant FAQ documents returned by the system are relevant to their queries.

It is worth pointing out that there has been recent developments in high-end devices (e.g smartphones) that support message concatenation[7]. For these mobile phones, a longer SMS from the sender's mobile phone is initially split into separate messages of 153 characters each. The other bits are reserved for the User Data Header (UDH) to be inserted at the beginning of each SMS message. Each SMS message is then transmitted and billed separately. Before being delivered to the recipient, the message is concatenated and then delivered as a long SMS message. The user will then be able to navigate easily through the 5 long retrieved FAQ documents without necessarily having to determine the beginning and the end of each FAQ document. Also, unlike low-end devices, high-end devices have a larger screen display and support web browsing. For this kind of devices, it will be ideal to develop a web-based application that can provide a ranked list of FAQ documents for each query submitted by the user. However developing a web-based application will be impractical since a majority of people in Botswana own low-end devices. For example, a recent survey has shown that desktop computers are the dominant platform for internet browsing in Botswana, with 91.7% and 8.3% of web traffic being desktop based and mobile based respectively (Deloitte, 2012). A similar trend was observed in neighbouring states. It was suggested that 87.4% and 12.6% of web traffic in South African in the year 2012 originated from desktops and mobile phones respectively.

1.3.4 Handling Cross-Lingual and Bi-lingual Queries

In a multilingual nation like Botswana, people often mix different languages together in informal modes of communication such as chat rooms, Twitter, SMS and email (Otlogetswe, 2008). This poses a challenge in developing an FAQ retrieval system for multilingual speakers when documents to be queried by users are expressed in only one language, for example in English. An FAQ retrieval system developed for these users must be able to accurately determine the language used in each of the words in the SMS query so that any word that is not expressed in the language used in the FAQ document collection can be accurately translated to the correct word in the language of the FAQ document collection. This phenomenon is known as cross language information retrieval (Oard and Dorr, 1996), where queries are expressed in one language and the FAQ documents expressed in the other language. During

[7] http://en.wikipedia.org/wiki/Concatenated_SMS

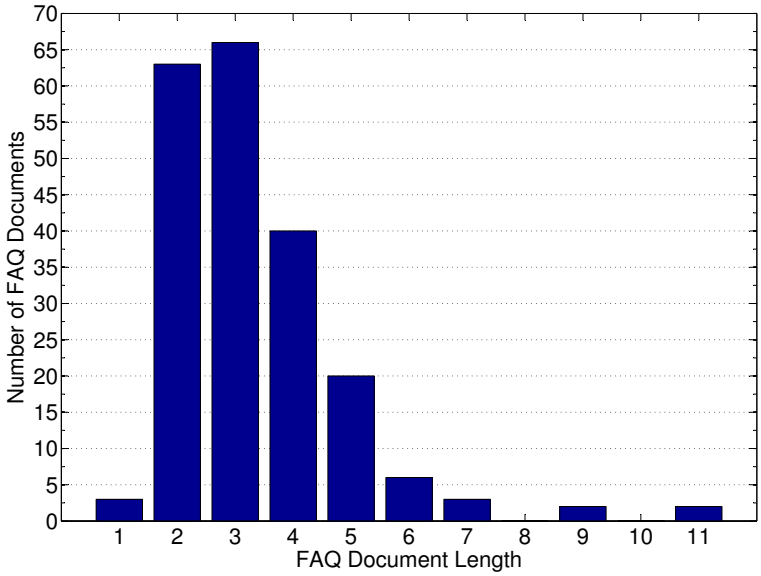


Figure 1.3: A histogram showing the number of FAQ documents in the HIV/AIDS corpus occurring given a certain document length. The length is measured by the number of SMS messages needed to send the each FAQ document.

retrieval either the query has to be expressed in the same language as the FAQ documents or the FAQ documents has to be expressed in the same language as the user's query.

1.3.5 Summary

This section discussed the four main aspects to consider when developing a semi-automated FAQ retrieval system. Some of the aspects discussed here have been studied in the literature such as handling noisy text as we will see in Chapter 2. This thesis will focus on the following two aspects: the FAQ document collection deficiency and the result presentation problem outlined in Section 1.3.2 and 1.3.3 respectively. Several approaches exist in the literature that attempt to address the challenges posed by the FAQ document collection deficiency problem. On the other hand, there is no work in the literature that addresses the issue of result presentation in an SMS-Based FAQ retrieval setting. This thesis will attempt to address this by proposing an iterative interaction retrieval approach where the user engages with the system in the question answering process. Details of this iterative interaction retrieval strategy are provided in Chapter 4. In this thesis, we will assume that the language used in the user's query and the FAQ document collection is English. The thesis will also assume that the user's SMS queries have been corrected for spelling errors.

1.4 Thesis Statement

In this thesis, we hypothesise that by using information from previous searches, we can improve the probability that any random user will be satisfied when using our FAQ retrieval system through the following techniques:

- Enriching the FAQ documents with additional terms from a query log in order to reduce the term mismatch problem between the users' queries and the relevant FAQ documents.
- Ranking the FAQ documents according to how often they have been previously identified as relevant by users for a particular query term.
- Detecting whether there is an answer for any user query.

1.5 Contributions

The main contributions of this thesis are the following:

- First, we develop a test collection using a query log collected from potential users of our FAQ retrieval system in Botswana. We later use this test collection in subsequent chapters to validate our thesis statement.
- We introduce our iterative retrieval strategy, which is designed to allow users of low-end mobile phones to be able to search a semi-automated FAQ retrieval system through SMS messages. Since the proposed retrieval strategy is iterative, this thesis investigates the number of iterations users are willing to engage with such a system. In addition, the thesis proposes a new evaluation measure that uses information from abandoned queries to estimate the probability that any random user will be satisfied when using the system. Furthermore, we conduct an empirical evaluation to determine the most appropriate way of representing the FAQ documents in our information source. In order to achieve the above goals, we identified the following research questions:
  - *Chapter 4-Research Question One (C4-RQ1)*: What is the maximum number of iterations that users are willing to tolerate before abandoning the iterative search process?
  - *Chapter 4-Research Question Two (C4-RQ2)*: Does the search length of previous searches influence the search length of subsequent searches?
  - *Chapter 4-Research Question Three (C4-RQ3)*: Does indexing the question part only improve the overall retrieval performance?

  - *Chapter 4-Research Question Four (C4-RQ4)*: Does removing stopwords improve the overall retrieval performance?

- In this thesis, several research questions concerning how we can use information from previous searches to improve the retrieval effectiveness of our semi-automated FAQ retrieval system are addressed. First we propose a new structure for our FAQ documents. Each FAQ document is divided into three fields, *QUESTION*, *ANSWER*, and *FAQLog* field. We then investigate whether we can resolve the term mismatch problem between the user query and the relevant FAQ documents in the collection by adding terms from a query log into the *FAQLog* field and deploying a field-based term weighting model for retrieval. Terms are added into this field according to two different enrichment strategies, which are Term Occurrence and the Term Frequency enrichment strategies. We thoroughly evaluate the different enrichment strategies with three different field-based term weighting models. Furthermore, we investigate whether increasing the size of the query log that we use to enrich the FAQ documents can improve the retrieval performance. The following research questions were identified to help us with our investigation:

  - *Chapter 5-Research Question One (C5-RQ1)*: Can we improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries for which the true relevant FAQ document are known.
  - *Chapter 5-Research Question Two (C5-RQ2)*: Can we improve the overall recall and the probability that any random user will be satisfied by taking into consideration the number of times a term occurs in the queries when enriching the FAQ documents.
  - *Chapter 5-Research Question Three (C5-RQ3)*: Does increasing the number of queries used in enriching the FAQ documents increase the overall recall and the probability that any random user will be satisfied.
  - *Chapter 5-Research Question Four (C5-RQ4)*: Does the proposed enrichment strategies produce similar results when deployed with different field-based term weighting models.

- Another aspect we investigate in this thesis is whether we can improve the retrieval effectiveness of our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously marked relevant by the user for a particular query term (click popularity score). In our investigation, we propose methods for combining this click popularity score with the BM25 term weighting model. In addition, we evaluate our proposed approach on an enriched FAQ document collection using a

field-based model. The following research questions were identified to help us with our investigation:

  - *Chapter 6-Research Question One (C6-RQ1)*: Can we improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users for a particular query term t.
  - *Chapter 6-Research Question Two (C6-RQ2)*: Can we improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users without taking into consideration the query terms associated with those FAQ documents.

- Moreover, since the collection being searched is very small, it is possible that some user queries might not have the relevant FAQ documents in the collection. In this thesis, we use our query log to empirically evaluate the different feature sets that we can use to build a classifier for detecting such queries. Our aim is to ensure that users do not iterate longer with the system as this might cost them more money and they might also loose interest in using the system because of previous failure in satisfying their information need. The following research questions were identified for our empirical evaluation:

  - *Chapter 7-Research Question One (C7-RQ1)*: Which set of features produce the best classification performance when classifying *MCQs* and *non-MCQs*?
  - *Chapter 7-Research Question Two (C7-RQ2)*: Does combining different feature sets produce a better classification performance compared to any individual feature set?
  - *Chapter 7-Research Question Three (C7-RQ3)*: Does increasing the size of the training set for the *MCQs* and the *non-MCQs* yield a better classification performance?

- Finally, in the closing chapter of this thesis, we carry out an empirical investigation to determine whether we can reduce the rate at which users abandon their search before their information need has been satisfied by combining all the proposed approaches. We also investigate whether our proposed approaches generalise well on a second dataset, including when we combine the different approaches/subsystems. In particular, we identified the following research questions to help us with our investigation:

  - *Chapter 8-Research Question One (C8-RQ1)*: Do the previous results generalise on a second dataset, including when we combine our subsystems.

- *Chapter 8-Research Question Two (C8-RQ2)*: What impact does the missing content queries detection system have on the probability that any random user will be satisfied when deployed in our FAQ retrieval system?
- *Chapter 8-Research Question Three (C8-RQ3)*: Does correcting spelling errors help improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system?

1.6 Origins of the Material

Some of the material that forms part of this thesis have been peer-reviewed and presented at various conferences. In particular:

- Chapter 4 - The experiments on how we measure the desired search length and how we use bad abandonment statistics to come up with an evaluation measure of satisfaction were published in: Edwin Thuma, Simon Rogers, and Iadh Ounis , "Evaluating Bad Query Abandonment in an Iterative SMS-based FAQ Retrieval System", In Proceedings of the 10th Conference on Open Research Areas in Information Retrieval (OAIR13), Lisbon, Portugal, 2013. ACM.
- Chapter 5 - The experiments on the use a query log and field-based models to resolve term mismatch problem were published in: Edwin Thuma, Simon Rogers, and Iadh Ounis, "Exploiting Query Logs and Field-Based Models to Address Term Mismatch in an HIV/AIDS FAQ Retrieval System", Natural Language Processing and Information Systems, Lecture Notes in Computer Science Volume 7934, 2013, pp 77-89.
- Chapter 7 - The experiments on our evaluation of the different feature set for classifying the missing content queries were published in: Edwin Thuma, Simon Rogers, and Iadh Ounis, "Detecting Missing Content Queries in an SMS-Based HIV/AIDS FAQ Retrieval System", Advances in Information Retrieval, Lecture Notes in Computer Science Volume 8416, 2014, pp 247-259

1.7 Thesis Outline

The rest of this thesis is organised as follows:

- Chapter 2 provides a literature review on retrieval strategies that are normally deployed in FAQ retrieval systems. We begin by describing the difference between FAQ retrieval systems and Question Answering systems. Furthermore, we categorise FAQ retrieval

systems into Desktop-Based and SMS-Based FAQ retrieval systems. We then provide an overview of the different approaches that are common to both Desktop-Based and SMS-Based FAQ retrieval systems. In particular, how different systems resolve the term mismatch problem between the user query and the relevant FAQ documents. This is followed by an overview of the approaches that are only deployed in SMS-Based FAQ retrieval systems. In particular, we focus on the different SMS normalisation techniques and on the different techniques for detecting missing content queries.

- Chapter 3 describes how we will evaluate the different retrieval strategies that we deploy in our FAQ retrieval system. In particular, we describe the Cranfield evaluation methodology and several Information Retrieval (IR) evaluation measures that we use throughout the thesis. We then describe how we created the test collection that we use through out this thesis.

- In Chapter 4, we describe in detail the main building blocks of the semi-automated FAQ retrieval system. We investigate in detail the number of iterations users are willing to tolerate before abandoning the iterative search process. We also investigate whether the search length of previous searches has an effect on the search length of subsequent searches. We then show how we can use information from abandoned queries to estimate the probability that any random user will be satisfied when using our system. In addition, we carry out an empirical evaluation to determine the term weighting model to use in our baseline system.

- Chapter 5 describes how we use a query log and field-based models to resolve the term mismatch problem between the user query and the relevant FAQ documents. In particular, we experiment with two different strategies for adding additional terms from a query log into a structured FAQ document that uses fields. We then deploy three different field-based models to evaluate or proposed approach.

- Chapter 6 describes how we rank the FAQ documents according to how often they have been previously marked relevant by users. In particular, we propose a novel way of incorporating the click popularity score of a query term t on an FAQ document d into the scoring and ranking process with the BM25 term weighting model. We compare our approach with a language model approach for information, which uses a click-based FAQ document prior to rank the FAQ documents.

- In Chapter 7, we carry out an empirical evaluation to determine the best combination of features for building a model that yields the best classification accuracy in identifying queries that do not have the relevant FAQ documents in the collection. We experiment with three different features sets. Using three different classifiers, we experimentally

examined the classification accuracy of the individual feature sets and the combined features sets.

- In Chapter 8, we describe how we test the generality of our previous approaches on a second dataset, including when we combine the different subsystems. In particular, we test whether incorporating the click popularity score of a query term t on an FAQ document d into the scoring process on an enriched FAQ document collection improves the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system. We also investigate whether deploying a binary classifier for detecting those queries that do not have the relevant FAQ document in the collection in our FAQ retrieval system can improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system. Finally, we investigate whether correcting spelling errors can help improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system.

- Chapter 9 closes the thesis by summarising the contributions and conclusions of the individual chapters. In addition, this chapter provides directions for future work related to, or stemming from this thesis.

Chapter 2

Related Work

2.1 Introduction

FAQ retrieval, Question Answering (QA) (retrieval) and Question and its Associated Answer (Q&A) retrieval are novel IR tasks (Jeon, 2007, Oleksandr and Marie-Francine, 2011). They are IR tasks because they deal with the representation, storage, organisation of, and access to information items (Baeza-Yates and Ribeiro-Neto, 1999). However, unlike traditional IR systems, FAQ retrieval, QA (retrieval) and Q&A retrieval directly return several possible answers instead of a list of potentially relevant documents for each user query (Jeon et al., 2005, Xue et al., 2008, Oleksandr and Marie-Francine, 2011).

FAQ retrieval is similar to Q&A retrieval but different from QA (retrieval) (Jeon, 2007). For example, FAQ retrieval systems and Q&A retrieval systems use pre-stored sets of question-answer pairs (FAQ documents) as an information source (FAQ document collection) to answer natural language questions posed by the users. However, the collection of documents used in FAQ retrieval systems is usually small (less than a few hundred) and specific to a domain (Jeon, 2007). This document collection is usually good in quality as it is created and maintained by domain experts (Jeon, 2007). Q&A retrieval systems on the other hand tend to use a large collection of documents (more than a million), which are much broader in coverage, spanning several domains (Jeon, 2007). This collection is usually created by experts and non-experts resulting in poor quality. Some of the publicly available Q&A retrieval systems are Baidu Zhidao[8], Yahoo! Answers[9] and Live QnA[10].

QA (retrieval) systems on the other hand return answers themselves, rather than documents containing answers, in response to a natural language question (Dang et al., 2006, Hirschman

[8]http://zhidao.baidu.com
[9]http://answers.yahoo.com
[10]http://qna.live.com

and Gaizauskas, 2001, Metzler and Croft, 2005). These answers are generally a short fragment of the text extracted from some of the documents in the collection (Metzler and Croft, 2005). Two types of QA (retrieval) systems have been proposed in the literature : open domain and restricted domain QA systems (Oleksandr and Marie-Francine, 2011). In restricted domain question answering, systems deal with questions under a specific domain and they extract answers for natural language questions from specific information sources that have either been developed for question answering or that have been developed for other purposes (Oleksandr and Marie-Francine, 2011, Mollá and Vicedo, 2007). They are often developed for a category of users who know and use domain specific terminology in their query formulation (for example, in the medical domain) (Athenikos and Han, 2010, Mollá and Vicedo, 2007). However, open domain systems deal with questions that are independent of the domain and they extract answers for natural questions from large text databases such as the web (Prager, 2006). Table 2.1 summarises the difference between FAQ retrieval, QA (retrieval) and Q&A retrieval Systems.

Table 2.1: The difference between FAQ retrieval, QA (retrieval) and Q&A retieval (Jeon, 2007).

	FAQ retrieval	**QA (retrieval)**	**Q&A retrieval**
Query Type	question	question	question
Collection	FAQs	documents	FAQs
Collection Size	small (< a few hundred)	large (> a million)	large (> a million)
Collection Quality	good (all answers are correct)	poor (some incorrect answers)	poor (some incorrect answers)
Output	question and answer	answer	question and answer

Current work in FAQ retrieval can be divided into two categories: Desktop-Based FAQ retrieval and SMS-Based FAQ retrieval. In Desktop-Based FAQ retrieval, since the screen displays are large, a large number of FAQ documents can be retrieved for each user query. This allows users to be able to see relevant FAQ documents that are ranked lower than the non relevant FAQ documents. On the other hand, in SMS-Based FAQ retrieval, because of a limited number of characters that can be sent per SMS and a limited screen display, only a few results can be returned for each user query (Adesina et al., 2014). Such systems require a high precision retrieval strategy in order to reduce the rate at which users abandon the search process before their information need has been satisfied. The other difference is in the nature of the query submitted to the retrieval system. SMS-Based queries normally contain a lot of spelling errors (Aw et al., 2006, Kothari et al., 2009). In this chapter, we review the different approaches that have been proposed in the literature for both Desktop-Based FAQ retrieval and SMS-Based FAQ retrieval. The remainder of this chapter is organised as follows:

- In Section 2.2, we review different approaches for developing Desktop-Based FAQ retrieval systems proposed in the literature.

- In Section 2.3, we review different approaches for developing SMS-Based FAQ retrieval systems proposed in the literature.
- In Section 2.4, we outline the gaps in the literature and describe how the work carried out in this thesis fits into the current research context.

2.2 Review of Desktop-Based FAQ Retrieval Systems

Several approaches for developing Desktop-Based FAQ retrieval systems are reviewed in this section. These are characterised as: Statistical approaches, template-based approaches, Natural Language Processing (NLP) and ontology-based approaches (Sneiders, 2009). Statistical-based approaches are more robust and efficient and are widely deployed in large collections (Romero et al., 2013). Template-based approaches require a cluster of templates that mimic the expected user question for each FAQ (Sneiders, 2009). Natural language processing and ontology-based approaches require hand crafted domain dependant resources (linguistic rules, lexicons and domain ontologies) to capture the semantics of terms (Romero et al., 2013). The concept of an ontology refers to knowledge representation for domain specific content (Yang et al., 2007). Since FAQ retrieval is a subclass of Question Answering (QA), this review will also examine related work in Question Answering (QA) and Question and its associated answer (Q&A) retrieval. The remainder of this section is organised as follows: Section 2.2.1 provides a review on Natural Language Processing and Ontology-Based Approaches; Section 2.2.2 provides a review on Template-Based approaches followed by a review of Statistical approaches in Section 2.2.3.

2.2.1 Natural Language Processing and Ontology-Based Approaches

Natural language processing and ontology-based approaches require knowledge bases such as domain ontologies, lexical resources and linguistic rules to capture the semantics of terms (Romero et al., 2013). FAQ-Finder is an example of a knowledge-based FAQ retrieval system that uses semantic knowledge from WordNet (Miller, 1995) to calculate the semantic distance between the user queries and the FAQ documents (Hammond et al., 1995). The system uses an IR system SMART[11] to index the FAQ documents. These FAQ documents are text files organized into questions, answers, section headings and keywords. FAQ-Finder uses statistical term vector similarity scores between the query term vector and the FAQ documents vectors to narrow the search to a small subset of the FAQ documents (Burke et al., 1997, Hammond et al., 1995). The system uses WordNet and the marker parsing algorithm

[11] http://foundfirst.com/faq/index.htm

to find the similarity score between each word in the user query and each word in the question part in the subset of FAQ documents already selected by the IR system (Burke et al., 1997, Hammond et al., 1995). The semantic relatedness between the user query and each question (in each FAQ document) is obtained by averaging the semantic similarity scores of all the terms in the query (Burke et al., 1997). FAQ-Finder then calculates a third score which measures the degree of coverage of the user terms by the FAQ question (question part only) (Burke et al., 1997). The intuition behind this is to penalise FAQ questions that do not have corresponding words for each word in the user query (Burke et al., 1997). The final score of each FAQ document is a weighted average of the statistical term vector similarity, the semantic similarity and the coverage score.

Another system that uses lexical resources (WordNet and HowNet) for semantic concept matching was proposed by Wu et al. (2005). HowNet is a Chinese-English bilingual knowledge base that exhibits the inter-conceptual relations and inter-attribute relations of concepts in lexicons of the two languages (Dai et al., 2008, Dong and Dong, 2006, Zhan and Chen, 2011). In their approach, the input query and the question and answer part of the FAQ documents (both in Chinese) are interpreted as independent aspects. The question part in the FAQ documents and the corresponding training queries are initially classified into ten question types. Each answer part in the FAQ documents is segmented into several paragraphs. These paragraphs are clustered using Latent Semantic Analysis (LSA) (Manning et al., 2008) and the K-Means algorithm. The system uses a WordNet and HowNet based constructed ontology to obtain the semantic representation of these independent aspects. The ontology is constructed by aligning the synsets in WordNet with the corresponding Chinese words defined in HowNet. Finally, the maximum likelihood estimation in a probabilistic mixture model based on the independent aspects is adopted in the retrieval phase.

In their concluding remarks, Wu et al. (2005) suggested that their proposed approach can effectively improve the retrieval performance on the medical domain FAQ retrieval compared to the baseline FAQ-finder system. This increase in performance may be largely attributed to the fact that participants were shown the answer part before they could provide the training and testing queries. This is not ideal since the participants are likely to use the terminology used in each answer part to create the training and testing queries. Consequently, the matching of user queries to the relevant FAQ documents becomes trivial as there would be less term mismatch. In a more realistic FAQ retrieval setting, users formulate queries based on their information need and may not be aware of the terminology used in the FAQ documents.

Ontology-based approaches on the other hand require domain ontologies to resolve the syntactic and semantic difference between the user query and the relevant FAQs in the FAQ document collection (Yang et al., 2007). For example, Yang et al. (2007) describes an FAQ retrieval system on the Personal Computer (PC) domain, which uses an ontology as the key technique to pre-process the FAQs and process user query. For each user query, the system

uses ontology supported NLP to help pinpoint users' intent in order to reduce the search scope during the retrieval of the FAQ documents. The system uses full keywords matching to retrieve only the FAQ documents that contain all the trimmed query keywords. If there is no FAQ document that matches the query keywords, the system uses partial keyword matching. If more than one FAQ document is retrieved, the system employs an enhanced ranking technique, which linearly combines the Appearance Probability, Satisfaction Value, Compatibility Value, and Statistic Similarity Value to rank the FAQ documents.

Yang et al. (2008) also describes a system (FAQ-master) that uses four agents working together through an ontology supported information source in order to provide high quality FAQ answers from the web to meet the user information needs (Yang, 2007a, Yang et al., 2008). Their system uses the four agents to try to solve three aspects of web search: document content processing, user intent, and website search (Yang, 2007a, Yang et al., 2008). The first agent, the Interface Agent, uses a problem ontology, which describes the question type (when, what, could, etc.) and the question operation (support, setup, download, etc.) (Yang, 2007a,b, Yang et al., 2008). This ontology was developed by collecting a total of 1215 FAQs from the FAQ website of six motherboard factories in Taiwan. The question part of these FAQs were analysed for the question type and the intention type, and then used to construct the corresponding query and answer templates with the help of the domain ontology (Yang, 2007a,b, Yang et al., 2008). FAQ-master uses a hierarchy of the identified intention types to organise all the FAQ documents in order to reduce the search scope during the retrieval of the FAQ documents after the intention of a user query is identified (Yang, 2007a).

Furthermore, FAQ-master uses the Proxy Agent to speed up the query processing at the same time reducing the loading of the Answerer Agent with a four-tier solution finding process (Yang, 2006, 2007a, Yang et al., 2008). The four tiers are: Solution Predictor tier, Case-Based Reasoning (CBR) tier, Rule-Based Reasoning (RBR) tier and Solution Aggregation tier. For example, for a given query from the Interface Agent, the Proxy Agent first uses the Solution Predictor to check for any possible cached or predicted solutions (Yang, 2006, 2007a, Yang et al., 2008). If no solution exists, it invokes the CBR to retrieve or adapt old solutions. If no solution is found, the RBR is triggered to generate new solutions. If there is still no solution found, the query is passed to the Answerer Agent, which then performs the retrieval of the best matched FAQ documents from the Ontological Database (OD) (Yang, 2007a, Yang et al., 2008).

The Answerer agent works as a back-end process to parse, extract and transform the FAQ documents on web pages collected by the Search Agent. The Search Agent uses an ontology-supported website model (Yang, 2007a, 2008) to search and retrieve in real-time user-oriented and domain-related FAQ documents. The Answerer Agent also performs the deletion of any conflicting FAQ documents, and ranks the retrieved documents according to whether full keywords or partial keywords method is applied. The matching in performed using four ma-

trices namely, Appearance Probability, Satisfaction Value, Compatibility Value, and Statistic Similarity Value (Yang, 2007a, Yang et al., 2007) . Yang et al. (2008) reported an improved precision rate when an ontology is deployed to help resolve the syntactic and semantic differences between the user query and the relevant FAQ documents.

Casellas et al. (2007), proposes a web-based, semantically enabled FAQ search system and a case law browser application for Spanish judges. The system uses a multi-stage search approach that uses an ontological database to find the FAQ documents that match the user input query. In the first stage of this multi-stage approach, their system uses the legal topic ontology to classify the user input query to one of the legal topic classes (immigration, domestic violence, property, family issues, etc.). The main purpose of this stage is to narrow the search to one of these classes. Legal experts manually classified the FAQ documents in the FAQ document collection to the different classes. A topic ontology was then constructed from these classes using OntoGen V 2.0. One of the shortcomings of the first stage as highlighted by the authors is that sometimes the user queries may be misclassified thus focusing the search within a wrong class.

In the second stage, keyword matching is applied to all the FAQ documents in the selected class. The purpose of this stage is to filter out non-matching FAQ documents within this class in order to remain with a small candidate set of FAQ documents. In the third and final stage, the system computes the semantic distance between the user input query and all the candidate FAQ documents selected by the second stage. To compute this semantic distance, the input query is first parsed to identify grammatical patterns. These patterns are searched for in the Ontology of Professional Judicial Knowledge (OPJK) in order to build a graph path. The system uses the semantic distance algorithm to match the user question ontology graph path with the legal ontology graph path of each candidate FAQ document. The FAQ documents with the smallest distance is the best match to the user query. In their evaluation, Casellas et al. (2007) compared the results obtained using a typical keyword based search engine and the results obtained on the application of keyword search in combination with ontology-based search. Their results suggest that the combination of keyword search strategy and ontology-based search yielded better results than when only keyword search is deployed.

In another study, Fang et al. (2008) uses a domain ontology in an Agricultural FAQ retrieval system in order to align the concepts in the query with those in the FAQ documents. They use the domain ontology to detect the FAQ documents that contain keywords that semantically match the user query. Just like in previous studies reviewed thus far, they also applied statistical keyword matching to calculate the similarity between the user query and each FAQ document in the collection. Their findings suggest that ontology-based automatic classification of user queries can effectively improve the performance of the baseline Agricultural FAQ retrieval system that uses only keyword matching.

2.2.1.1 Summary

This section reviewed approaches that use knowledge bases to help resolve the syntactic and semantic difference between the user query and the relevant FAQs in the FAQ document collection. One of the most widely used lexical resources for building knowledge bases to help resolve the term mismatch problem is WordNet. Although, it has been widely used in the QA and FAQ retrieval community, different authors have reported mixed results (Fang, 2008, Voorhees, 1994b). In particular, Fang (2008) and Wu et al. (2005) have shown significant performance improvement when lexical resources such as WordNet are used for semantic concept matching. On the other hand, Voorhees (1994b) did not show any significant improvement if queries are expanded with terms from WordNet. One plausible explanation for this dissimilarity in the findings by different authors is that the selection of terms for semantic concept matching is usually automatic. This may result in the selection of irrelevant terms when the same term has different meanings in different contexts. Ontology-based approaches on the other hand are not robust as they require a lot of domain modelling when ever application domain changes. For example, FAQ-master uses a PC ontology created by the authors using the Protege framework (Duineveld et al., 2000) for building ontologies and the system by Casellas et al. (2007) uses an ontology of Professional Judicial Knowledge developed by legal experts.

2.2.2 Template-Based Approaches

Frequently Asked Question lists were invented because it was clearly evident that people who share the same interest tend to ask the same question over and over again (Sneiders, 2009). The whole paradigm of the template-based approach in FAQ answering systems relies on this recurring nature of user queries (Andrenucci and Sneiders, 2005, Sneiders, 2009). Auto-FAQ (Whitehead, 1995), Omnibase (Katz et al., 2002) and the START natural language system (SynTactic Analysis using Reversible Transformations) (Katz, 1997) are examples of early representative template-based FAQ retrieval systems.

In Auto-FAQ, the system does not perform deep semantic analysis of the user query before the retrieval of the relevant FAQ documents (Whitehead, 1995). Deep semantic analysis refers to the process where by the system tries to comprehend the user query. This is usually accomplished by using an external resource such as WordNet to determine if two different terms are related by considering many WordNet relationships (hypernyms, synonyms, metonym etc.). Auto-FAQ however, relies on shallow language understanding – where the system does not comprehend the user query and the matching of the users' queries to the FAQ documents is based on keyword comparison. The question-answer pairs (FAQ documents) stored in the information source have an additional field, the context field. This field stores

a series of comma delimited phrases and keywords, which together help to enhance the vocabulary and help to focus the search onto a specific topic (Whitehead, 1995). In Auto-FAQ, domain experts are responsible for adding suitable keywords and phrases to these context fields. The rationale for adding these additional terms is to help to bridge the term mismatch problem between the user query and the relevant FAQ document. One of the drawback of the approach proposed in Auto-FAQ is that the keywords are generated by the domain experts and they might not cover the lexicon that a typical user of the system may use.

Another system that relies on manual rules is the Sneiders' system (Sneiders, 1999). The Sneiders' system also follows the Auto-FAQ approach. Unlike Auto-FAQ, each FAQ entry in the Sneiders' system is stored with required, optional and forbidden keywords specified. Each of these keywords are enhanced with synonyms and phrases. For each user query, the Sneiders' system uses Prioritised Keyword Matching algorithm to match the user query to each FAQ entry in the database separately (Sneiders, 1999). The system uses the aforementioned algorithm to reject a match between the user query and the FAQ document if at least one required keyword does not appear in the query terms. The system also matches the optional keywords of each FAQ entry to the user query. If there is more than 1 optional keyword missing in the user query, the system rejects the match between the user query and the FAQ entry. The forbidden keyword is used to reject a match between the user query and the FAQ entry if the user query has at least one forbidden keyword specified in the FAQ entry.

Sneiders' also used the notion of question templates to adapt the earlier approach in order to create a question answering interface for a relational database (Sneiders, 2002a,b). This involved replacing static FAQ entries with dynamic question and answer templates. A question template is a question with entity slots for data instances that represent the main concepts of the question. The data instances are stored in the underlying relational database and are bound to the question templates (Sneiders, 2002a,b). In this new approach, when the question answering system receives a user query, it selects tokens in the query for closer examination in order to retrieve data instances that are relevant to the user query (Sneiders, 2002a,b). The system then retrieves question templates bound to these data instances. The retrieved data instances and the question templates are then combined to create one or several interpretations of the original question (Sneiders, 2002a,b). The user then selects the interpretation to be answered and the system returns the associated answer (Sneiders, 2002a,b).

In another template-based FAQ retrieval system, Brill et al. (2001) proposed a data intensive question answering system that uses the web as an external source to find possible answer strings. In their study, they used the TREC-9 QA queries 201 – 400 as training data. They manually reformulated each query in the training data and they used these new query reformulations to search the web for the best 100 matching pages. Summaries for these returned pages were then harvested to extract a set of potential answer strings. Each potential answer string was then weighted by how well it matched the expected answer type and how often it

occurred in the retrieved page summaries. The best 5 possible answer strings were used as the new query to retrieve 5 possible supporting documents in the local collection (TREC QA document collection). The Okapi Best Match ranking function (BM25) (Robertson et al., 1996) was used in the retrieval phase. The final answer was generated from these retrieved supporting documents. A reasonably high Mean Reciprocal Rank (MRR) of 0.437 was reported in their TREC 2001 results. In their concluding remarks, Brill et al. (2001) suggested that although the answer projection approach was designed to work for the TREC QA track, it is more generally applicable to other datasets as it depends on data redundancy rather than sophisticated linguistic analyses.

Another template-based approach that relied on redundant web data to automatically learn regular expressions that are normally used to answer open domain fact based questions in the TREC-10 QA collection was proposed by Ravichandran and Hovy (2002). Their approach uses the Machine Learning Technique of bootstrapping to build a large corpus starting with only a few examples of QA pairs (Ravichandran and Hovy, 2002). In the learning phase, for each question type (BIRTHDATE, LOCATION, DEFINITION, etc.), the query string and the answer string are selected and then submitted to the AltaVista search engine. The top 1000 web documents are retrieved for each query. A sentence breaker is then applied to the retrieved documents and only those sentences that contain the query and the answer terms are retained. Each retained sentence is passed through a suffix tree constructor to extract the longest matching sub-string with a score representing the length of that sub-string. Only the phrases that contain the query and answer terms are extracted.

For each extracted phrase, the query terms are replaced by the tag <NAME> and the answer terms used in the query are replaced by the tag <ANSWER>. For example, for the BIRTHYEAR question type query, "MOZART 1756" submitted to AltaVista and one of the phrase extracted from the retained documents "MOZART was born in 1756", the text pattern learned will be "<NAME> was born in <ANSWER>". For each learned text pattern, a precision score is calculated and only the top 5 ranked text patterns are retained. This precision score is the probability of each text pattern containing the answer (Ravichandran and Hovy, 2002). For each question type (BIRTHDATE, LOCATION, DEFINITION e.t.c) fewer than 10 query examples were used to learn the text patterns.

In the retrieval phase, the unseen query term is first analysed for its query type (BIRTHDATE, LOCATION, DEFINITION e.t.c). The query term is then used to search a local collection of documents (TREC-10 collection). All the documents retrieved are then segmented into sentences. Each query term in the retrieved sentences is then replaced by the query tag <NAME>. Using the previously created text patterns for the particular question type, each retrieved sentence is then scanned for the presence of each text pattern and words matching the answer tag *ANSWER* are selected as the answer. The answers are then sorted by their patterns precision scores and duplicates are discarded. Only the best 5 matching answers are

returned. Their results suggest that the retrieval performance in an open domain fact based question answering system can be improved markedly by using the text pattern learned from the web to help pinpoint answers to user queries in a local collection (e.g TREC corpus).

In another template-based approach, Moreo et al. (2012a) introduced a new algorithm called Minimal Differentiator Expression (MDE). In their approach, they solve the term mismatch problem by using linguistic classifiers trained using expressions that totally differentiate each FAQ. These expressions/query templates were generated using the aforementioned MDE algorithm. They enhance the performance of their system during the life of its operation by continuously training the classifier with new evidence from the users' queries. In their evaluation, they reported that their approach outperformed the cluster-based retrieval proposed in Kim et al. (2007). This cluster-based retrieval will be reviewed in Section 2.2.3.

More recently, Moreo et al. (2013) proposed a semi-automatic method for creating query templates from a collection of previously collected query reformulations. In their approach, for each query reformulation, a query template is generated using the MDE algorithm(Moreo et al., 2012a). These query templates were regarded as candidate solutions for the corresponding FAQ documents. In the retrieval phase, two optimisation strategies were deployed : Simulated Annealing (Cerny, 1985, Kirkpatrick et al., 1983) and Genetic Programming (Koza, 1992). For each query, an initial solution (regular expression) is created using the MDE algorithm. Neighbouring solutions to this initial solution for the query were then obtained from the previously created sets of candidate solutions (query templates) using the aforementioned optimisation strategies. These strategies optimises the quality of the template set based on their degree of correctness, generalization and interpretability. In their evaluation, they compared their approach to several other classifier based retrieval strategies based on SVM_s (Zhang and Lee, 2003) and AdaBoost (Esuli et al., 2008). They reported an improvement in retrieval performance when SVM_s and AdaBoost are used compared to their proposed approach. However, the optimisation search strategies proposed in their study performed better than the classical Term Frequency-Inverse Document Frequency (TF-IDF) (Robertson, 2004) and the MDE approach without the optimisation algorithms.

2.2.2.1 Summary

Section 2.2.2 reviewed template-based FAQ retrieval approaches proposed in the literature. Evidence from this survey suggest that a template-based approach is more portable and can be easily adapted to other domains. The effectiveness of this approach was first acknowledged at the TREC-10 QA evaluation (Voorhees, 2001) after the winning submission by Soubbotin and Soubbotin (2001) used a fairly extensive list of surface patterns. Although some template-based approaches have been found to perform well in fact based open domain QA answering systems, these approaches might not work well in FAQ retrieval systems as

they were designed to extract answers from the retrieved set of documents. For example, the approaches proposed in Brill et al. (2001), Ravichandran and Hovy (2002) relies heavily on redundant web data to learn text patterns and this might not work well in FAQ retrieval systems. This is due to the fact that some FAQ documents in the FAQ document collection (HIV/AIDS answer booklet) might not have a lot of related content on the web. However, this can be alleviated by manually creating query templates for those FAQ documents by the domain expert as proposed by Sneiders (2002a,b). Alternatively, these query templates can be automatically generated using approaches such as MDE, which was proposed by Moreo et al. (2012a). The main drawback with these template-based approaches is that they do not take into consideration the importance of each term used in creating these templates when ranking the FAQ documents. As per our thesis statements, we are proposing a template-based approach that uses information from previous searches to enriching the FAQ documents with additional terms from a query log in order to reduce the term mismatch problem between the users' queries and the relevant FAQ documents.

2.2.3 Statistical Information Retrieval and Machine Learning Based Approaches

Evidence from previous studies suggest that knowledge based FAQ retrieval systems provide precise answers in general to a majority of user queries (Romero et al., 2013). However, they require a lot of knowledge modelling (Romero et al., 2013). In order to overcome this disadvantage, several authors have proposed statistical keyword matching methods (Berger et al., 2000, Kim et al., 2007, Kim and Seo, 2006, Romero et al., 2013).

For example, Berger et al. (2000) explored five different statistical algorithms for mining correlations between questions and answers from two different collections of FAQ documents in order to address the term mismatch problem between the user query and the relevant FAQ documents. One of the FAQ document collections was made up of Usenet[12] FAQ documents selected from the comp. * Usenet hierarchy containing 1800 FAQ documents. The other was a collection of questions submitted by customers to the Ben & Jerrys call centre along with the answer supplied by a customer representative. Their baseline system was based on the traditional TF-IDF ranking approach. The first learning technique they used was the adaptive TF-IDF. This retrieval approach merely used the held out training data to adjust the Inverse Document Frequency (IDF)-weights of each word so as to maximise the retrieval of the correct answer for each question in the training set (Berger et al., 2000). This approach does not address the term mismatch problem between the query and the relevant FAQ documents. The approach only exploits the labelled training data to improve the baseline TF-IDF retrieval performance.

[12]http://www.faqs.org/faqs/

Furthermore, Berger et al. (2000) deployed query expansion in an attempt to address the aforementioned term mismatch problem. They learned the expansion terms by calculating the mutual information between the query terms and the answer terms in the training set. In their experimental evaluation, they expanded the unseen queries with terms that have the highest mutual information. In their concluding remarks, they reported an improvement in the retrieval performance when each query term was expanded with at least one word having the highest mutual information to that query term. In the third learning approach, they attempt to solve the aforementioned term mismatch problem by training a translation model with a sufficient amount of question-answer pairs in order to learn how answer-words translate to question-words (translation probabilities). They use the IBM 1 translation model (Brown et al., 1993) in order to learn these translation probabilities. Having learned these translation probabilities, they deployed a translation based retrieval system to equate the relevance of an answer to any unseen user query (Berger et al., 2000).

In their evaluation, Berger et al. (2000) reported a significant improvement when a translation based system is used to find the relevant answer compared to when the traditional TF-IDF is used. Although the results presented are promising, a lot of training data is needed to learn very good translation probabilities. In the final learning strategy, they investigated the use of a latent variable model. In the latent variable model, each question and answer are considered to belong to a cluster. Therefore, they used a training set of question-answer pairs to learn a factored model that maps words used in a similar context to each other. In the retrieval phase, this factored model of translation was used to find the answer documents that closely matched the topic/cluster that generated the query (Berger et al., 2000). In their experimental evaluation, they reported improved retrieval performance compared to the traditional TF-IDF.

Jeon et al. (2005) on the other hand proposed a method that relied on the similarity between the answers in the question-answer archive to estimate probabilities for a translation-based retrieval model (in the language modelling framework). In this approach, they collected 68000 question-answer pairs from Naver[13], a leading South Korean portal. From this collection, they randomly selected 50 documents to use for testing and evaluation. The remaining question-answer pairs were used for learning the translation model.

In addition, Jeon et al. (2005) devised a new algorithm, the LM-HRANK to use for automatic identification of semantically similar question-answer pairs in the training data. This algorithm was based on the notion that, if an answer A retrieves an answer B at rank r_1 and answer B retrieves answer A at rank r_2, then the similarity between the two answers is defined as the reverse harmonic mean of r_1 and r_2, as shown in Equation 2.1.

[13]http://www.naver.com

$$sim(A, B) = \frac{1}{2} \times \left(\frac{1}{r_1} + \frac{1}{r_2} \right) \tag{2.1}$$

Using this algorithm, they empirically set a threshold value of (0.005) to judge whether an answer pair in the training set is related. Based on this algorithm and the threshold, they learned/identified 331965 answer pairs that had scores above the threshold. A collection of similar question pairs was created based on the identified answer pairs. This new collection was used to estimate words translation probabilities using the IBM 1 translation model (Brown et al., 1993). They used this translation model to retrieve the relevant question-answer pairs for the held out testing data. When they compared the retrieval performance of this translation model to different baseline retrieval models; the vector space model with cosine similarity, the BM25 (Robertson et al., 1996) weighting model and the query-likelihood language model, they reported statistically significant improvement in retrieval performance across all evaluation measures. The approach proposed by Jeon et al. (2005) is more robust and adaptable to different domains. The only limitation with this approach is that a lot of training data is needed to learn good translation probabilities. Xue et al. (2008) also used this translation based language model to solve the term mismatch problem.

Kim et al. (2007) investigated the use of query logs as knowledge sources to solve the term mismatch problem between the user query and the FAQ document collection. In their investigation, they used an FAQ Retrieval and Clustering Technique (FRACT), which was made up of two sub-systems, a query log clustering system and a cluster-based retrieval system. The query log clustering system considers each FAQ as an independent category and it periodically collects and refines users' queries, which are then classified into each FAQ category by using a vector similarity in the latent semantic space. Based on the classification results, the query log clustering system groups the query logs and computes centroids for each query log cluster.

Furthermore, for each and every user query, the cluster-based retrieval system calculates the similarity between the user query and the FAQs smoothed by query log clusters and then ranks and returns a list of FAQs based on those similarities. To evaluate the effectiveness of their approach, they implemented two versions of their system to perform query classification in the original term-documents space and the latent term weight space. By comparing the performance of each version of their system, they observed that query classification in the latent term weight highly outperformed the query classification in the original term-documents weights.

To evaluate the effectiveness of FRACT as compared to conventional IR systems, they implemented two baseline IR systems using the Lemur Toolkit version 3.0[14]. One system used the BM25 (Robertson et al., 1996) weighting model while the other used the Kullback-Leibler (KL) divergence language model (Zhai, 2008) using Jelinek-Mercer (JM) smoothing (Zhai

[14]http://www.lemurproject.org/

and Lafferty, 2001). Their results suggest that FRACT outperformed the implemented baseline IR systems in both average MRR and average miss rate (MissRate). MRR measures the average value of the reciprocal ranks of the first relevant FAQs given by each query, as shown in Equation 2.2 while the MissRate computes the ration that the search engine fails to return the relevant FAQs given by Equation 2.3 . In Equation 2.2, $rank_i$ is the rank of the first relevant FAQ given by the i_{th} query, and num is the number of queries.

$$MRR = \frac{1}{num} \times \sum_{i=1}^{num} \frac{1}{rank_i} \tag{2.2}$$

$$MissRate = \frac{\text{the number of failure queries}}{\text{the number of queries}} \tag{2.3}$$

To evaluate the performance of the query log clustering, they used the F1 measure as shown in Equation 2.4 and their results suggest that query log clustering using Latent semantic space out performs query log clustering in the original vector space. In Equation 2.4, P is the precision, which is the proportion of correct ones out of returned query logs. R is the recall rate that means proportion of returned query logs out of classification target (Kim et al., 2007).

$$F1 = \frac{2PR}{P + R} \tag{2.4}$$

Even though FRACT outperformed the implemented baseline IR systems, when they analysed the cases where FRACT failed to rank highly relevant FAQs, they found that there were still lexical disagreement problem between the user's queries and the FAQs. One main reason was that there were many cases where there was little overlap between the words in the queries and the query log clusters. Also they found that there were many cases where the query was relevant to many FAQ categories. One other deficiency that they observed is that there were instances where the implemented baseline IR systems outperformed the FRACT system.

Kim and Seo (2006) also proposed another clustering technique, which they used as a form of document smoothing based on a Machine Readable Dictionary (MRD) to resolve the term mismatch problem between the users' queries and the FAQ collection. Their study was motivated by results from earlier studies (Tombros et al., 2002, Willett, 1988), which suggests that cluster-based retrieval only outperforms document-based retrieval on collections of documents of small sizes (Kim and Seo, 2006). The proposed system, High Performance FAQ retrieval of FRACT has two subsystems, FRACT/CL and FRACT/IR.

FRACT/CL periodically collects query logs and clusters them using the original FAQs as seed data. To cluster the query logs, they used a modified K-means algorithm that uses the MRD for similarity measure. This similarity measure is called DicSim since it uses definitions of words. This similarity measure is based on two steps; the first step is word by word

comparison of query logs and the second step is based on the proportion of overlap between definitions. However, since most contents words have more than one definitions, all the definitions are selected from the MRD and their significance score computed. The definition with the highest significance score is selected for use in computing the similarity measure.

The evaluate the performance of their clustering approach, they compared the average precision and average recall of FRACT/CL using the DicSim and three other popular similarity measures, the cosine measure (COS) (Jardine and Rijsbergen, 1971), the DICE coefficients (DICE) (Jardine and Rijsbergen, 1971) and the Jaccard Coefficients JACCARD (Jardine and Rijsbergen, 1971). Their results suggest that the DicSim outperformed all other measures (COS, DICE and JACCARD) in both average precision and average recall rate.

When they compared the average performance of FRACT/IR with traditional IR system, they observed that FRACT/IR managed to reduce the average miss rate on this traditional IR system by 5.2% – 9.6%. These results suggest that FRACT/IR can resolve the lexical disagreement problem between queries and FAQs. When comparing FRACT/IR to the IDEAL-FRACT/IR (where precision and recall of FRACT/CL average is 1.0), they observed that IDEAL-FRACT/IR outperforms FRACT/IR. Their results suggest that the more the performance of FRACT/CL is increased, the more the performance of FRACT/IR can also be increased (Kim and Seo, 2006).

2.2.3.1 Summary

In summary, this section reviewed statistical and machine learning approaches that are currently being used to resolve the term mismatch problem that is prevalent in FAQ retrieval systems. The main advantages of the statistical-based approaches is that they are more potable and efficient and they do not require any domain modelling or manual construction of domain ontologies when ever domains are changed. However, they are only desirable when dealing with large FAQ document collections containing several thousands FAQ documents.

2.2.4 Conclusion

A thorough review of the different approaches for developing Desktop-Based FAQ retrieval systems was presented in this section. Throughout this review, a considerable amount of literature investigating how to resolve the term mismatch problem between the users' queries and the relevant FAQ documents in the collection was consulted. Different authors proposed different approaches for resolving this term mismatch problem in order to improve the ranking of the relevant FAQ documents for each user query. In particular, they enhanced the FAQ documents representation through different approaches. For example, they used knowledge-based sources, query templates, statistical and machine learning techniques. All

the approaches proposed used a ranking function to rank this new document representation for each user query. The main disadvantage of knowledge-based systems that was identified throughout the survey is that they are not portable as they require a lot of domain modelling when ever application domain changes. For example, FAQ-master uses a PC ontology created by the authors using the Protege framework (Duineveld et al., 2000) for building ontologies and the system by Casellas et al. (2007) uses an ontology of Professional Judicial Knowledge developed by legal experts (Casellas et al., 2007). On the other hand, the template-based and the statistical-based techniques are more portable as they can be easily adapted across different domains. However, statistical-based approaches have been found to be effective and efficient only in very large documents collections. Since the FAQ document collection used in this study is very small, this thesis will use a template-based approach in order to help resolve the term mismatch problem between the queries and the FAQ documents. The next section presents FAQ retrieval approaches adapted in SMS-Based FAQ retrieval systems. We will see in the next section that apart from solving the text mismatch problem that is prevalent in Desktop-Based FAQ retrieval systems, SMS-Based FAQ retrieval systems also require an additional step to address the challenges that were outlined in Chapter 1 (Section 1.3), in particular, detecting queries for which there are no relevant FAQ documents in the collection.

2.3 Review of SMS-Based FAQ Retrieval Systems

SMS-Based FAQ retrieval systems are those that can be queried by users through SMS to provide answers, which are related to the users' SMS queries. They have unique problems because their keypads and screen displays are very small. For example, we have seen in Chapter 1 (Section 1.3) that noisy text is very common in SMS retrieval because of the size of the keypad used for entering text. Another problem that is very specific to low-end mobile phones is that there is a limited number of FAQ documents that can be displayed on the mobile phone displays.

In order to address some of the aforementioned problems, current work on SMS-Based FAQ retrieval is focused on the following research themes: SMS normalisation, the retrieval of the FAQ documents that the system believe are relevant to the user query and the identification of Missing Content Queries (MCQs) (Hogan et al., 2011, Vilario et al., 2013). MCQs are those queries for which there are no relevant documents in the collection (Yom-Tov et al., 2005). State-of-the-art approaches train a binary classifier to detect these MCQs (Contractor et al., 2013, Hogan et al., 2011, Vilario et al., 2013). SMS normalisation on the other hand involves correcting a noisy SMS query so that it closely resembles the text in the FAQ documents (Hogan et al., 2011). Noise in an SMS query can be the result of spelling

errors, abbreviations, and word spacing errors, e.g 'lemme' –> 'let me' (Byun et al., 2008, Kothari et al., 2009).

Several approaches have been proposed in the literature for transforming a noisy SMS query Q to a more grammatical form (Kothari et al., 2009). Some systems exist that use a Hidden Markov Model (HMM) approach to transform noisy SMS queries to clean SMS queries. These systems have been found to be reasonably effective (Choudhury et al., 2007, Contractor et al., 2013). Other approaches proposed in the literature rely on Statistical Machine Translation (SMT) to transform a noisy SMS query to clean SMS query (Contractor et al., 2010). A traditional SMT system relies on a large parallel datasets of clean and noisy text to learn the translation model of the parallel dataset and the language models that can be used by the decoder for translation (Brown et al., 1993). The lack of such a parallel datasets of clean and noisy text makes it very difficult to use SMT for cleaning noisy text (Contractor et al., 2010). The translation models are needed in the decoding process to ensure that the noisy text and the clean text are good translations of each other while the language models are needed to ensure that the output is grammatically correct (Contractor et al., 2010).

Since 2011, the FIRE[15] has been organising three different SMS-Based FAQ retrieval tasks in order to advance research on the aforementioned research themes. These tasks are: The mono-lingual FAQ retrieval task (Section 2.3.2), the cross-lingual FAQ retrieval task (Section 2.3.3) and the multi-lingual FAQ retrieval task (Section 2.3.4). This section will conduct a comprehensive review on current approaches adapted for SMS normalisation and MCQs detection. The review will begin with a description of the standard test collection (dataset) used at the FIRE SMS-Based FAQ retrieval tasks in Section 2.3.1. This will be followed by a description of the aforementioned tasks in Section 2.3.2, 2.3.3 and 2.3.4. Section 2.3.5 will conduct a review on work that addresses both the MCQs and the SMS normalisation problems using the standard FIRE SMS-Based FAQ retrieval dataset. Section 2.3.7 will conduct a review on several other SMS-Based FAQ retrieval approaches that do not address the MCQs detection problem.

2.3.1 Test Collection for the FIRE SMS-Based FAQ Retrieval Tasks

The FAQ document collection for the FIRE2011 SMS-Based FAQ retrieval tasks (mono-lingual, cross-lingual, multi-lingual) contained 7251 English FAQ documents, 1994 Hindi FAQ documents and 681 Malayalam FAQ documents (Bhattacharya et al., 2013, Contractor et al., 2013). The English FAQ documents (FAQs) were collected from several websites including but not limited to banks, railway services and government departments. These FAQs covered 15 domains namely: Career, Agriculture, General Knowledge, Health, Insurance,

[15]http://www.isical.ac.in/fire/2011/index.html

Sports, Tourism, Bank, Loan, Personality Developments, Recipes, Visa, Web, Telecommunications and Railways. Amongst the Hindi FAQs, some were collected from websites, while other FAQs were manually generated by translating the English FAQs into Hindi (Contractor et al., 2013). In total, there were 10 domains making up the Hindi FAQ document collection. These domains were: Agriculture, Bharat, Business, Constitution, General Knowledge, Health, Railways, Rajya Sabha, Telecommunication and Videsh (Contractor et al., 2013). The Malayalam FAQs were obtained by manually translating the English FAQs of the following domains: Railways, General Knowledge and Telecommunications.

The English SMS queries used for training and testing in the FIRE2011 SMS-Based FAQ retrieval tasks were generated by several volunteers who were told the domains and shown the FAQs in these domains by the tasks organisers (Contractor et al., 2013). The volunteers were instructed to write the SMS queries as they would normally write an SMS. On the other hand, the Hindi and Malayalam training and testing SMS queries were generated by dropping dialectic marks and some non-content words (Contractor et al., 2013). Table 2.2 shows the number of queries collected for each task (mono-lingual, cross-lingual, multi-lingual) (Bhattacharya et al., 2013). These SMS queries were divided into training and testing sets. In the cross-lingual retrieval task (Section 2.3.3), participants only had to retrieve the Hindi FAQ documents that were relevant to the English SMS queries provided (Bhattacharya et al., 2013). It is for this reason that in the cross-lingual retrieval task only the English SMS queries are shown in the training and testing set in Table 2.2.

Table 2.2: Details of the SMS queries used for training and testing in the FIRE2011 SMS-Based FAQ retrieval tasks. The percentage of the SMS queries having relevant FAQ documents in the collection is shown in parenthesis (Bhattacharya et al., 2013).

Task	**Training Set**			**Testing Set**		
	English	Hindi	Malayalam	English	Hindi	Malayalam
Mono-lingual	1071 (64.5%)	230 (78.6%)	140 (85.7%)	3405	324	50
Cross-lingual	472 (61.6%)	–	–	3405	–	–
Multi-lingual	460 (63.0 %)	230 (79.5%)	80 (75.0%)	3405	324	50

2.3.2 Mono-Lingual SMS-Based FAQ Retrieval

In the mono-lingual FAQ retrieval task, researchers were provided with a standard collection of FAQ documents and SMS queries in the same language. The goal of the task was to retrieve the top 5 FAQ documents that best matches each SMS query of the same language as illustrated in Figure 2.1 (Contractor et al., 2013). The SMS queries used in this task were fewer than 160 characters in length and contained typical noise such as shortened and non-grammatical text (Contractor et al., 2013). Some of the SMS queries provided for this task

did not have relevant FAQ documents in the FAQ document collection to be searched. Contractor et al. (2013) referred to these SMS queries as out-of-domain. This thesis however, will follow the definition by Yom-Tov et al. (2005) and refer to these type of queries as MCQs. Such queries are common in a realistic scenario as users are often unaware of the contents of the FAQ documents in the FAQ document collection (Yom-Tov et al., 2005). Section 2.3.1 provides details of the test collection used in this task. In this thesis, the FAQ documents and the SMS queries used will be of same language (English).

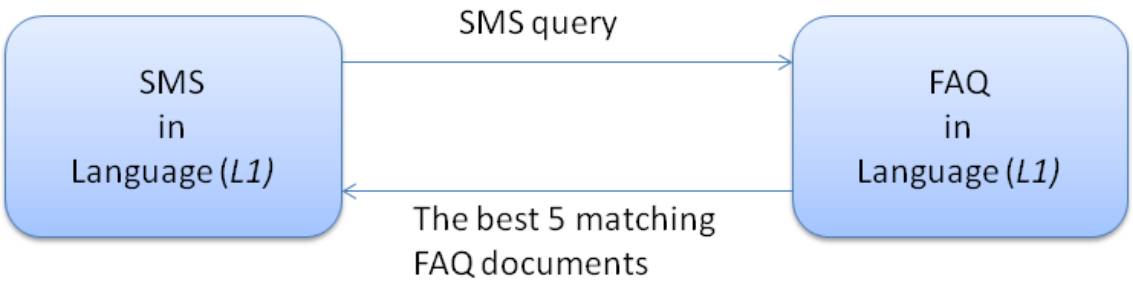


Figure 2.1: SMS-Based FAQ Mono-Lingual retrieval task. The FAQ document collection and the SMS Query are in the same language (L1) (Contractor et al., 2013).

2.3.3 Cross-Lingual SMS-Based FAQ retrieval

The cross-lingual FAQ retrieval task is similar to the mono-lingual retrieval task, except that the SMS query and the collection of FAQ documents are written in different languages (Contractor et al., 2013). Hence, the goal of this task was to retrieve the top 5 FAQ documents from the set of FAQ documents written in Language $L2$ (e.g Hindi) that best matches each SMS query written in Language $L1$ (e.g English) as illustrated in Figure 2.2 (Contractor et al., 2013). A typical solution for this task will be to first correct the English SMS queries for spelling errors. These clean SMS queries (English) can then be translated to the language used in the collection of the FAQ documents (Hindi) in order to reduce the problem to a mono-lingual retrieval task. Section 2.3.1 also provides details of the test collection used in this task.

2.3.4 Multi-Lingual SMS-Based FAQ retrieval

In the multi-lingual FAQ retrieval task, participants were provided with FAQ documents and SMS queries written in multiple languages. In essence, some FAQ documents in the FAQ document collection were written in language $L1$ (English), other FAQ documents were written in language $L2$ (Hindi) and others were written in language $L3$ (Malayalam). The SMS queries were also expressed in one of the many languages as shown in Figure 2.3. The

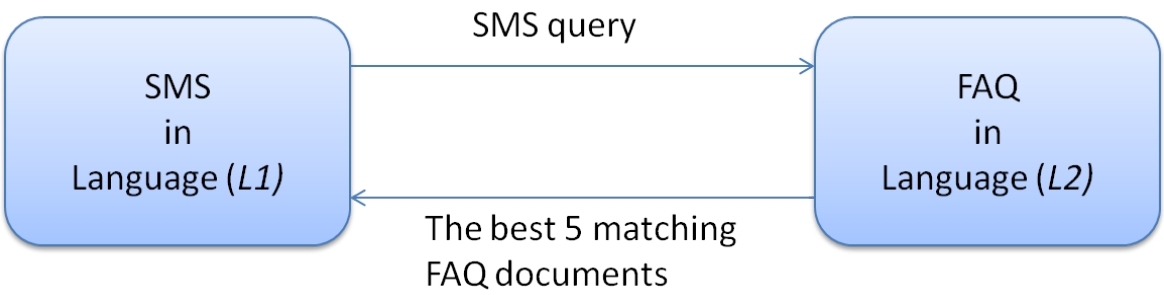


Figure 2.2: SMS-Based FAQ Cross-Lingual retrieval task. The FAQ document collection and the SMS Query are in different languages. SMS query may be written in English (language L1) while the FAQ documents are written in Hindi (language L2) (Contractor et al., 2013).

most trivial way to address this problem is to separate the FAQ documents into respective Languages, where each sub-collection only contains FAQ documents in one language. Each incoming SMS query that has been corrected for spelling errors can then be translated into each of the languages used in the sub-collections of the FAQ documents. Hence, following the aforementioned approach, this multi-lingual retrieval task reduces to a mono-lingual retrieval task because each translated SMS query can be sent separately to the appropriate sub-collection for the retrieval of the relevant FAQ documents. The retrieved FAQ documents from the different sub-collections of FAQ documents in different languages can then be merged into a single result set to be returned to the user. Section 2.3.1 also provides details of the test collection used in this task.

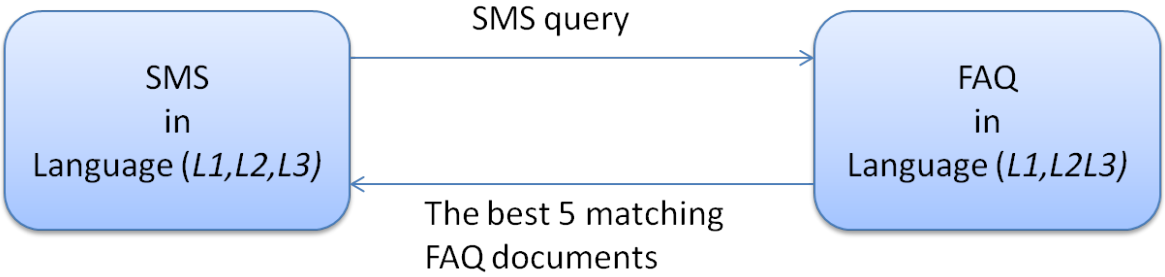


Figure 2.3: SMS-Based FAQ Multi-Lingual retrieval task. The FAQ document collection and the SMS Query are in different languages. SMS query may be written in any of the languages L1,L2 and L3 while the FAQ documents are also written several languages L1,L2 and L3 (Contractor et al., 2013).

2.3.5 Missing Content Queries Detection and SMS Normalisation

Systems that participated at the FIRE2011 SMS-Based FAQ retrieval task were the first to address both the detection of MCQs and the SMS normalisation problem in an SMS-Based FAQ retrieval setting. Almost all the teams that participated in this task used the SMS queries

in the training set to learn translation rules for normalising the SMS queries. These learned translation rules were then used to transform the noisy SMS queries in the testing set into the normalised correct forms (Contractor et al., 2013). In Table 2.3, we provide examples of noisy and normalised SMS queries. After normalisation, the SMS queries were used to retrieve a ranked list of FAQ documents from the collection of FAQ documents (Contractor et al., 2013). In the final step, most systems used the information from this retrieved results to identify the MCQs.

Table 2.3: Examples noisy and normalised SMS queries.

Noisy SMS Queries	**Normalised SMS Queries**
Are the carier conselling sessionss confidensial?	Are the career counselling sessions confidential?
Whr can i find info abt pesticide estb reg and rep?	Where can I find information about pesticide establishment registration and reporting?
Wat precautns are necc 2 store paddy?	What precautions are necessary to store paddy?
Hows smallpox sprd?	How is smallpox spread?
hw 2 buy season tkts?	How to bus seasonal tickets

For example, the system that performed better than other systems in the English monolingual retrieval task by Hogan et al. (2011) first created rules for SMS normalisation by manually correcting the SMS queries in the training set and then learning the correction rules from them. These learned correction rules were used to normalise each SMS query in the testing set. After normalisation, the SMS queries were used to retrieve a ranked list of FAQ documents by combining the results of three different retrieval approaches (Solr BM25 (Robertson et al., 1996), Lucene BM25, and a simple word overlap metric).

In order to identify the MCQs, Hogan et al. (2011) combined 3 different lists of MCQs generated through three different approaches and then applied a simple majority voting approach to identify MCQs in an SMS-Based FAQ retrieval setting. The first list of candidate MCQs was generated using an approach proposed by Ferguson et al. (2011) for determining the number of relevant documents to use for query expansion. In this approach, a score for each query was produced based on the IDF component of the BM25 (Robertson et al., 1996) for each query without taking into consideration the term frequency and the document length. First, the maximum score possible for any document was calculated as the sum of the IDF scores for all the query terms. Following this approach, documents without all the query terms generated a score less than the maximum score. A threshold was then used to determine if a query should be added to the list of candidate MCQs. They added queries that had all their document scores below 70 % of the maximum score to this list.

The second list of candidate MCQs was generated by training a K-nearest-neighbour classifier to identify MCQs and non-MCQs. The features used to train this classifier included

query performance estimators: Average Inverse Collection Term Frequency (AvICTF) (He and Ounis, 2006), Simplified Clarity Score (SCS) (He and Ounis, 2004), the derivatives of the similarity score between collection and the query (Summed Collection Query Similarity (SumSCQ), Averaged Collection Query Similarity (AvSCQ), Maximum Collection Query Similarity (MaxSCQ)) (Zhao et al., 2008), result set size and the un-normalised BM25 (Robertson et al., 1996) document scores for the top five documents. Their classifier achieved 78% (80% non-MCQs and 76% MCQs) accuracy on the FAQ SMS training data using a leave-one-out validation.

The third list of candidate MCQs was generated by simply counting the number of term overlaps for each incoming query and the highest ranked documents (For example, if a query consists of more than one term and had only one term in common with the document, that query was marked as a MCQs). Hogan et al. (2011) used the held-out training data to evaluate their approach and they concluded that combining the three lists of candidate MCQs through a simple majority voting yielded better results. Their system produced the best overall retrieval performance with a mean reciprocal rank (MRR) of 0.89. The system by Hogan et al. (2011) also performed better than other systems that particpated at the FIRE2011 SMS-Based FAQ retrieval system task in the detection of MCQs and non-MCQs. Using the three combined lists of candidate MCQs, they reported a fairly high detection rate of 69.78% for the non-MCQs and 86.18% for the MCQs.

The second best SMS-Based FAQ retrieval system that participated at the FIRE2011 English mono-lingual retrieval task by Shivhre (2013) differed with the general trend followed by other teams. Unlike other teams, Shivhre (2013) combined the SMS normalisation with the retrieval step (Contractor et al., 2013). Their approach uses the sum of the keyword score and the similarity score to match the question part of each FAQ document in the collection to an SMS query. The keyword score for the question part of each FAQ document in the collection was obtained by first removing the vowels and stopwords for each question part. (Shivhre, 2013) reasoned that they removed the vowels after observing that, in general, users compress the text by omitting some vowels from the text. The vowels and stopwords were also removed in the SMS queries. The keyword score for each question part in the FAQ document collection (corpus) was calculated as the ratio of the words (tokens) of the SMS queries it contains. In essence, a question part in the collection that has no vowels and stopwords but having all the SMS query terms/words will have a keyword score of 1.

Moreover, for each word in the SMS query, Shivhre (2013) used a combination of the Longest Common Sequence Ration (LCSR), the similarity ratio using Ratcliff/Obershelp algorithm, the levenshtein distance and the inverse document frequency to assign weights to the words in each question part in an FAQ document in the collection. The similarity score between the SMS query and each question part in an FAQ document in the collection was then calculated using these weights (Shivhre, 2013). The question part in the collec-

tion which had words that were the best possible matches for the words in the query got the highest similarity score. The final total score for each question part in an FAQ document in the collection was then calculated by summing the keyword score and the similarity score. These values were then scaled between 0 and 1 for the top 5 retrieved documents.

In order to detect the missing content queries, Shivhre (2013) used the total sum of the scores of the top 5 retrieved FAQ documents as a minimum threshold to decide whether queries are MCQs or non-MCQs. If all the matching question parts of the top 5 retrieved FAQ documents had a total score below that threshold, the SMS query was considered a MCQ. The approach proposed by Shivhre (2013) yielded fairly reasonable results as they were able to accurately detect 54% non-MCQs and 72.5% of the MCQs. Shivhre (2013) reported a mean reciprocal rank of 0.86, which was second highest compared to the other participants (Contractor et al., 2013, Shivhre, 2013).

Another system that performed fairly well at the FIRE2011 SMS-Based FAQ retrieval task was the system by Gupta (2013). In the normalisation phase, Gupta (2013) deployed an approximate string-matching algorithm. This algorithm converted all the words in the question part of each FAQ document to their metaphonic equivalent using a metaphone library. These new representations of the question part of each FAQ document was then indexed in Lucene [16]. During the indexing process, only the new question part representation was indexed without the answer part. They only indexed the question part because their preliminary investigations indicated a reduction in the retrieval performance when the answer part was included in the index (Gupta, 2013). The SMS queries were also transformed to their phonetic equivalent using a metaphone library so that they can have the same representation as the question part that have been indexed in Lucene. In the retrieval phase, Gupta (2013) used this new query representation to query the Lucene index. The traditional TF-IDF (Robertson, 2004) was used to score and rank the FAQ documents (question part only) in the index. The top 5 documents were retrieved.

In addition, Gupta (2013) devised a heuristic where a threshold was set to determine whether to mark an SMS query as a MCQ or a non-MCQ. This was achieved by determining if all the retrieved documents had a score less than the *No. of Tokens* * *C* (tokens in the SMS query). The value of C was obtained experimentally and was set at 1.15. Gupta (2013) reported a reasonably high *MRR* of 0.744 using the traditional TF-IDF. The simple heuristic that was devised for identifying MCQs and the non-MCQs also performed fairly well as they reported that 56.5% MCQs and 59.3% non-MCQs were accurately detected (Gupta, 2013). Overall, the system by Gupta (2013) was ranked third amongst all the teams that participated at the FIRE2011 SMS-Based English mono-lingual FAQ retrieval task (Contractor et al., 2013).

In another approach, Shaikh et al. (2013) used a domain dictionary (terms in the FAQ doc-

[16] https://lucene.apache.org/

ument collection) and a synonym dictionary to identify a candidate set of terms that closely match each SMS token using a similarity measure. They calculated this similarity measure using an approach earlier introduced by Kothari et al. (2009) and this approach is described in detail in Section 2.3.7. In addition to this measure, they included other measures, the proximity score and the length score in calculating the score for the question part of each FAQ document. They used the proximity score in order to improve the overall score of each FAQ document in the collection if the question part had two consecutive terms as the SMS query. On the other hand, they used the length score to penalise the FAQ documents which had the question part that differed in length considerably compared to the SMS query. They reported a reasonably high MRR of 0.9041, which was slightly higher than the current state-of-the-art approach when all the scoring functions are used together. The results they reported in this work were far much better than what they had submitted at the FIRE2011 SMS-Based FAQ retrieval task (Contractor et al., 2013). For the FIRE2011 submission, they only used the similarity score without the proximity score and the length score.

Shaikh et al. (2013) also set a threshold value for the scores of the question part in order to determine if a query is a MCQ or a non-MCQ. In their work, they did not describe how they determined this threshold. However they reported a reasonably high detection accuracy of 74.0% for the non-MCQs and 84.7% for the MCQs (Shaikh et al., 2013).

Leveling (2012) viewed the detection of the missing content queries in an SMS-Based FAQ retrieval setting as a classification problem. In their approach, they trained an IB1 classifier as implemented in TiMBL (Daelemans et al., 2002) using numeric features generated during the retrieval phase on the training data (FIRE2011 SMS-Based FAQ retrieval monolingual English data) to identify the MCQs and non-MCQs. The features used for training were comprised of the result set size for each query, the raw BM25 (Robertson et al., 1996) document scores for the top five documents (5 features), the percentage difference of the BM25 document scores between the consecutive top 5 documents (4 features), normalised BM25 document scores for the top five retrieved documents (5 features) and the term overlap scores for the SMS query and the top 5 retrieved documents (5 features). Their approach essentially yielded a binary classifier that can determine whether a query is a MCQ or a non-MCQ. This approach is much simpler compared to the approach proposed by Hogan et al. (2011) because it relies on a single classifier instead of relying of several classifiers. Leveling (2012) evaluated this approach using a leave-one-out validation approach which is supported by TiMBL and reported a classification accuracy of 86.3% for MCQs with the best performing system. Such a high classification accuracy for MCQs resulted in a very low classification accuracy of 56.0% for non-MCQs.

2.3.6 Summary

Unlike in Desktop-Based FAQ retrieval systems, current approaches in SMS-Based FAQ retrieval systems place much emphasise on SMS normalisation and the detection of missing contents queries (MCQs). The detection of MCQs has become a central issue in SMS-Based FAQ retrieval research because in a realistic scenario, users of an FAQ retrieval system are not aware of the contents in the information source and are likely to submit queries that do not have relevant FAQ documents in the collection (FAQ collection deficiency problem). The introduction of the MCQs detection system in an SMS-Based FAQ retrieval setting can be used to eliminate unnecessary interaction with the systems for the SMS queries that do not have the relevant FAQ documents in the collection. If the MCQs are correctly identified, this can improve the overall efficiency (overall shorter search length) of the system.

Two approaches have been proposed in the literature for detecting MCQs and non-MCQ. These are: binary classification and setting a threshold value on the score of the retrieved FAQ documents. However, in the state-of-the-art binary classification approach, it is not yet clear which combination of features can build a model that yields the best classification. In this thesis, we will carry out a thorough empirical evaluation in order to determine the combination of features to use in our binary classifier for detecting MCQs and non-MCQ. Another issue that most systems did not address in the FIRE2011 SMS-Based FAQ retrieval task is the term mismatch problem between the SMS query and the relevant FAQ document in the collection. This thesis will devise novel techniques for bridging this term mismatch problem in order to improve the overall retrieval performance in an SMS-Based FAQ retrieval setting.

2.3.7 SMS-Based FAQ Retrieval Approaches that do not Address the Missing Content Queries Detection Problem

In the previous section, we reviewed several approaches proposed in the literature for addressing both the Missing Content Queries (MCQs) detection and the SMS normalisation problem in SMS-Based FAQ retrieval. This section presents a review on several other SMS-Based FAQ retrieval approaches that do not address the MCQs detection problem. In particular, Adesina and Nyongesa (2013) proposed an algorithm (SMS*ql* algorithm) for ranking the FAQ documents given a normalised SMS query in an HIV/AIDS FAQ retrieval system. Their algorithm considered the similarity in the words between the keyword phrases extracted from the SMS query and each FAQ document in the collection, the length of the SMS query and the question part of each FAQ document and the order in which the words were placed to rank the FAQ documents. These keyword phrases were learnt from a query log made up of 2000 FAQ SMS query formats. In total, 205 keywords were learnt from this query log. Adesina and Nyongesa (2013) compared their algorithm with a simple naive al-

gorithm whereby the query terms are traversed to count the frequency of occurrences of each word in an FAQ document in the collection. In their comparison, there was no significant difference in the retrieval performance in terms of precision and recall.

A major criticism of Adesina and Nyongesa (2013) work is that the participants were shown the FAQ documents in the collection to build the query log. Therefore, it is highly likely that they used the same keywords or phrases used in the FAQ documents to generate the shortened SMS queries. Since their algorithm uses keyword matching, it will be difficult to evaluate how such a system may perform in a real life setting. In a more realistic scenario, users of an FAQ retrieval system generate queries using their own terms and phrases. In our study, we will simulate a real life setting by not showing the participants the FAQ documents when creating the query log. Participants will be asked to provide several SMS queries on the general topic of HIV/AIDS.

In another major study, Kothari et al. (2009) proposed an approach that handles noise in an SMS query by determining the SMS query similarity over the question part of each FAQ document in the collection as a combinatorial search problem. In their approach, for each and every SMS query term, they used a weighting function to create a ranked list of all possible clean English tokens/terms from the domain dictionary. They used an aligned synonym dictionary to handle semantic variations between the query terms and terms in the domain dictionary. In order to retrieve the best matching question that corresponds to the SMS query, they used two different algorithms namely the pruning algorithm and the naive algorithm. The two algorithms had similar function but differed in run time efficiency.

The naive algorithm queries the index for each and every term appearing in the ranked list and the returned questions are added to a collection. A maximum scoring question is then selected from the collection using a scoring function. On the other hand, the pruning algorithm queries fewer terms as compared to the naive algorithm. It iterates through the lists and at each iteration, it picks terms with higher weights to be used for selecting the best questions to put in the collection. A threshold is set by upper bounding the score achievable by a possible question that matches the query in order to stop the iteration process. If at any iteration the threshold is satisfied, the iteration process stops since the collection is guaranteed to contain the maximum scoring question. A maximum scoring question is then selected from this collection.

In their evaluation, Kothari et al. (2009) reported that the pruning algorithm outperformed the naive algorithm in correctly answering the SMS queries. The pruning algorithm also gave a near constant runtime performance on all queries since it queries much less number of terms and ends up with a smaller candidate set compared to the Naive algorithm. Their results also suggest that using a synonym dictionary could improve the quality of the correctly cleaned SMS queries. This was achieved by cleaning the SMS queries with the pruning

algorithm only and then comparing the results with cleaning the SMS queries using the pruning algorithm and the synonym dictionary. The latter approach returned a large number of correctly cleaned SMS queries as compared to the former. In order to test the performance of their system, they compared their results to the results obtained using the Lucene in-built fuzzy match, as a benchmark. Their evaluation results suggest that their approach outperformed the Lucene in built fuzzy match. Although their approach suggests promising results, it did not address the detection of spacing errors in the SMS token. If spacing errors could be detected before the ranked list is created, this could improve the quality of results as demonstrated by Byun et al. (2008).

Byun et al. (2008) on the other hand proposes a two phase model for SMS text refinement. In the first phase, they used a Hidden Markov Model (HMM) approach proposed by Lee et al. (2007). In their approach, they correct spacing errors by using a HMM-based spacing model trained from partially revised SMS messages where all spacing errors are manually corrected but spelling errors still remain. They differed slightly to the approach used by Lee et al. (2007) where the training data had no spelling errors and consisted of only clean text. They argued that, using training text with spelling errors could enable them to accurately correct some errors with the noisy context on the first phase and then the remaining errors could be corrected by the second phase. They gave an example with the English SMS message 'lemme c' that corresponds to the sentence 'let me see'. This SMS has two types of errors, spacing error and spelling errors. The spacing errors ('lemme' –> 'lem me') could be accurately corrected by the first phase and the spelling errors ('lem me c') corrected by the second phase.

On the second phase, they used a rule based correction model to correct spelling errors. These correction rules were automatically extracted from pairs of partially revised SMS and its spelling refined reference. The main disadvantage of this two phase model is that it relies heavily on the large corpora of partially revised SMS messages for spacing error correction and a fully revised reference SMS messages corpus for spelling error correction. One other disadvantage of the second phase is that it depends heavily on manually intensive error correction rules to make spelling correction. With the texting language changing almost every day, this is not ideal because these correction rules will require constant updating. According to their results, their two phase model performed much better than the baseline method proposed by Aw et al. (2006) which combined both spelling and spacing error correction. In their approach, Aw et al. (2006), considers the SMS refinement problem as a translation problem from the SMS texting language to normal language and they built a phrase-based statistical translation model from an aligned corpus consisting of raw messages and manually revised ones.

In a study which set out to normalise the SMS queries, Contractor et al. (2010) proposed a two step process for translating the noisy SMS text to clean text using a Statistical Machine

Translation (SMT) approach. In the first step, they tokenise each SMS sentence to generate a ranked list of possible clean English tokens together with their scores. These scores corresponds to translation probabilities of the pseudo translation model, which is based on the Model 1 of the IBM translation model (Brown et al., 1993).

In the second step, Contractor et al. (2010) created a tri-gram language model from a collection of 100000 clean text documents. They used Moses[17], an open Source decoder for SMT, to obtain clean English sentences from the pseudo translation model and the tri-gram language model. To evaluate their system, they used Bilingual Evaluation Understudy (BLEU) and Word Error Rate (WER). The BLEU scores were used to measure the similarity between the human reference text and the sentence generated by their method. According to their results, their approach yielded fairly high BLEU scores on cleaned text as compared to unprocessed noisy text. This suggests that the sentences generated are nearly similar to the human generated text as compared to the noisy text. Also their comparison on WER suggests that their approach had 10% lower WER as compared to unprocessed text. This as well demonstrates that the generated words were nearly similar to the human generated words as compared to the noisy text that had a higher WER. Their WER results also suggest that 75% of clean sentences had correct words present as compared to 60% of the noisy text.

2.3.8 Summary

In Section 2.3.7, we conducted a comprehensive review on several other SMS-Based FAQ retrieval approaches that do not address the missing content query detection problem. All the proposed approaches reported promising results but because of lack of comparative data, it is difficult to compare the performance of the proposed approaches. One important thing to note is that even if the SMS query can be correctly normalised, there is no guarantee that the normalised SMS query will be able to accurately retrieve the relevant FAQ documents in the collection. This may be because of the term mismatch problem between the normalised SMS query terms and the relevant FAQ documents. The other reason may be that the relevant FAQ document for a particular user query does not exist in the FAQ document collection. These are some of the FAQ document collection deficiencies identified by Sneiders (1999, 2009) that we will address in this thesis in order to improve the satisfaction of users of our FAQ retrieval system.

[17] http://www.statmt.org/moses

2.4 Conclusion

In this chapter, we have conducted a literature review on several approaches for addressing the term mismatch problem between the user query and the relevant FAQ document in the collection, the detection of missing content queries and the SMS normalisation problem when developing both SMS-Based FAQ retrieval systems and Desktop-Based FAQ retrieval systems. Finally for each approach, we have outlined its limitation. In particular:

- In Section 2.2, we discussed several approaches for developing Desktop-Based FAQ retrieval systems with a special focus on addressing the term mismatch problem. We introduced the template-based approach and discussed the few works that have leveraged information from previous searches to address the term mismatch problem between the user query and the relevant FAQ document in the collection. However, there has been no work examining whether term frequencies from a query log can be leveraged to better address this term mismatch problem.

- In Section 2.3, we introduced several approaches for developing SMS-Based FAQ retrieval systems with a special focus on addressing the missing content queries detection problem and the SMS normalisation problem. We described how prior works used binary classification to identify these missing content queries without carrying out an empirical evaluation to determine the best combination of features to use for building such a classifier. In contrast, in this thesis, we will conduct such an empirical evaluation in order to determine the set of features that can build a model that yields the best classification performance.

Chapter 3

FAQ Retrieval Evaluation Dataset

3.1 Introduction

In Information Retrieval (IR), a system can be evaluated in two different ways in order to assess how well it meets the information needs of its users. These are user-based evaluation and system evaluation (Voorhees, 2002). System evaluation measures how well the system can rank the retrieved documents. User-based evaluation, also known as interactive information retrieval evaluation, measures the extent to which the user is satisfied with the system. This often involves studying users' behaviours and experiences and the interactions that occur between the users and the systems and the users and information (Kelly, 2009). One of the main disadvantages of using a user-based evaluation is that it requires a representative sample of the actual users of the system. This sample of users might have to evaluate the system over a long period of time (Voorhees, 2002, Sparck Jones and Willett, 1997). Users may also require expensive and time-consuming training (Sparck Jones and Willett, 1997).

Because of the complexity and the expensive nature of user-based evaluation, the IR research community has often adopted the less expensive system evaluation methodology referred to as the Cranfield paradigm (Harman, 2010, Sanderson, 2010, Voorhees, 2002). Experiments conducted in this way require a resource known as a test collection and an evaluation measure (Sanderson, 2010). Test collections are re-usable and standardised resources that can be used to measure the retrieval effectiveness of an information retrieval system (Clough and Sanderson, 2013). Test collections are used in evaluation conferences such as the Text REtrieval Conference (TREC)[18], the Conference and Labs of the Evaluation Forum (CLEF)[19], the FIRE[15], and the NII Testbeds and Community for Information access Re-

[18]http://trec.nist.gov/
[19]http://www.clef-initiative.eu/

search (NTCIR)[20]. The main components of an information retrieval test collection are the document collection, topics and the relevance assessments. The following is a description of each component:

- A static set of documents (document collection) to be searched. Each document in the collection has a unique Document Identifier (docid). In this thesis, we use the FAQ document collection described in Section 1.2 as a static set of documents to be searched in the test collection.
- A set of information needs (also know as topics/queries). Each topic/query has an identifier. Topics are often structured to provide a detailed statement of the information need behind the query. The main components that make up a topic are the following: a topic id; a short title that could be viewed as a query; a description of the information need written in no more than one sentence and a narrative to provide a complete description of what documents the searcher would consider as relevant. In this thesis, our information needs will be expressed as SMS queries, which are expressed as natural language questions. In Section 3.2, we describe in detail how these SMS queries were collected.
- A set of known documents for each of the information needs. This is also known as query relevance judgements (qrel). This is created by linking each query identifier to a set of docids corresponding to the relevant documents in the collection. In this thesis, we use the query relevance judgements created as described in Section 3.2

With an appropriate test collection, and a chosen evaluation measure, an IR researcher can assess and compare the effectiveness of different retrieval strategies when deployed in an IR system (Voorhees, 2002). Evaluating an information retrieval system in this manner involves loading the documents in the test collection into a retrieval system using a suitable format for searching and retrieval. This process is referred to as indexing (Van Rijsbergen, 1979). After the documents have been indexed, the queries (in the test collection) are submitted to the system to retrieve the documents that the system has identified as relevant to the query. The list of documents retrieved for each query is examined to determine the documents that are relevant and those that are not relevant to the query based on the query relevance judgements (Sanderson, 2010). A suitable evaluation measure is then used to quantify the retrieval effectiveness of the system. The test collection, together with an evaluation measure simulate users of a search system in an operational setting and enable the effectiveness of an information retrieval system to be quantified (Clough and Sanderson, 2013). The remainder of this chapter is organised as follows:

[20]http://research.nii.ac.jp/ntcir/index-en.html

- In Section 3.2, we describe how we collected the queries and the query relevance judgements to be used in evaluating our FAQ retrieval system.
- In Section 3.3, we describe how we created 10 different test collections to be used in evaluating all the retrieval approaches deployed in our FAQ retrieval system. We also describe in detail several evaluation measures that we will use along with the widely accepted Cranfield paradigm evaluation methodology.
- In Section 3.4, we assess the quality of these query relevance judgements we created earlier in Section 3.2 by asking another group of participants who were recruited through crowdsourcing to provide additional query relevance judgements for our query log.

3.2 Collecting Users HIV/AIDS Queries and Building a Query Relevance File

We conducted a study in Botswana from 1^{st} September 2011 to 25^{th} March 2012 to collect SMS queries on the general topic of HIV/AIDS. This study was granted the University of Glasgow ethics approval and was allocated the ethics project reference number: CSE00840. The main aims and objectives of this study was to create a test collection that could be used for training and evaluating the FAQ retrieval system. In this study, 85 participants were recruited to provide SMS queries on the general topic of HIV/AIDS. The participants were recruited randomly across the city of Gaborone. Since this application is being developed for users of low-end mobile phones, only a subset of the population of users that use low-end mobile phones were recruited to take part in this study. The participants were not shown the FAQ documents in the information source for the FAQ retrieval system. A description of this information source is provided in Chapter 1 (Section 1.2). The rationale for not showing them these FAQ documents was to enable us to determine if the HIV/AIDS FAQ document collection to be used in this study suffers from the FAQ document collection deficiency problems described by Sneiders (1999). In particular, our aims were:

- To determine if the current information source covers a majority of the information needs of potential users of the FAQ retrieval system.
- To capture a wide variety of words and phrases for each of the user's information needs. Previous work has shown that people who share the same interest tend to ask the same questions over and over again (Sneiders, 2009). Our intuition is that different people will ask the same questions, but do so differently. Hence, this would give us different words and phrases for each information need. As per our thesis statement

in Chapter 1, these words and phrases could be used to enrich the FAQ documents in order to reduce the term mismatch problem between the users' queries and the relevant FAQ documents in the FAQ document collection.

- To build a query relevance file for use in evaluating the FAQ retrieval system.

3.2.1 Task 1: Collecting Training Data

In the first task, participants were asked to provide SMS queries on the general topic of HIV/AIDS using their mobile phones. The participants were given examples of topics for which they can derive their SMS queries. The participants were advised to type the SMS queries in a way they believe the FAQ retrieval system would be able to accurately retrieve the relevant question-answer pair for each SMS query. All the SMS queries provided by the participants were stored in a MySQL database and they were also automatically written to a Word document in order to enable us to be able to quickly identify and correct spelling errors. Participants were given 20 minutes to provide at least 10 SMS queries. In Appendix A, we provide a selection of SMS queries provided by participants. Immediately after completing task 1, the participants were given another task in order to build a query relevance file.

3.2.2 Task 2: Building a Query Relevance File

For the second task, participants were provided with a web-based FAQ retrieval system for HIV/AIDS to use for building a query relevance file for the queries they provided earlier in the first task (Task 1). The web-based FAQ retrieval system for HIV/AIDS used the FAQ document collection described in Section 1.2 as an information source. Participants were asked to search the web-based FAQ retrieval system for the relevant FAQ documents using their own queries, which they provided earlier in task 1. The participants were advised to use a spell checker to correct any spelling errors before submitting any queries to the web application. The web-based FAQ retrieval system for HIV/AIDS used the BM25 term weighting scheme to retrieve 20 FAQ documents for each query submitted. The participants were asked to assess the retrieved FAQ documents using the following topical relevance types (Huang and Soergel, 2004):

- Direct relevance - The retrieved FAQ document directly answers the query.
- Indirect or circumstantial relevance - The FAQ document indirectly answers the query.
- Context relevance - The retrieved FAQ document provides background information and sheds additional light on the query.

Using the aforementioned topical relevance types, participants were asked to mark the retrieved FAQ documents that answered their queries with a lot of detail as relevant. They were asked to mark those that answered their queries with less detail as slightly relevant and those that did not answer their queries at all as irrelevant. Figure 3.1 presents the web-based FAQ retrieval system for HIV/AIDS used for gathering the query relevance assessments.

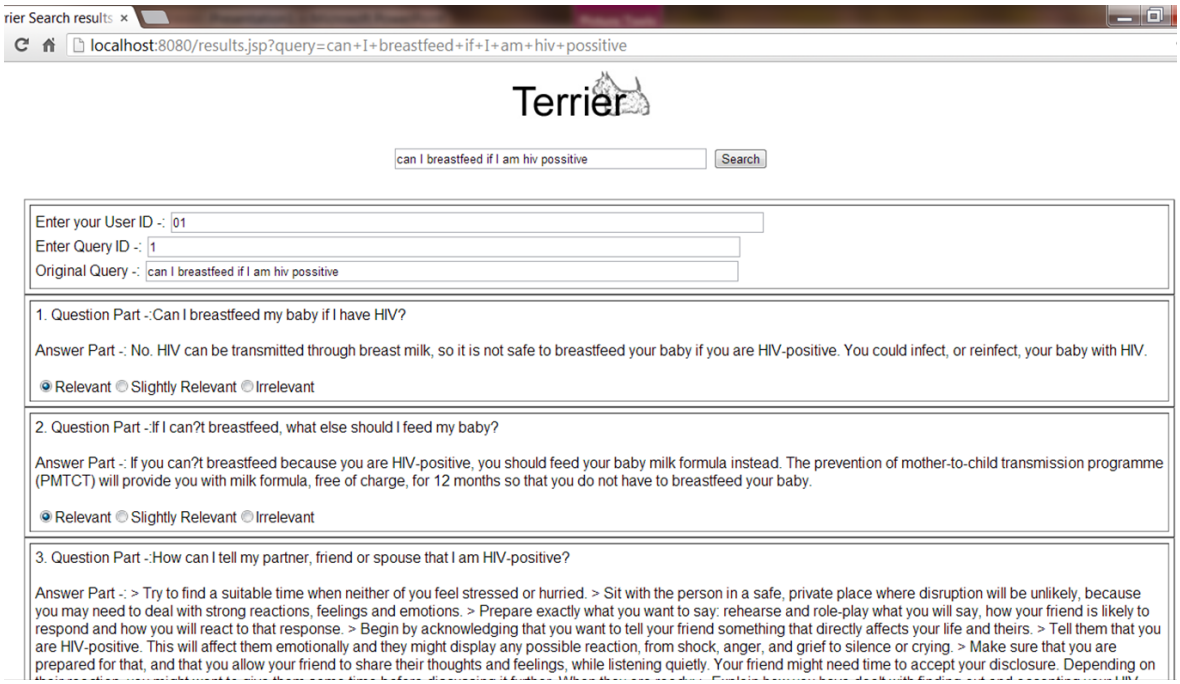


Figure 3.1: The web-based FAQ retrieval system for HIV/AIDS used for collecting query relevance information.

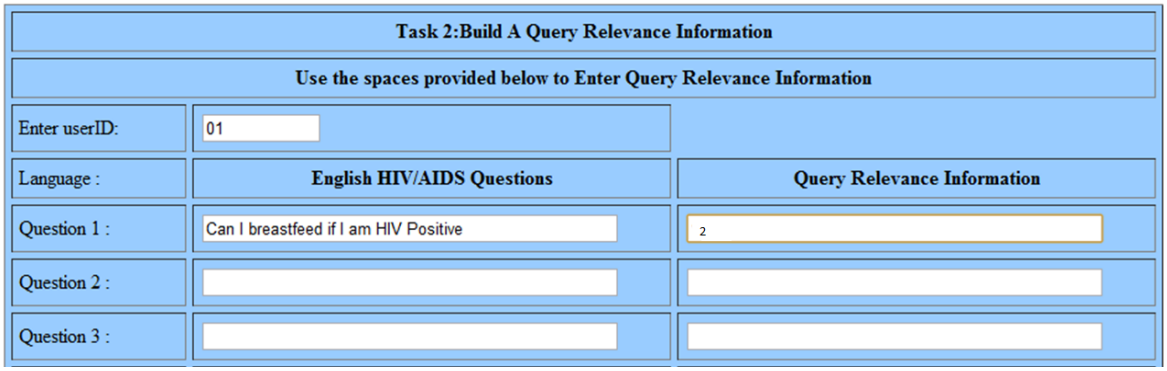


Figure 3.2: The web-based interface for collecting additional query relevance information directly from the printed version of the HIV/AIDS question answer booklets

In order to ensure that all the relevant FAQ documents for a user query have been found (completeness) (Liu, 2009), participants were also asked to browse a chapter in the printed version of the HIV/AIDS question-answer booklet which they believed might contain the

relevant FAQ documents. For example, if a participant asked a question related to ARV therapy, they were asked to browse the whole of Chapter 8 (Introduction to ARV therapy) of the printed version of the HIV/AIDS question answer booklet. Participants were asked to record these additional query relevance judgements in another web-based application as shown in Figure 3.2 for all the relevant and slightly relevant FAQ documents. Participants were given 40 minutes to complete this task.

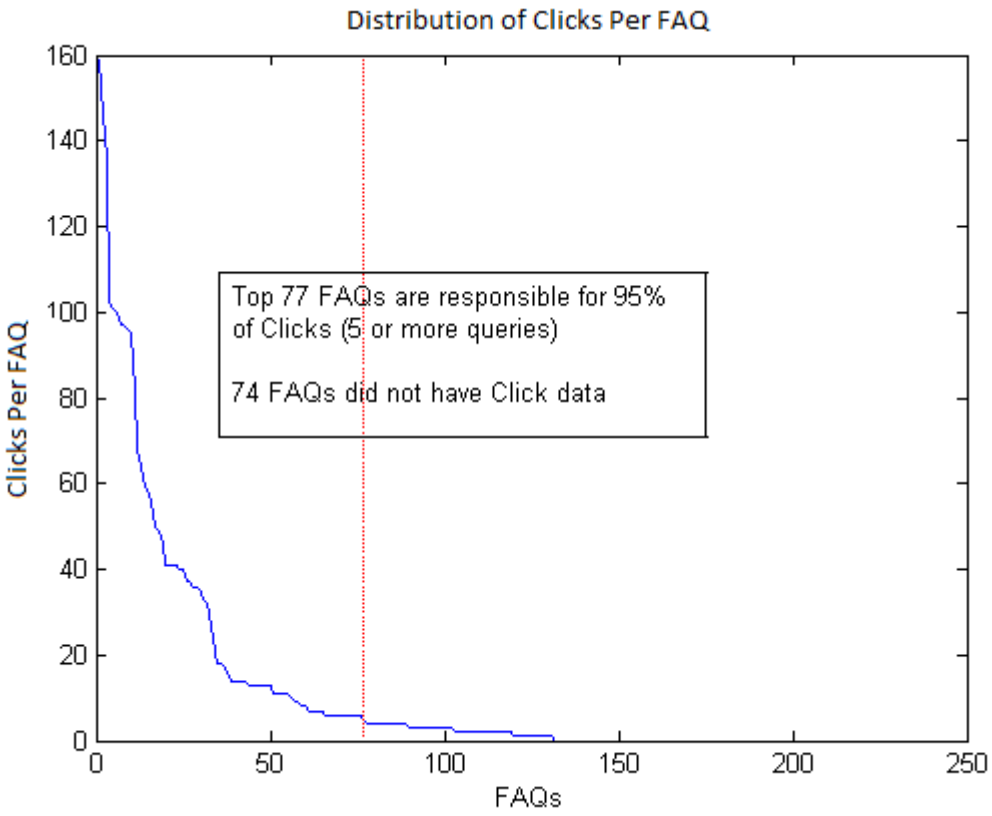


Figure 3.3: Distribution of Clicks Per FAQ

3.2.3 Query Log Analysis

In Figure 3.3 and Table 3.1 we present a summary of the data collected. In Figure 3.3, the y-axis represents the number of clicks for each FAQ document. As per our previous suggestion, we observed that people tend to ask the same questions over and over again. This was most evident on the topic related to the prevention and transmission of HIV and AIDS. The query log analysis suggests that there were 74 FAQ documents in the FAQ document collection that had no SMS queries associated with them from the participants as shown in Table 3.1 and Figure 3.3. Only 131 FAQ documents in the FAQ document collection were associated with 750 SMS queries (non missing content queries *non-MCQs*) from the participants. Also 207

SMS queries (missing content queries, *MCQs*) from the participants could not be associated with any FAQ document. These *MCQs* were on the general topic of HIV/AIDS (on-topic). These findings suggest that the FAQ document collection used in this study suffers from the same FAQ document collection problems described by Sneiders (2009), as described in Chapter 1. This thesis will attempt to improve the probability that users of our FAQ retrieval system are satisfied by addressing these FAQ document collection deficiency problems.

Table 3.1: Query Log Statistics showing the total number of Missing Content Queries (*MCQs*) and Non-Missing Content Queries (*non-MCQs*) collected from potential users of the system

	Number
Number of collected SMS queries	957
Number of *non-MCQs*	750
Number of *MCQs*	207
Number of FAQs that matches the *non-MCQs*	131
Number of FAQs that did not match any *non-MCQs*	74

Table 3.2: Query click-through data analysis. A click signifies that an FAQ document was identified as either relevant or slightly for a given query.

Chapter	Number of FAQs	Clicks/Chapter	AvClicks/FAQ	% Rel	% Slightly Rel
Men and HIV and AIDS	5	3	0.60	100	0
Nutrition,Vitamins and HIV/AIDS	8	49	6.13	73.4	26.6
Routine HIV Test	9	38	4.22	94.7	4.3
Understanding Tuberculosis	11	13	1.18	100	0
Taking the Test	12	132	11	61.4	38.6
What if you are HIV+	13	109	8.39	79.8	21.2
Masa Programme	18	92	5.11	90.2	9.8
Understanding HIV and AIDS	23	940	40.87	45	55
Protecting Yourself	28	712	25.43	33	67
Women, Children and HIV	28	197	7.04	47.2	52.8
Introduction to ARV Therapy	50	507	10.14	38.1	62.9
Total	205	2792		46	54

An analysis of the query log indicates that participants were able to generate 2792 clicks for the 750 *non-MCQs* as shown in Table 3.2. A click signifies that an FAQ document was identified as either relevant or slightly relevant for a given query. We refer to these set of FAQ documents as clicks because the non-relevant FAQ documents were not marked by the participants. This is similar to the classical Web IR notion of clicks, where clicks can be seen as an implicit indicator that may indicate relevance (Silvestri, 2010). In this study, all the FAQ documents marked as relevant and slightly relevant were considered to be relevant to a given query. This was done to enable us to be able to aggregate the relevance judgements using a union of the assessments from several users. For example, for the following queries:

- How can one avoid contacting HIV/AIDS?

- how can someone prevent HIV/AIDS?
- How can HIV transmission be stopped?
- how can you protect yourself from getting infected by aids?

There were some inconsistencies in the relevance judgements as some participants took a more liberal view of relevance than the others. Some considered the following FAQ documents to be relevant to these queries while others considered them to be slightly relevant.

- How can you get infected with HIV? The main ways in which you can get infected with HIV are: By having oral or penetrative sex without a condom. blood-to-blood contact i.e. by sharing sharp objects like razor blades or needles with an infected person, or by coming into contact with an HIV-positive persons blood, through sores or cuts on your body. from an HIV-positive mother to her child, either in the womb, when giving birth or through breastfeeding. By way of blood transfusion. However, in Botswana, this risk is low as all blood donations are tested for HIV.
- I am scared of getting infected with HIV. Which bodily fluids could contain the virus? Amounts of HIV that are large enough to infect somebody can be present in blood and blood products, semen, vaginal fluids or breast milk. - Very small amounts of HIV may be found in saliva or spit (only in a very small number of people), blister fluid, or tears. However, as far as it is known, no one has been infected by coming into contact with tears or blister fluid. - HIV has not been found in urine, faeces, vomit or sweat.

Table 3.3: The number of relevant FAQ documents per query.

Number of Relevant FAQs	Number of *non-MCQs*	Total Number of Clicks
1	190	190
2	203	406
3	48	144
4	130	520
5	22	110
6	4	24
7	19	133
8	46	368
9	53	477
12	35	420
Total	750	2792

For example, there were 8 FAQ documents that were considered to be relevant or slightly relevant to the query "*How can AIDS be transmitted*". Consequently, the number of clicks exceeded the number of *non-MCQs* (see Table 3.3) because some *non-MCQs* were identified as relevant to more than one FAQ document in the FAQ document collection. Also, a thorough analysis of these clicks as shown in Table 3.2 indicates that 77.3% (2159 clicks) of the

relevance judgements provided (clicks) were for just three chapters (Understanding HIV and AIDS, Protecting Yourself, and Introduction to ARV therapy). The whole paradigm of the template-based approach as described earlier in Chapter 2 (Section 2.2.2) exploits this repetitive nature of user queries (Andrenucci and Sneiders, 2005, Sneiders, 2009). Templates are created using these frequently occurring queries in order to help resolve the term mismatch problem. In Chapter 5, we investigate whether we can improve the probability that users of our FAQ retrieval system are satisfied by deploying a template-based approach that uses a query log to enrich the FAQ documents in order to help resolve the term mismatch problem between the users' queries and the relevant FAQ documents.

3.3 Test Collection and Evaluation Measures for Evaluating the FAQ Retrieval System

As was pointed out earlier in Section 3.1, one of the main advantages of using the Cranfield evaluation methodology when developing an IR system is that it is easier to assess and compare the effectiveness of different retrieval strategies deployed in an IR system. This can be achieved by assessing the retrieval effectiveness of each retrieval strategy on a fixed test collection (fixed document collection, fixed test queries and qrel for the test queries) using a suitable IR evaluation measure. In Section 3.3.1, we describe how we created 10 different test collections to use in assessing the retrieval effectiveness of the different retrieval strategies that we are proposing to deploy in our FAQ retrieval system. In Section 3.3.2, we describe in detail several evaluation measures that we will use along with the widely accepted Cranfield paradigm evaluation methodology to asses the retrieval effectiveness of the proposed FAQ retrieval system.

3.3.1 Creating the Test Collection For Evaluating the FAQ Retrieval System

Recall from our thesis statement in Chapter 1 (Section 1.4) that we postulate that we can improve the probability that users of our FAQ retrieval system are satisfied by enriching the FAQ documents with additional terms from a query log, which are added as a separate field in a field based model in order to reduce the term mismatch problem between the users' queries and the relevant FAQ documents. In order for us to validate this hypothesis, we require a fixed set of FAQ documents for retrieval, a fixed set of training queries for enriching the FAQ documents and a fixed set of queries for testing our enrichment strategy. Since we already have a fixed set of FAQ documents for retrieval, we randomly split the 750 *non-MCQs* 10 times to create a training sets of 600 queries and a testing sets of 150 queries. In Figure 3.4,

we illustrate how we split the 750 *non-MCQs* into 10 training and testing set. In Chapter 5, we use these 10 different train/test splits to evaluate our proposed enrichment strategies. As per our thesis statement, we also use the these 10 different train/test splits in Chapter 6 to investigate whether we can improve the probability that users of our FAQ retrieval system are satisfied by ranking the FAQ documents according to how often they have been previously identified as relevant by users for a particular query term.

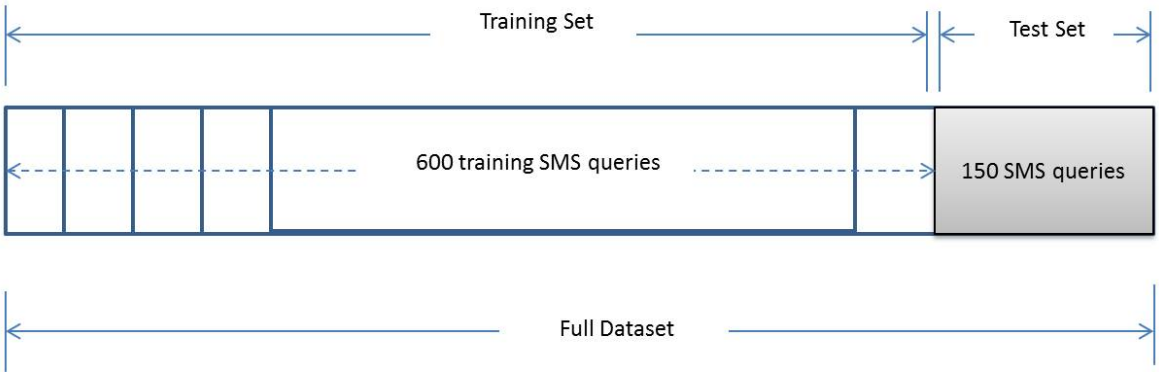


Figure 3.4: Splitting the *non-MCQs* into training and testing sets

3.3.2 Evaluation Measures

The first IR evaluation measures were defined by Kent et al. (1955) and Cleverdon and Kean (1968) to assess unordered set of retrieved documents matching a user's query in Boolean search. These evaluation measures were precision and recall. Precision measures the fraction of retrieved documents that are relevant as expressed in Equation (3.1). Recall measures the fraction of relevant documents retrieved as expressed in Equation (3.2).

$$precision = \frac{R_r}{R_r + RN_r}, \tag{3.1}$$

$$recall = \frac{R_r}{R_r + NR_r}, \tag{3.2}$$

where R_r represents the number of documents retrieved that are relevant to the user query, RN_r represents the number of documents retrieved that are not relevant to the user query and NR_r represents the number of documents that are not retrieved but are relevant to the user query. In Chapter 5, we use recall to evaluate whether enriching the FAQ documents with terms from our query log can alleviate the term mismatch problem between the users' queries and the relevant FAQ documents. We also use recall in Chapter 4 together with other evaluation measures to help us select a suitable baseline term weighting model to use in the remainder of this thesis.

In ranked retrieval however, three commonly used evaluation measures in the IR community are the: Mean Average Precision (MAP), Precision at a fixed ranking, and the Mean Reciprocal Rank (MRR) (Sanderson, 2010). The MAP is derived from the mean of the average precision for a set of queries (Voorhees, 1994a). The average precision for a given query is defined as:

$$AP = \frac{\sum_{rn=1}^{N_r}(P(rn) * rel(rn))}{R_Q}, \tag{3.3}$$

where N_r is the number of documents retrieved for a given query, rn is the rank number for the retrieved documents, $rel(rn)$ can either be 1 or 0 depending on the relevance of the document at rank rn. $P(rn)$ is the precision measured at rank rn and R_Q is the total number of relevant documents for the given query. Hence, assuming there are N_Q queries, the mean average precision is given as:

$$MAP = \frac{\sum_{qn=1}^{N_Q} AP(qn)}{N_Q}, \tag{3.4}$$

where qn is the query number. In this thesis, we also use this evaluation measure in subsequent chapters to evaluate the retrieval effectiveness of our FAQ retrieval system.

Another important evaluation measure is Precision measured at a fixed rank $P(rn)$. This evaluation measure is used when it is assumed that a user will only examine a fixed number of retrieved documents for a given query (Salton, 1968, Van Rijsbergen, 1979). This measure is expressed as:

$$P(rn) = \frac{R_r(rn)}{rn}, \tag{3.5}$$

where rn is the rank at which precision is measured and $R_r(rn)$ is the number of relevant documents retrieved in the top rn. This evaluation measure does not take into account the rank position of the relevant documents retrieved above the rank rn and all the relevant documents ranked below rank rn are ignored. In this thesis, we only use this evaluation measure in Chapter 4 to help us select a suitable baseline term weighting model to use in the remainder of this thesis.

Finally, Mean Reciprocal Rank was defined by Kantor and Voorhees (2000) to assess retrieval systems that have one relevant document in the collection being searched. MRR measures the average value of the reciprocal ranks of the first relevant documents given by each query as shown in Equation (3.6).

$$MRR = \frac{1}{N_Q} \times \sum_{i=1}^{N_Q} \frac{1}{r_i}, \tag{3.6}$$

where r_i is the rank of the first relevant document retrieved by the i_{th} query and N_Q is the number of queries.

We have seen in Table 3.3 that several *non-MCQs* have more than one relevant FAQ document in the collection. Therefore, the Mean Reciprocal Rank (MRR) and the Mean Average Precision (MAP) are both necessary in our evaluation because they provide different information. For example, the MRR will give us an insight into how quickly users are likely find a relevant FAQ document. The MAP on the other hand will only give us an insight into how good the FAQ retrieval system is in retrieving the relevant FAQ documents in the top k retrieved documents.

3.4 Crowdsourcing to Evaluate the Quality of the Query Relevance Judgements

In Section 3.2.2, we ensured that our query relevance judgements were complete by asking the participants to browse a chapter in the printed version of the HIV/AIDS question-answer booklet, which they believed might contain the relevant FAQ documents. We also ensured that the query relevance judgements were consistent by considering all the FAQ documents marked slightly relevant as relevant. In this section we assess the quality of these query relevance judgements by asking another group of participants who we recruited through crowdsourcing to provide additional query relevance judgements for our query log. Crowdsourcing refers to the process of outsourcing tasks or human intelligence tasks (HITs) to an online community in the form of an open call, often in exchange for micro-payments, social recognition, or entertainment value (Kazai, 2011, Whitla, 2009). Our aim is to measure the agreement between the query relevance judgements provided by a group of participants recruited in Botswana and those recruited through crowdsourcing.

3.4.1 Literature Review on Gathering Query Relevance Assessments through Croudsourcing

Crowdsourcing services such as CrowdFlower[21], Amazon Mechanical Turk[22] and CloudCrowd[23] are increasingly looked upon as a feasible alternative to traditional methods of gathering query relevance judgements for evaluation of search engines (Kazai, 2011). The availability of these crowdsourcing services makes it possible for anyone to create and publish HITs, and gather vast quantities of data from a large population of workers within a short time and at a relatively low cost (Kazai, 2011, Vuurens and de Vries, 2012). However, the quality of the relevance judgements obtained through crowdsourcing raises a range

[21] http://crowdflower.com/
[22] http://www.mturk.com/
[23] http://www.cloudcrowd.com/

of questions, because it uses workers of unknown quality with possible spammers among them (Kazai, 2011, Vuurens and de Vries, 2012).

Some authors (e.g. Alonso and Mizzaro (2012)) have demonstrated that the quality of the query relevance judgements can be improved by using quantification tests. Quantification tests are a set of questions that the participant/worker must answer to qualify to work on the published HIT. Later in section 3.4.3, we describe how we use these quantification tests to identify random/malicious assessments. Sorokin and Forsyth (2008) on the other hand injected gold standard data on the task in order to encourage the participants to follow the task instructions. If a participant's response deviated significantly from the gold standard, the standard would be shown to help the participant learn what was required. One study by Alonso and Baeza-Yates (2011) explored the design and execution of relevance judgements using Amazon Mechanical Turk as a crowdsourcing platform. In this study, they reported that the bulk of the experimental design should be on the user interface and instructions. In particular, they argued that a badly designed user interface can have effects on relevance and readability, hence making crowdsourcing task difficult.

3.4.2 Collecting Query Relevance Judgements through Crowdsourcing

In this crowdsourcing task, we collected additional query relevance judgements for our query log of 957 SMS queries from an online community of users through CrowdFlower[24]. For each query, we created a pool of FAQ documents to be assessed. The FAQ documents in this pool contained all those that were identified as relevant and slightly relevant by participants in an earlier study in Botswana. To increase the pool size for the relevant FAQ documents, we also generated other FAQ documents to be added to this pool by ranking the top 20 FAQ documents for each query in our query log using the BM25 term weighting model. These different groups of FAQ documents were merged to provide a final ranking of up to 20 FAQ documents to be judged for each query. It is worth pointing out that not all of the queries retrieved 20 FAQ documents. Therefore, some queries had fewer than 20 FAQ documents to be judged. This resulted in 19029 judged FAQ documents for the 957 queries. Figure 3.5 shows the user interface used to collect the query relevance judgements for each query. As shown in this figure, the participants were asked to identify each FAQ document as either relevant, slightly relevant or irrelevant. We collected three different judgements for each query-documents pair. Participants were paid $0.01 after providing judgements for 20 query documents pairs. They were given up to 10 minutes to provide query relevance judgements for 20 query document pairs.

Figure 3.6 shows the answer distributions for our crowdsourced task. There was a 90.33% agreement in the query relevance judgements provided. This high agreement indicates that

[24]http://crowdflower.com/

different participants frequently gave the same response to the same query-document pair. As suggested in Section 3.2.3, we created a final query relevance judgements using this crowdsourced data by considering all the FAQ documents identified as slightly relevant as relevant. A total of 117 participants contributed in this task. Initially, our setting allowed participants to provide an unlimited number of query relevance judgements. We subsequently capped the maximum number of contributions per participant to 2000 and then to 1000 so that we can get as many query relevance judgements as we can from a large number of participants as shown in Figure 3.7. This figure also shows that the participant who provided more judgements was more trustworthy than other participants. This participant recorded a trust score of 94% after providing 5404 judgements. As described later in Section 3.4.3, this trust score is based on the number of correctly answered test questions, which were introduced before participants could be allowed to participate in the task, and also in the middle of the task.

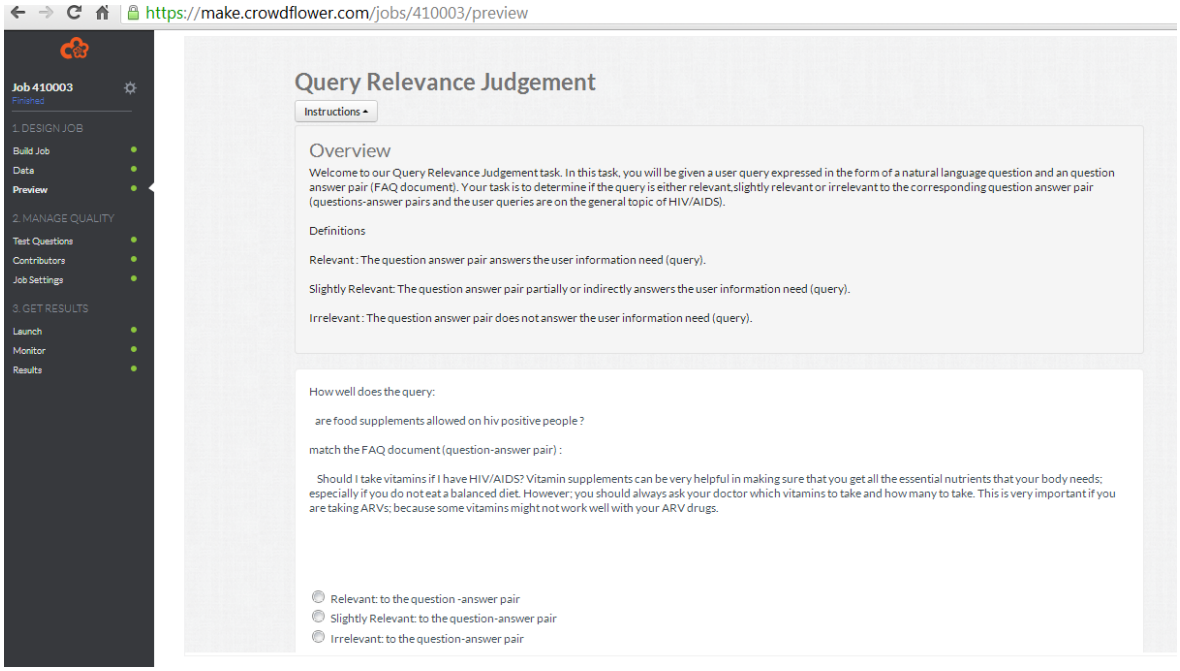


Figure 3.5: User Interface for Collecting Query Relevance Judgements

3.4.3 Worker Validation

One of the main disadvantages of crowdsourcing is that participants can randomly provide assessments in order to get payment for each task completed. Several approaches have been proposed to alleviate this. In particular, Crowdflower provides a mechanism of identify random/malicious assessments by introducing test questions with verifiable answers before participants can be allowed to contribute in a task. For example, in our case, participants were required to complete 4 test questions and attain a score of at least 75% before they

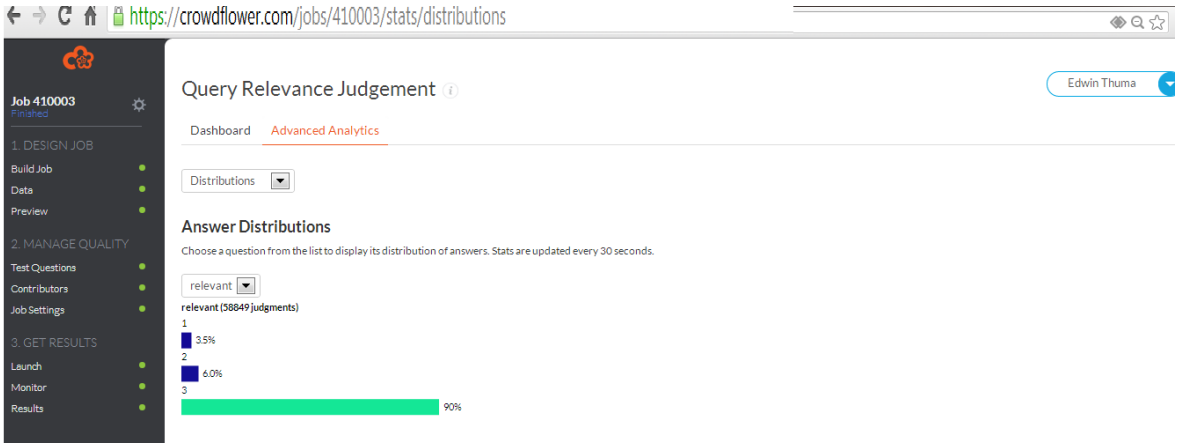


Figure 3.6: Answer Distributions

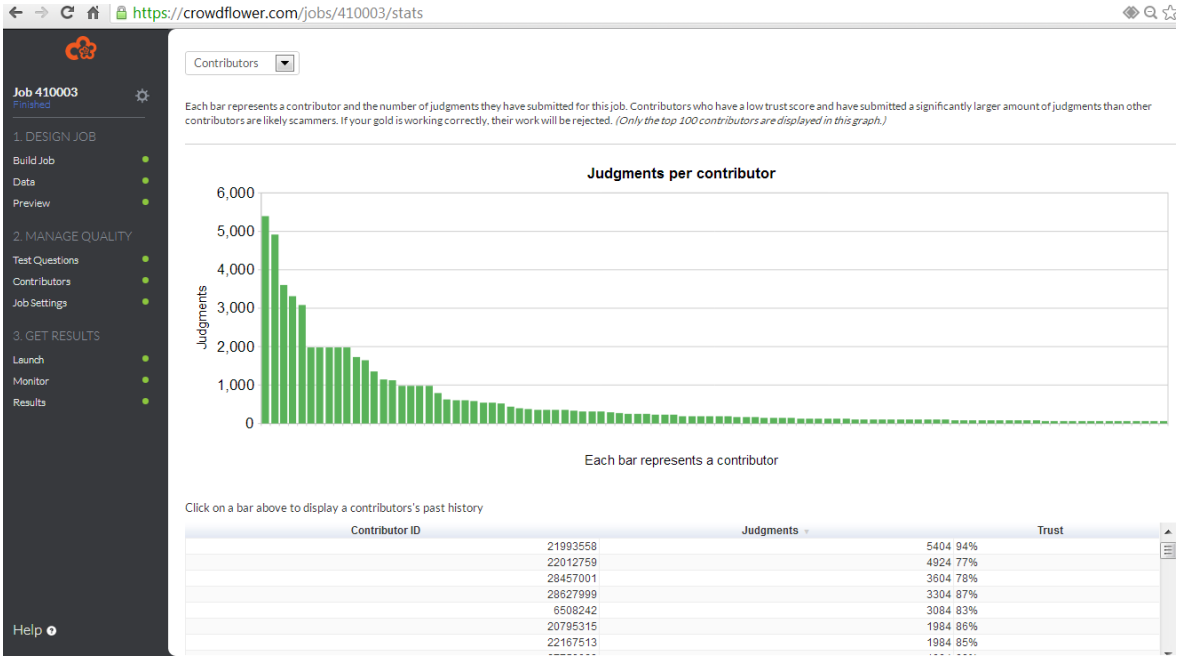


Figure 3.7: Judgements Per Participant

could be allowed to participate in a task. These test questions were also randomly introduced in the middle of a task to ensure that the quality of the relevance assessment provided by participants is maintained. Those participants who averaged below 75% in the middle of a task were also ejected from the task and their contributions discarded. For example, in Figure 3.8, a total of 1713 query relevance judgements were discarded after participants who were allowed to contribute in the task scored below 75% in the middle of the task.

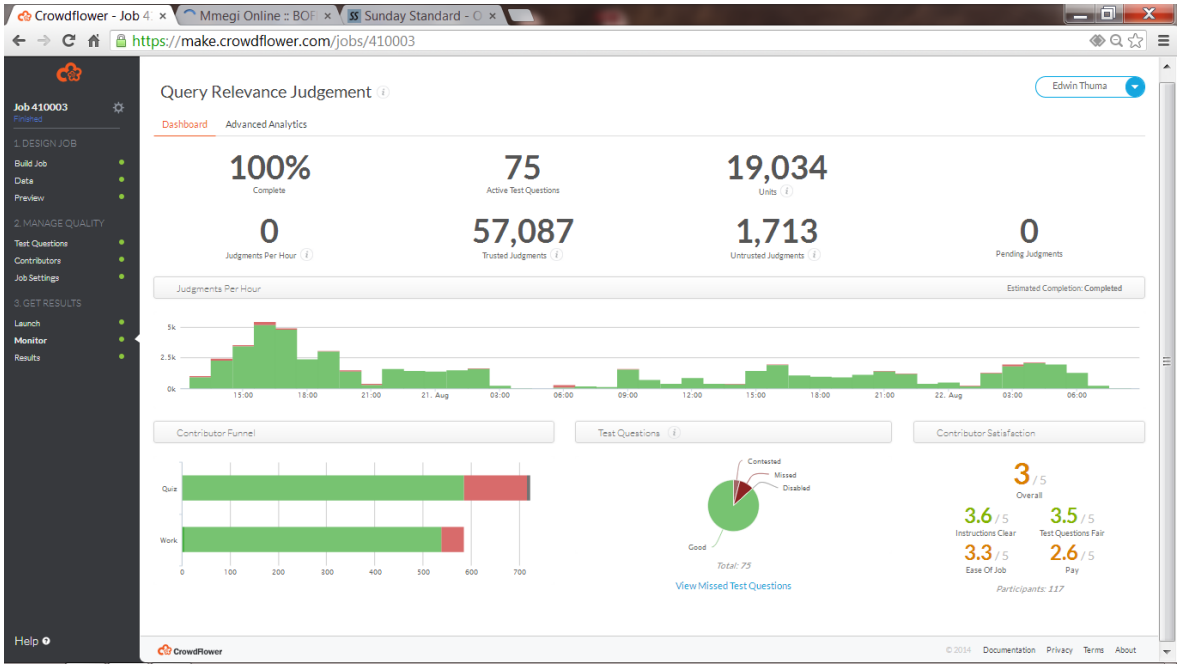


Figure 3.8: Complete Statistics of the Query Relevance Judgements

3.4.4 Measuring the Agreement Between the Query Relevance Judgements

The purpose of the crowdsourcing task was to gather additional query relevance judgements in order to use them for validating the quality of our previous relevance assessments provided by participants in Botswana. We validate the quality of our relevance judgements by measuring the agreement between the assessments provided by the two groups. In particular, we measure how often participants from the different groups provided the same assessment for each query document pair. Recall that in Section 3.2.2 we converted all the query relevance judgements from our previous study in Botswana into binary and aggregated them into a single query relevance file. Similarly, the crowd sourced judgements were converted into binary and they were also aggregated to form a single query relevance file. We used the Cohen's kappa statistic to measure the agreement between the assessments provided by the two groups (using the two query relevance files). The Cohen's kappa statistic is expressed as (Fleiss et al., 2004):

$$\kappa = \frac{Pr(a) - Pr(e)}{1 - Pr(e)}, \tag{3.7}$$

where $Pr(a)$ is the relative observed agreement between the query relevance judgement, and $Pr(e)$ is the probability of random agreement. So applying the Cohen's kappa statistic on the data in Table 3.4 we get a kappa statistic of 0.611 signifying a fair to good agreement in

Table 3.4: Analysis of the Query relevance Judgements Provided by Botswana participants and crowdsourced participants.

		Botswana Participants	
		Relevant Documents	*non-Relevant Document*
Crowdsourced Participants	*Relevant Documents*	2586	2750
	non-Relevant Documents	206	51545

the query relevance judgements provided by the two groups. A kappa statistic of 1 signifies total agreement and a value of 0 signifies total disagreement. Fleiss et al. (2004) characterised kappa statistics above 0.75 as excellent, 0.40 to 0.75 as fair to good and below 0.40 as poor. Other magnitude guidelines have been proposed in the literature for interpreting the kappa statistic. For example, Landis and Koch (1977) characterized values < 0 as indicating no agreement and 0.0-0.20 as slight,0.21-0.40 as fair, 0.41-0.60 as moderate, 0.61-0.80 as substantial, and 0.81-1.0 as almost perfect agreement.

3.5 Conclusions

In this chapter, we described the test collection that we created to use in subsequent chapters to evaluate whether we can reduce the rate at which users abandon their search before their information has been satisfied by using information from previous searches. In Section 3.2.2, we described how we built the query relevance judgements (qrels) for this test collection. These qrels are vital in subsequent chapters because we use then in our evaluation of our FAQ retrieval system. Also they help us in linking the queries to the FAQ documents. For example, in Chapter 5, we use the information provided by the qrels to enrich the FAQ documents with additional terms from our query log in order to help resolve the term mismatch problem between the queries and the relevant FAQ documents. Since these qrels are an important part of this thesis, in Section 3.4 we assess the quality of these by measuring the agreement between the judgements provided by two different groups of participants. Using the Cohen's kappa statistic, we obtained an agreement of 0.611 . This value signifies a fair to good agreement in the qrels provided by the two groups.

Chapter 4

Baseline Iterative Semi-Automated SMS-Based FAQ Retrieval System

4.1 Introduction and Motivation

In this chapter, we describe a baseline system for our semi-automated FAQ retrieval system. The baseline system will enable us to assess in subsequent chapters whether we can improve the retrieval performance of our FAQ retrieval system by using information from previous searches. In Section 4.2, we discuss several ways in which any user can interact with our FAQ retrieval system. As a result, we propose an iterative interaction retrieval strategy in order to overcome the issue of presentation of results on low-end mobile phones. The main disadvantage of this retrieval strategy is that users are likely to abandon the search if the system fails to return the relevant FAQ document after a few iterations. In this chapter, we will carry out an investigation to determine the search length desired by users.

Furthermore, it is imperative that we are able to assess the usability of our system. Several Information Retrieval (IR) evaluation measures that we reviewed in Chapter 3 are only designed to measure the retrieval effectiveness of an IR system. They do not necessarily measure whether users are likely to be satisfied, which is another important measure of usability (Frøkjær et al., 2000). In this chapter, we propose to use the bad abandonment statistics from previous searches to estimate the probability that any random user will be satisfied when using our system. In addition, we also carry out an empirical evaluation to determine the term weighting model to use in our baseline system. The remainder of this chapter is organised as follows:

- In Section 4.2, we discuss several ways in which any user can interact with our FAQ retrieval system.

- In Section 4.3, we describe in detail the main building blocks of our semi-automated FAQ retrieval system.
- In Section 4.4, we investigate in detail the number of iterations users are willing to tolerate before abandoning the iterative FAQ retrieval search process. We also investigate whether the search length of previous searches has an effect on the search length of subsequent searches. We then use the bad abandonment statistics from previous searches to estimate the probability that any random user will be satisfied when using our system.
- In Section 4.5, we carry out an empirical evaluation to determine the term weighting model to use in our baseline system.

4.2 User Interaction with the FAQ Retrieval System

Previous studies have proposed one method for displaying a ranked list of FAQ documents on low-end mobile phones for each SMS query. In particular, they proposed that the top 5 ranked FAQ documents be returned to the user for each SMS query (Contractor et al., 2013, Kothari et al., 2009, Leveling, 2012). The main disadvantage of this approach in low-end devices as discussed earlier in Chapter 1 (Section 1.3.3) is that the maximum number of characters per SMS is 160. If an SMS exceeds that limit, it is split into multiple SMS messages that are delivered to the recipients' mobile phones as separate messages. Therefore, users may find it difficult to navigate and identify the relevant FAQ document from several SMS messages that are returned to the user for each SMS query. For example, considering the length of our FAQ documents, for each SMS query, the user will receive approximately 15 SMS messages, which may be addressing different information needs. Some users, especially the semi-literate, may find it difficult to navigate through this long list of SMS messages to find the relevant FAQ document.

Returning a ranked list of the question part of the top 5 ranked FAQ documents for each user query is another option that has not drawn the attention of may researchers. From this ranked list, a user selects the FAQ document he or she believe is related to the submitted query to retrieve the answer part. The main advantage of this approach is that it is likely to reduce the number of iterations between the user and the FAQ retrieval system if a user can identify lowly ranked FAQ documents as related to the original query. However, since only the question part is displayed, it is likely that some users may ignore some questions returned even though they are related to the original query because of the lexical difference (term mismatch) between the query and the returned question part.

In this thesis, we differ with the aforementioned methods for displaying results on low-end mobile phones by proposing an iterative interaction retrieval strategy. Our aim is to overcome the issue of presentation of results on low-end mobile phones that were discussed in Chapter 1 (Section 1.3.3). In our proposed iterative interaction retrieval strategy (see Figure 4.1), Users send an SMS query. For each SMS query, the system automatically ranks the FAQ documents in the FAQ document collection. The top ranked FAQ document is returned to the user. If the user is satisfied that this FAQ document matches the SMS query, the user responds with "*YES*" or remain idle for time τ and the interaction terminates (see Figure 4.1). If the user is not satisfied, they reply with "*NO*", and the system displays the next highest ranked FAQ document (see Figure 4.2). The process is repeated until the user responds with "*YES*".

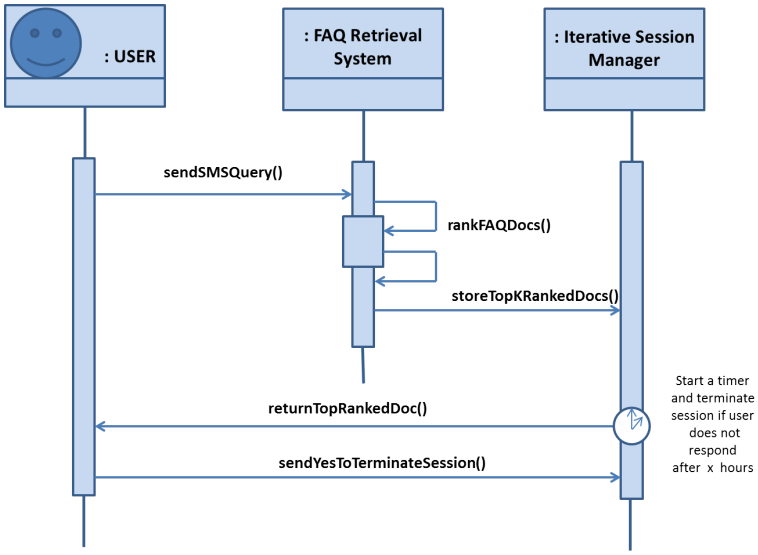


Figure 4.1: User responds with "*YES*" or remain idle for x hours and the interaction terminates.

4.3 System Architecture

This section describes and discusses the main building blocks of the baseline FAQ retrieval system, illustrated in Figure 4.3. The first part of this section starts by describing how the FAQ documents are stored and prepared for retrieval (Section 4.3.1). This is then followed by a description of how the system accepts the user's SMS queries and prepares them for matching and retrieving the relevant FAQ documents (Section 4.3.2). In Section 4.3.3, a description of how the system matches and ranks each FAQ document in the collection is provided. This is followed by a description of how the system manages communication with the user (Section 4.3.4).

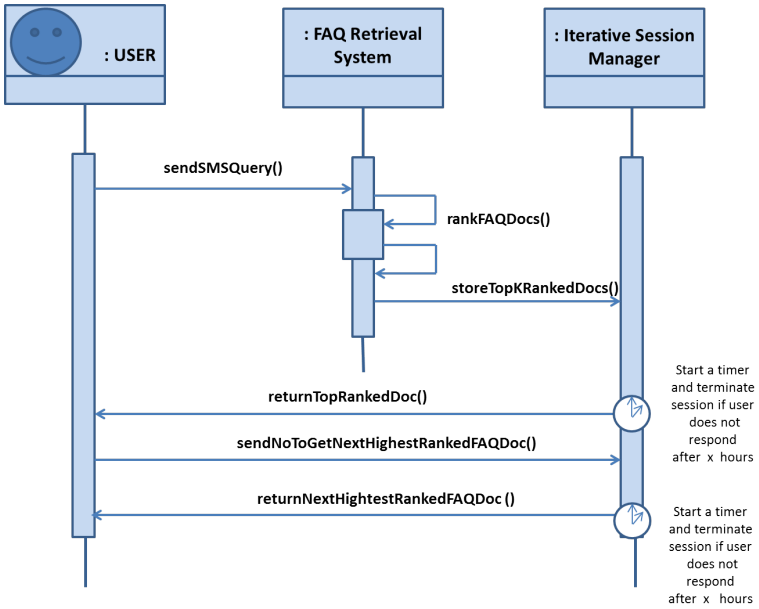


Figure 4.2: User responds with "*NO*" and the system displays the next highest ranked FAQ document.

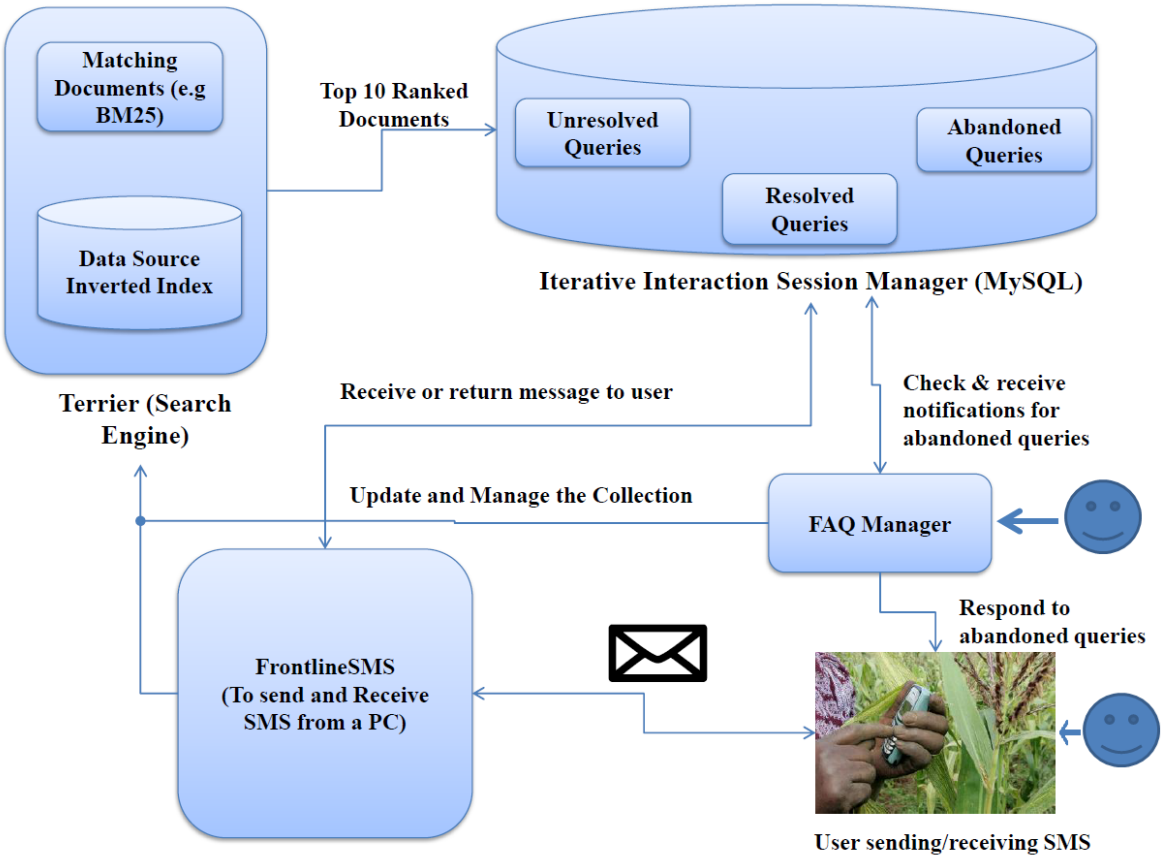


Figure 4.3: Baseline System Architecture

4.3.1 Data Source and the Inverted Index

The static set of FAQ documents in the test collection, which we created as described in Chapter 3 were first preprocessed and stored in a suitable data structure called an inverted

index to allow for efficient retrieval (Frakes and Baeza-Yates, 1992, Van Rijsbergen, 1979). This is depicted as the data source (inverted index) in Figure 4.3. During the preprocessing steps, the FAQ documents were tokenised to identify each term/token (Manning et al., 2008). These tokens were then normalised. Normalisation is the process of standardising the tokens so that matches occur between the query and the indexed FAQ documents despite differences in the character sequence of the tokens (Manning et al., 2008). In this work, we used mapping rules that remove character like hyphens to create equivalence classes to use for normalisation. For example, all occurrences of "breast feeding" were normalised to "breastfeeding" and all occurrences of "check-up" were normalised to "checkup". After normalisation, stopwords were removed and the remaining tokens were stemmed using the Porter Stemming algorithm (Porter, 1997). In this work, all the preprocessing and the indexing of the FAQ documents were performed using the open source Terrier-3.5[25] Information Retrieval Platform (Ounis et al., 2005).

4.3.2 Retrieving the FAQ Relevant Documents

The FAQ retrieval system for HIV/AIDS receives the user's SMS queries through a modem connected to a computer using FrontlineSMS[26] as illustrated in Figure 4.3. These SMS queries can be natural language questions or a set of keywords. For example, a user might send a natural language question "Should I stop drinking alcohol now that I am HIV positive?" or a list of keywords "Alcohol hiv positive" as an SMS query. The system then performs preprocessing steps to the users' SMS query following the same steps carried out during the indexing of the FAQ documents (tokenisation, normalisation, stopword removal and stemming). After preprocessing, the system uses a term weighting model to automatically match and rank the FAQ documents in the collection to the SMS query.

4.3.3 Matching and Ranking the FAQ Documents

For each SMS query sent by the user, the FAQ retrieval system ranks the FAQ documents in the collection in decreasing order of relevance. The top k ranked FAQ documents are stored in a MySQL database to be returned iteratively to the user by the Iterative Interaction Session Manager as described in Section 4.3.4. Typically, the order of relevance of an FAQ document to a given query can be estimated using an IR term weighting model. An ideal term weighting model must be *effective* and *efficient*. An *effective* term weighting model ranks as many relevant documents as possible above the non-relevant documents. On the other-hand, an *efficient* term weighting model in an iterative system must be able to respond to the user

[25] http://terrier.org
[26] http://www.frontlinesms.com/

with the correct FAQ document after a few iterations (shorter search length) so that users do not abandon the search before their information needs have been satisfied. In this thesis, a suitable term weighting model that is *effective* and *efficient* for the proposed baseline FAQ retrieval system will be selected based on the results of a thorough empirical evaluation.

4.3.4 Iterative Interaction Session Manager

Recall that in Section 4.3, we proposed an iterative interaction retrieval strategy, where the system engages the user in the question answering process. This communication between the user and the system is managed by the Iterative Interaction Session Manager as illustrated in Figure 4.3. In particular, for each user query, the top k ranked FAQ documents are initially stored in a table of unresolved queries and the top ranked FAQ document is returned to the user (see Figure 4.1). The user will then respond with either a "*YES*" or "*NO*" to indicate whether the systems' response is relevant to the query or not. If the user responds with a "*YES*", the SMS query and the rank of the relevant FAQ document are moved to a table of resolved queries and the session terminates. If on the other-hand the user responds with a "*NO*", the FAQ document that is a rank below the previous one is returned to the user (see Figure 4.2). The session is maintained until the user sends a "*YES*" or rephrases the query or submits another query.

At the core of the session management is the users' mobile phone number, which is stored as a primary key in the unresolved queries table. If a user still has an unresolved query and then rephrases the query or submits another query without sending a "*YES*" or "*NO*", a new session will be initiated and the unresolved query will be moved to the abandoned queries table together with the rank at which it was abandoned. The session manager also periodically checks the timestamps of the unresolved queries. Any query that remains unresolved for more than x hours is moved to the abandoned queries table. Following some of the suggestions in the framework proposed by Moreo et al. (2012b), the FAQ manager (a human expert) will be notified automatically whenever queries are added to the abandoned queries table. The FAQ manager will then manually check the FAQ document collection to determine if there is a relevant FAQ document to those queries. If it exists, the FAQ manager will return the FAQ document to the user. If it does not exist, the FAQ manager will then consult the relevant sources for a solution / answer to update the collection. If there is no solution at all, the user will be notified by the FAQ manager.

4.4 Measuring Search Length and Estimating the Probability of User Satisfaction

Previous research findings into Web search query abandonment have shown that if not satisfied, users will quickly disengage with a system (Chuklin and Serdyukov, 2012, Li et al., 2009). Therefore, it is crucial that the proposed FAQ retrieval system provides the correct FAQ documents within as few iterations as possible. Cooper (1968) defined a single measure of retrieval effectiveness based on the weak ordering action of retrieval systems called the Expected Search Length. This evaluation measure is based on the calculations of the expected number of irrelevant documents in the collection that would have to be searched through before the desired number of relevant documents could be found. In this section, we measure the number of iterations (search length) that users will tolerate before giving up (before reaching the expected search length). Also in this section, we use the users' abandonment statistics to produce a means of evaluating how likely users are to be satisfied when using our system. The ultimate goal is to develop an FAQ retrieval system that will return as many relevant FAQ documents as possible to the users before they abandon the search process. The following two research questions were identified to help us in our investigation:

Chapter 4-Research Question One (C4-RQ1): What is the maximum number of iterations that users are willing to tolerate before abandoning the iterative search process?

Chapter 4-Research Question Two (C4-RQ2): Does the search length of previous searches influence the search length of subsequent searches?

4.4.1 Detecting Good and Bad Query Abandonment

In Web search, query abandonment or search session abandonment is when users do not select any results presented for a given query or information need (Chuklin and Serdyukov, 2012, Koumpouri and Simaki, 2012, Li et al., 2009, Stamou and Efthimiadis, 2010). Understanding why users abandon the search process is always difficult as there are many reasons that might have prompted the user not to select the results presented. Some of the reasons might be that the results presented are not satisfactory or that the user closed the search session by mistake. Previous work on query abandonment relied on clicks on the presented results to infer user satisfaction with the results. For example, Li et al. (2009) introduced the concept of good abandonment and bad abandonment. They identified good abandonment by the presence of clicks on the results presented to the user for any query submitted and bad query abandonment by the absence of such clicks. Li et al. (2009) compared abandonment for desktop and mobile search across different locales and their findings suggest that the good abandonment rate for mobile search is slightly higher than for desktop search.

Huang et al. (2011) on the other-hand examined cursor movement and gaze positions on Search Engine Results Pages (SERP) to infer good and bad abandonment. They found that cursor movement over SERP can also provide information on user satisfaction as it correlates with eye gaze and can capture the behaviour that does not lead to clicks. Diriye et al. (2012) have highlighted that there is no perfect way of measuring good or bad abandonment and they have shown that one in five good abandonment instances does not relate to user satisfaction. In their work, they studied the underlying reason for abandonment by training Multiple Additive Regression Trees (MART) (Friedman et al., 2000) classifiers using features of the query and the results, interaction with the result page and the full search session. Next, they used these classifiers to predict the reasons for observing search abandonment.

To the best of our knowledge, no work has been reported in the literature that investigates the number of iterations users are willing to tolerate in an iterative interaction retrieval strategy. In order to accurately measure the number of iterations a user will tolerate before giving up, we must be able to detect the good and bad query abandonment. Hence, the next section describes a new iterative interaction retrieval approach, which is tailored for detecting good and bad query abandonment.

4.4.2 Detecting Good and Bad Query Abandonment in an SMS-Based FAQ Retrieval System

In this section, we define a new iterative interaction retrieval strategy, which is tailored for detecting good and bad query abandonment in an SMS-Based FAQ retrieval setting. In this new iterative interaction retrieval strategy, for any SMS query sent by the user, the system ranks the FAQ documents in the FAQ document collection. The question part of the top ranked FAQ document is returned to the user. If the user is satisfied that this question matches their SMS query, they respond with a "*YES*" and the system sends the associated answer. If the user is not satisfied, they reply with a "*NO*", the system then displays the next highest ranked question part and the process is repeated.

Based on this new definition of the iterative interaction retrieval approach, this work follows the work by Li et al. (2009) and define two ways in which a user interaction can be terminated: good and bad abandonment. Good abandonment is defined as the termination of the iterative process by the user sending a "*YES*" to retrieve an answer. Bad abandonment is defined as the termination of the iterative process by the user not responding to a question returned by the system for over an hour or when they respond by sending another query or by rephrasing the query. As highlighted earlier, it is important that this good and bad abandonment can be accurately measured. Therefore, the question and answer parts are not returned together. Forcing the user to respond gives an unambiguous indicator that the search process

has terminated successfully.

In Section 4.4.3, we present a description of the FAQ retrieval platform used in the experimental investigation and evaluation to answer the aforementioned research questions *C4-RQ1* and *C4-RQ2*.

4.4.3 FAQ Retrieval Platform

In this experimental investigation, Terrier-3.5[25], an open source IR platform was used for indexing and searching for the relevant FAQ documents. Each FAQ document from the information source described in Section 1.2 (Chapter 1) was indexed as a single FAQ document. Before indexing, the FAQ documents were pre-preprocessed. This involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). To filter out terms that appear in many FAQ documents, a stopword list was not used during the indexing and the retrieval process. Instead, terms that had low IDF were ignored when scoring the FAQ documents. Indeed, all the terms with term frequency higher than the number of the FAQ documents (205) were considered to be low IDF terms. The weighting model used for the retrieval of the relevant FAQ documents was BM25 and the default Terrier-3.5 settings were used: $k_1 = 1.2$, $k_3 = 8$ and $b = 0.75$. The Terrier-based FAQ retrieval system was receiving and responding to any incoming SMS message through a GSM modem connected to a desktop computer as illustrated in Figure 4.3. For each query received by the system, the system would rank 10 FAQ documents in the FAQ document collection and would return a question associated with the top ranked FAQ document to the user. Recall that in Section 4.3.4, our FAQ retrieval system can be configured to rank and retrieved the top k FAQ documents. In this study, we configured our system to retrieve up to 10 FAQ documents only. The search sessions for each user were monitored across three tables created in MySQL Server 5.1. The first table stored queries that have not yet been resolved (user has not sent a "*YES*" to retrieve the relevant answer pair). The second table stored queries that have been resolved (user has sent a "*YES*" to retrieve the relevant answer part) and the third table stored abandoned queries.

4.4.4 Methodology

We conducted a second user study at the University of Glasgow, School of Computing Science from form 10^{th} August 2012 to 30^{th} August 2012 to investigate the number of iterations users are willing to tolerate in an iterative interaction retrieval strategy. This study was granted the University of Glasgow ethics approval and was allocated the ethics project reference number: CSE01082. A total of 20 participants were recruited to take part in this study. In total, 8 were female and 12 were male. Their ages ranged from 18 to 40. The participants

were recruited through an e-mail request, which was sent to the University of Glasgow graduate students mailing list. Most of the participants were students at the University of Glasgow and a few of them were their friends and family members. The participants completed the task during their spare time over a total period of two weeks and were compensated for their time and efforts after completing the study.

In an earlier study (Described in Chapter 3), 85 participants in Botswana generated 957 SMS queries, 750 of which could subsequently be matched to the relevant FAQ documents. These SMS queries were corrected for spelling errors so that such a confounding variable does not influence the outcome of the experiments. We selected 16 SMS queries from the 750 SMS queries that could be matched to the relevant FAQ documents in the collection to use in this study. In order to be able to answer research question *C4-RQ2*, these queries were chosen and split into two groups based on how highly the system ranked the relevant FAQ documents. *Set-1* contained 8 queries for which the relevant FAQ documents were ranked between 1 and 3. *Set-2* contained 5 queries for which the relevant FAQ documents were ranked between 4 and 7 and 3 queries for which no relevant FAQ document could be found using the FAQ retrieval system described in Section 4.4.3.

The 20 participants were randomly divided into two groups of 10 (A and B). Participants were asked to query the system using the SMS queries in *Set-1* and *Set-2*. Participants in group A were given *Set-1* followed by *Set-2* whilst those in group B were given *Set-2* followed by *Set-1*. This experimental design is suitable for investigating research question *RQ2* as participants within the two groups will be exposed to very different search lengths in their initial use of the system. In particular, participants in group A were initially exposed to a shorter search length (given queries in *Set-1*, while those in group B were initially exposed to a longer search length (given queries in *Set-2*. Participants in group A were later exposed to a longer search length (given queries in *Set-2*, while those in group B were later exposed to a shorter search length (given queries in *Set-1*).

The participants were given a demonstration on how to retrieve the relevant FAQ documents through SMS using a separate set of queries. After the demonstration, participants were given up to two weeks (one week for each set of queries) to perform the task in their spare time. For each question returned by the system, the participants were asked to respond by identifying whether the question they received was relevant to what they asked or if it was irrelevant. If it was irrelevant, the participants were required to send a "*NO*" to obtain the next ranked question. If on the other-hand the question was relevant, the participants were required to send a "*YES*" to retrieve the answer part of the FAQ document.

The participants were not advised to send another query from the list when the initial question retrieved was not relevant (implicitly advising them to terminate the iterative interaction process). They were also not advised to remain idle if they did not receive relevant questions

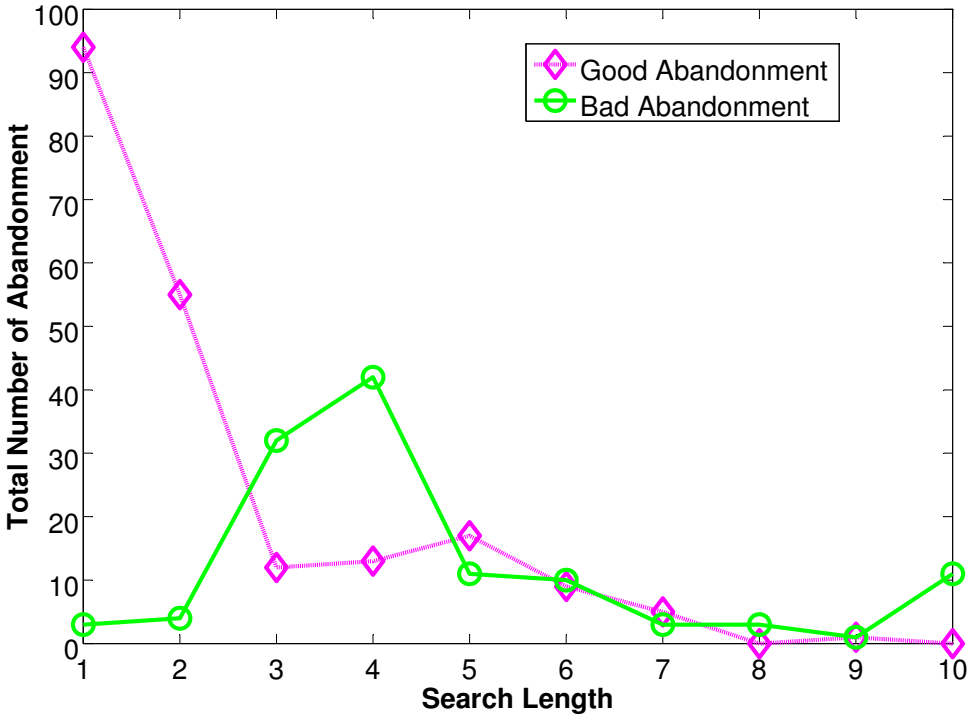


Figure 4.4: The number of iterations (search length) to good and bad abandonment for all the participants.

as responses. The reason for this was to be able to set an hour as a threshold for a permissible idle time per-user session. For each query received, the FAQ retrieval system recorded either bad query abandonment or good query abandonment based on the interaction with the users. Also recorded was the query set, either *Set-1* or *Set-2* corresponding to that abandonment and the number of iterations between the users and the FAQ retrieval system to reach that abandonment.

4.4.5 Results and Analysis

Figure 4.4 shows the number of iterations to good and bad abandonment for all the participants. In this figure, the x-axis represents the number of iterations (search length) a user tolerates before abandoning the iterative search process. The y-axis represents the total number of abandonments at each search length. The results suggest that most participants from both groups can tolerate two to three iterations as evidenced by the high number of bad abandoned queries after three iterations (*C4-RQ1*). These values will be discussed within the context of evaluation of the FAQ retrieval system in Section 4.4.6. When the results are split by the two groups (A and B), the behaviour plotted in Figure 4.5 is observed. The behaviour across the

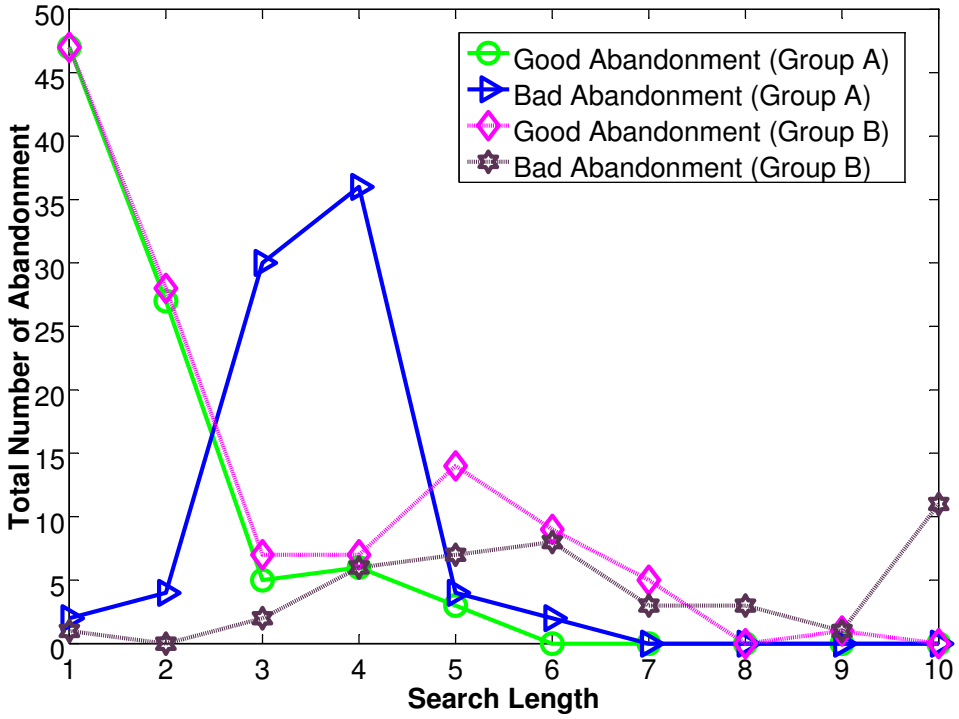


Figure 4.5: The number of iterations (search length) to good and bad abandonment when the participants are split into two groups (A and B). Participants in group A were initially exposed to a shorter search length and they were later exposed to a longer search length. Participants in group B on the other hand were initially exposed to a longer search length and they were later exposed to a shorter search length.

two groups is clearly different. In particular, bad abandonment in group A tends to happen sooner than in group B (Mann-Whitney U test, $p < 0.05$), suggesting that previous searches can significantly influence subsequent behaviour (*C4-RQ2*). One plausible explanation for this result is that group A participants were used to receiving the relevant question part after a few iterations and when they were given the test set with longer search length they became displeased and abandoned the search earlier. These results highlights the importance of having an FAQ retrieval system that performs well across all the user queries to avoid a high rate of bad abandonment.

It was also discovered that there were a few instances where participants responded with a "*YES*" when the question returned was not relevant and in some instances they responded with "*NO*" even though the question they asked was related to the question returned by the system. These instances illustrate the limitations of our approach in measuring good or bad abandonment. However, they do still provide information regarding how many iterations users are willing to tolerate. Put simply, a user who says "*NO*" to a question that is relevant is clearly still willing to engage with the system. During the debriefing, some participants

reported that they responded with a "*YES*" to view an answer to the question returned even though they knew it was not related to the original query. A small number of those interviewed suggested that, it is important to return the whole FAQ document at each iteration so that users can see answers to the questions returned. They argued that, without returning the whole FAQ document, there may be instances were they are forced to abandon their original query in order to view an answer to another question returned to them.

4.4.6 Using the Bad Abandonment Data to Evaluate the FAQ Retrieval System

One of the goals of this research was to use the abandonment statistics to produce a means for evaluating the proposed iterative FAQ retrieval system's performance. Traditional evaluation metrics such as Mean Reciprocal Rank (MRR) do not take into account the user abandonment statistics. The Expected Reciprocal Rank (ERR) (Chapelle et al., 2009) is an example of an evaluation metric that takes into account the probability that the user is satisfied. However, this measure simplifies to the traditional MRR in a binary relevance setting (when the returned documents are either relevant or non-relevant) as in the case of our system.

Table 4.1 summarises the retrieval performance of the current FAQ retrieval system. The system was evaluated using 300 randomly selected SMS queries from the 750 SMS queries that could be matched to the relevant FAQ documents in the collection. For each query, a maximum of 5 FAQ documents were retrieved. The total number of retrieved and relevant FAQ documents was 361. This number exceeded the number of queries because some queries had more than one relevant FAQ documents. A reasonably good MRR of 0.4319 was recorded, which means that on average the first relevant FAQ document is ranked approximately second on the retrieved set.

Table 4.1: Retrieval Performance for the FAQ System.

	Retrieval Perfomance Evaluation
Number of SMS Queries	300
Number of Retrieved FAQ documents	1500
Number of Relevant FAQ documents in the Collection	860
Number of Retrieved and Relevant FAQ Documents	361
Mean Reciprocal Rank (MRR)	0.4319

To incorporate the user abandonment statistics into the evaluation, the following scheme was devised using the empirical distributions of user abandonment and the rank of the correct FAQ document. Specifically, two population distributions $Q(q, r)$ and $U(u, t)$ were used. The population distribution $Q(q, r)$ was made up of 300 randomly selected SMS queries q from the 750 SMS queries that could be matched to the relevant FAQ documents in the collection. r is the rank at which the system would rank the relevant FAQ document for the

query q in the FAQ document collection. The second population distribution $U(u,t)$ was made up of all bad abandoned queries u from this study (110 in total); t is the rank/search length at bad abandonment. The rank of r ranged from 0 to 5 and the range of t ranged from 1 to 10. The population distributions $Q(q,r)$ and $U(u,t)$ were randomly sampled 100000 times simultaneously and the instances where the rank $r \leq t$ for all instances where r > 0 were counted. In essence, this is approximating the probability that a randomly picked user will be satisfied by the system (i.e. good abandonment). There were 58570 instances recorded for samples where $r \leq t$ for all instances where r > 0 and this value implies that the probability that users would reach good abandonment if using the current system is 0.5857. The estimated metric is far more useful for this particular system than standard evaluation metrics such as MRR and it will help in estimating the percentage gained in good abandonment for any modification made to the system.

4.4.7 Summary

In Section 4.4, we carried out an investigation to determine the number of iterations that users are willing to tolerate before abandoning the iterative search process (*C4-RQ1*). The results of this investigation suggest that the majority of users can tolerate approximately 2 to 3 iterations before abandoning their search process. In addition, we also carried another investigation to determine whether the search length of previous searches influence the search length of subsequent searches (*C4-RQ2*). Our results suggest that people who initially reached good abandonment after a few iterations (3 or fewer) tend to abandon the search faster if their information need is not satisfied (bad abandonment) compared to those who initially reached good abandonment after more iterations (4 or more). The bad abandonment statistics were subsequently used to develop a novel evaluation metric to use in future developments of the system (see Section 4.4.6). Using this metric, it was estimated that the probability that users would reach good abandonment when using the current system is 0.5857. This will serve as a baseline metric to help us estimate the percentage gained in good abandonment for any future modification made to the current system.

4.5 Empirical Evaluation - Choosing a Suitable Baseline Weighting Model

In this section, we present a set of experiments conducted to determine a suitable baseline term weighting model to use in the FAQ retrieval system. Recent evidence from the FIRE2011 and FIRE2012 SMS-Based FAQ retrieval tasks suggests that indexing the whole FAQ document reduces the overall retrieval performance (Contractor et al., 2013, Leveling,

2012, Shaikh et al., 2013). Several experiments will be conducted in this chapter using different term weighting models to establish whether this assertion holds on the HIV/AIDS dataset. Systems at the FIRE SMS-Based FAQ retrieval tasks were evaluated based only on the MRR evaluation measure without taking into consideration other evaluation measures. However, this evaluation measure is only suitable for evaluating retrieval systems that have one relevant document in the collection being searched (Sanderson, 2010). Since the query log analysis conducted in Chapter 3 has shown that some user queries have more than one relevant document in the collection, using the mean reciprocal rank will not give us a clear indication of the retrieval effectiveness of the FAQ retrieval system. This will only give us an indication of how quickly users are likely to find a relevant FAQ document for each SMS query. Instead of relying on a single evaluation measure, this thesis will however rely on several other evaluation measures. In particular, the following evaluation measures will be used: the Mean Average Precision (MAP), MRR, $P@5$ (the precision at a fixed rank 5), and the Recall (Sanderson, 2010). We will also use the abandonment statistics from Section 4.4 to estimate the probability that any random user will be satisfied when using the current system. A term weighting model that is more stable and showing the best retrieval performance across all the evaluation measures will be selected as the baseline weighting model. The other aspect investigated in this section is whether stopword removal has an effect on the retrieval performance of the FAQ retrieval system on the HIV/AIDS dataset. Previous work by Leveling (2012) has shown that using a stopword list provided by several IR retrieval platforms such as Terrier-3.5[25] and Lemur[14] to filter out non informative terms reduces the overal retrieval performance. The following research questions will be investigated:

Chapter 4-Research Question Three (C4-RQ3): Does indexing the question part only improve the overall retrieval performance?

Chapter 4-Research Question Four (C4-RQ4): Does removing stopwords improve the overall retrieval performance?

The remainder of this section is organised as follows: Section 4.5.1 presents a description of how the testing sets for these experiments was created followed by a description of the experimental settings in Section 4.5.2. The results and analysis are presented in Section 4.5.3.

4.5.1 Testing Sets

Recall that in Section 3.3.1 (Chapter 3), we created 10 different training sets and their corresponding testing sets. These 10 different testing sets were used in our empirical evaluation to answer research questions *C4-RQ3* and *C4-RQ4*. All the SMS queries in these testing sets were manually corrected for spelling errors so that such a confounding variable does not influence the outcome of these experiments.

4.5.2 Experimental Settings

For this empirical evaluation, we used Terrier-3.5 (Ounis et al., 2005), an open source IR platform. All the FAQ documents used in this study were first pre-processed before indexing, this involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). In order to be able to answer research question $RQ3$, two different inverted indices were created. The first inverted index was built using the question part only and the second inverted index was built using the whole FAQ document. To filter out non-informative terms, a stopword list was not used during the indexing and the retrieval process. Instead, terms that had low IDF were ignored when scoring the documents. Indeed, all the terms with term frequency higher than the number of the FAQ documents (205) were considered to be low IDF terms. The term weighting models used in this study were: BM25 (Robertson et al., 1996), Poisson Model with Laplace After-Effect and Normalisation 2 (PL2) (Macdonald et al., 2006), Hyper-Geometric DFR Model using Popper's Normalization (DPH) (Amati et al., 2007), Hiemstra Language Model (Hiemstra_LM) (Hiemstra, 2001) and TF-IDF (Robertson, 2004). Default parameters for this weighting models were used as provided in Terrier-3.5 IR retrieval platform. However no document length normalisation was performed for BM25 and TF-IDF (b parameter set to 0). A significant improvement in retrieval performance was observed across all the evaluation measures for the BM25 term weighting model when the b parameter was set to 0.

4.5.3 Results and Analysis

In this section, we report on our empirical evaluation. In particular, we report on whether indexing the question part only for retrieval can improve the overall retrieval effectiveness of our FAQ retrieval system (*C4-RQ3*). We also report on whether stopword removal can improve the overall retrieval effectiveness of our FAQ retrieval system(*C4-RQ4*).

Effect of Indexing The Question Part Only for Retrieval

To answer research question *C4-RQ3*, a Multiple Comparison test was conducted across all the term weighting models to infer whether indexing the question part only for retrieval can improve the overall retrieval performance. We carried out this investigation with stopword removal enabled and also with stopword removal not enabled. A significant improvement in retrieval performance in terms of *MRR* and *MAP* was observed for the Hiemstra_LM term weighting model as denoted by $\triangleright$ in Figure 4.6 and 4.8 (there is no overlap on the confidence intervals between the groups denoted by $\triangleright$ and $\circ$) when only the question part was indexed for retrieval. In this experiment, stopword removal was not enabled for retrieval. Similar

results were observed when stopword removal was enabled for retrieval as denoted by ▷ in Figure 4.7 and 4.9 (there is no overlap on the confidence intervals between the groups denoted by ▷ and ○). However, there was a significant decrease in retrieval performance for the DPH term weighting model when only the question part was indexed for retrieval (there is no overlap on the confidence intervals between the groups denoted by ▷ and ○ in Figure 4.6, 4.7, 4.8 and 4.9). This also resulted in a significant decrease in the probability that any random user will be satisfied when using the system (Figure 4.10 and 4.11 for DPH).

Different results were observed for other term weighting models (*TF-IDF* and *BM25*). There was a significant improvement in retrieval performance when the whole FAQ document was indexed for retrieval as denoted by ♢ and ▷ in Figure 4.7 for *BM25* and *TF-IDF* respectively. This also resulted in a significant increase in the probability that any random user will be satisfied when using the system for BM25 with stopword removal enabled as denoted by ♢ in Figure 4.11. Indexing the whole FAQ document for retrieval also significantly improved the overall recall. This is illustrated in Table 4.2 and 4.3. The increase in recall implies that previously non-retrieved and relevant FAQ documents have been retrieved. The observed increase in recall could be attributed to the reduction in term mismatch between the query and the relevant FAQ document. BM25 yielded the best retrieval performance compared to other term weighting models when the whole FAQ documents were indexed for retrieval and stopword removal enabled.

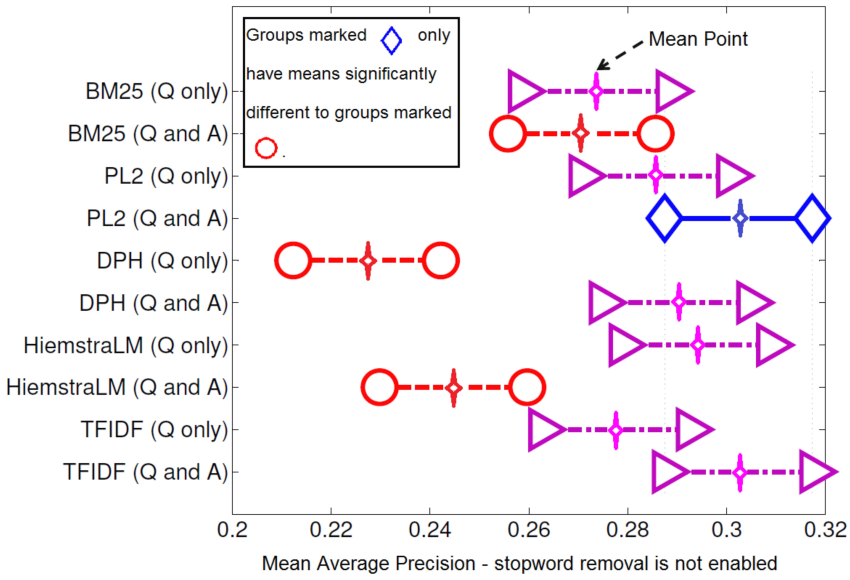


Figure 4.6: The confidence intervals of the MAP means for the 10 different test sets when stopword removal is not enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

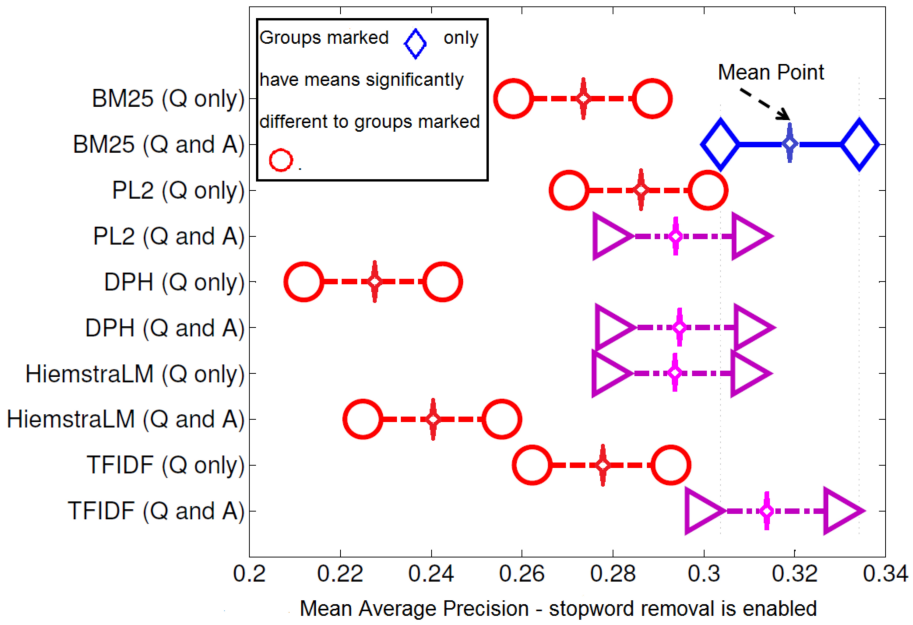


Figure 4.7: The confidence intervals of the MAP means for the 10 different test sets when stopword removal is enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

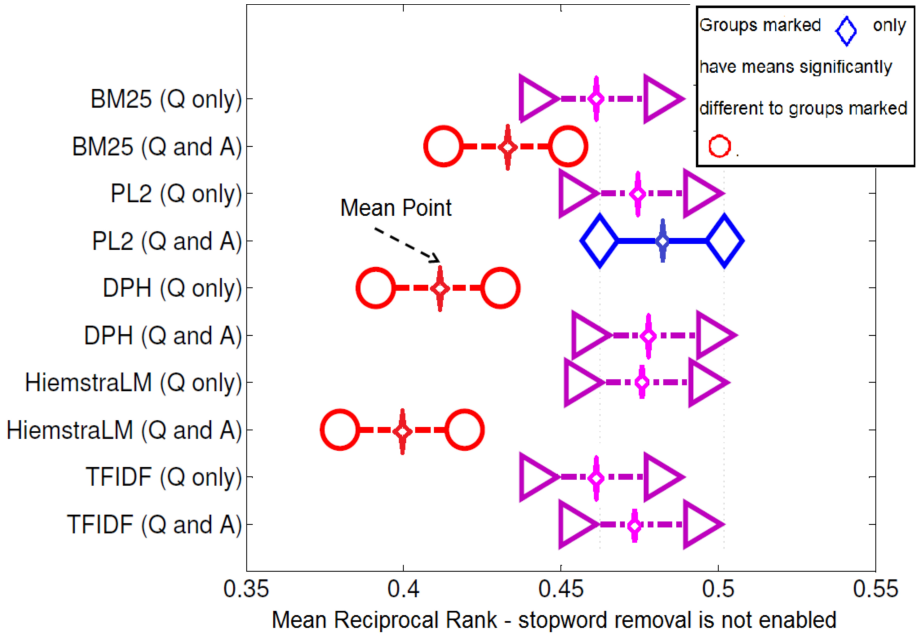


Figure 4.8: The confidence intervals of the MRR means for the 10 different test sets when stopword removal is not enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

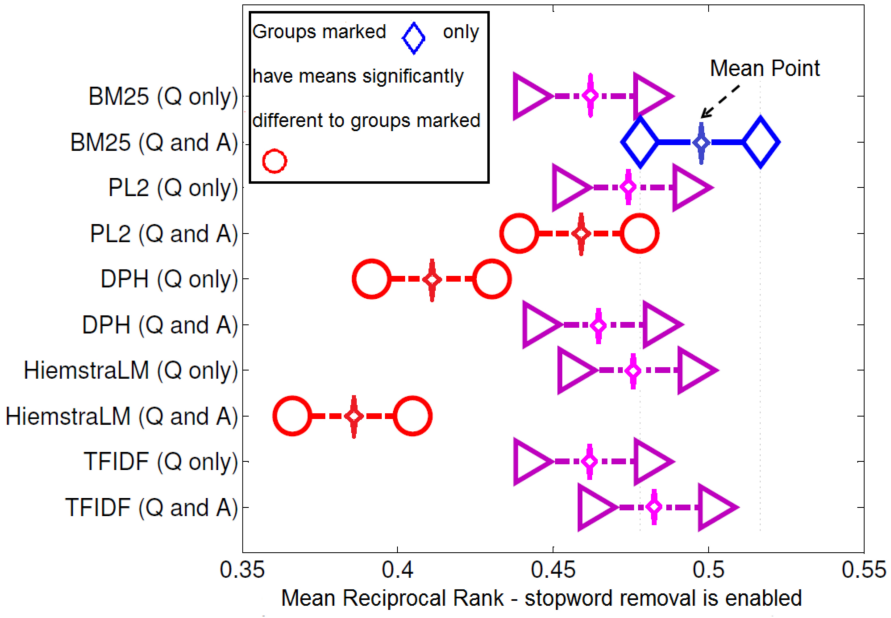


Figure 4.9: The confidence intervals of the MRR means for the 10 different test sets when stopword removal is enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

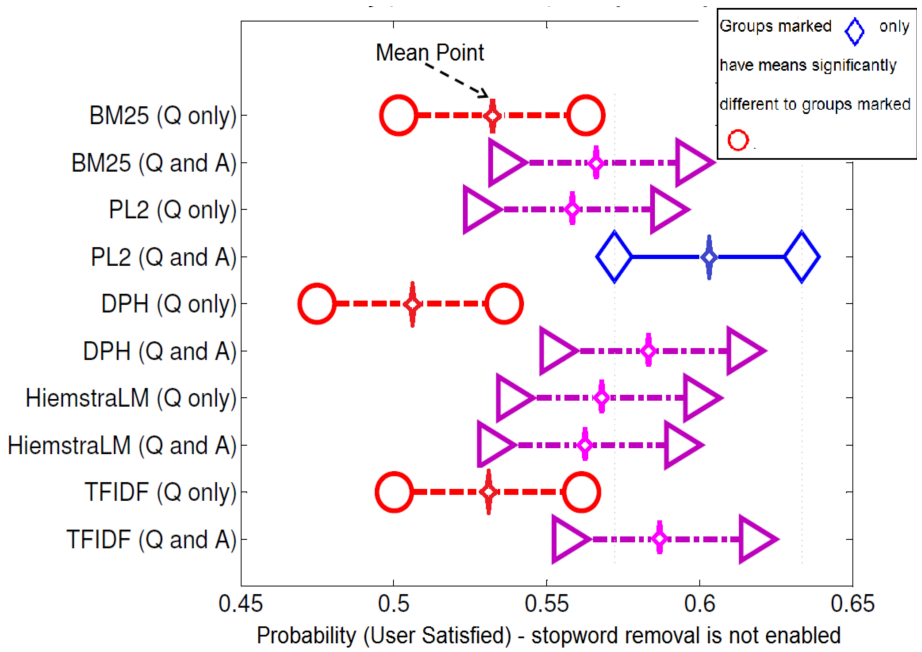


Figure 4.10: The confidence intervals of the probability that any random user will be satisfied for the 10 different test sets when stopword removal is not enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

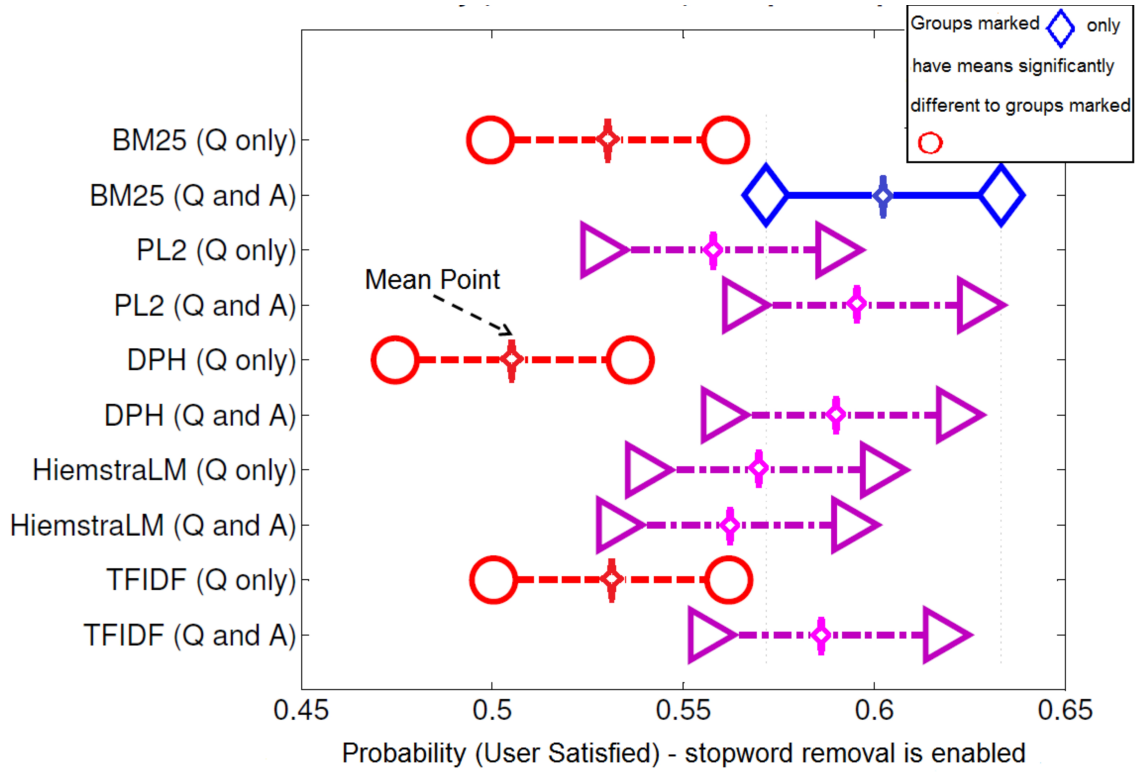


Figure 4.11: The confidence intervals of the probability that any random user will be satisfied for the 10 different test sets when stopword removal is enabled during retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25 (Q only) (BM25 with the question part only indexed for retrieval, BM25 (Q and A) (BM25 with both the question and answer part indexed for retrieval).

Effect of Stopword Removal on Retrieval Performance

The other aspect investigated is whether stopword removal improves the overall retrieval performance (research question *C4-RQ4*). It was observed that when only the question part is indexed for retrieval, there was no term in the collection that had a frequency more than the number of documents (No low IDF terms). Hence, identical MAP and MRR were recorded across the different term weighting models when the system was configured to ignore low IDF terms compared to when it was configured not to remove stopwords (No SWR). This is illustrated by the Multiple Comparison tests in Figures 4.12 and 4.14 for the MAP and MRR respectively. However, when the whole FAQ documents were indexed for retrieval, there were some terms in the collection that had a frequency more than the number of documents (low IDF terms). Hence, different MAP and MRR were recorded across the different term weighting models when the system was configured to ignore low IDF terms compared to when it was configured not to ignore low IDF terms (no stopword removal). This is illustrated by the Multiple Comparison tests in Figures 4.13 and 4.15 for the MAP

Table 4.2: The mean retrieval performance for each collection and term weighting model. Stopword removal was not enabled. A significant improvement in retrieval performance when both the question and the answer part are indexed for retrieval, as denoted by $*$ (Multiple comparison test, $p < 0.05$). Also, there was a significant improvement in retrieval performance when only the question part is indexed for retrieval, as denoted by $\triangleleft$ (Multiple comparison test, $p < 0.05$).

		Test Evaluation Measure				
Collection	Weighting Model	MRR	MAP	P@5	Recall	P(Satisfied)
Q (Only)	BM25	0.4609	0.2734	0.1733	0.3811	0.5322
Q and A		0.4325	0.2707	0.1597	0.3963	0.5660
Q (Only)	PL2	0.4735	0.2856	0.1825	0.3801	0.5577
Q and A		0.4821	0.3024	0.1839	0.4393	0.6027
Q (Only)	DPH	0.4110	0.2273	0.1419	0.3539	0.5055
Q and A		**0.4776**$*$	**0.2897**$*$	**0.1647**$*$	**0.4214**$*$	**0.5827**$*$
Q (Only)	Hiemstra_LM	**0.4752**$\triangleleft$	**0.2937**$\triangleleft$	**0.1869**$\triangleleft$	0.3838	0.5685
Q and A		0.3996	0.2447	0.1455	0.4033	0.5623
Q (Only)	TF_IDF	0.4611	0.2775	0.1789	0.3814	0.5308
Q and A		0.4735	0.3025	0.1751	0.4279	0.5867

Table 4.3: The mean retrieval performance for each collection and term weighting model. Stopword removal enabled. A significant improvement in the retrieval performance when both the question and the answer part are indexed for retrieval, as denoted by $*$ (Multiple comparison test, $p < 0.05$). Also, there was a significant improvement in retrieval performance when only the question part is indexed for retrieval, as denoted by $\triangleleft$ (Multiple comparison test, $p < 0.05$).

		Test Evaluation Measure				
Collection	Weighting Model	MRR	MAP	P@5	Recall	P(Satisfied)
Q (Only)	BM25	0.4609	0.2734	0.1733	0.3811	0.5303
Q and A		0.4973	**0.3189**$*$	0.1880	**0.4246**$*$	**0.6024**$*$
Q (Only)	PL2	0.4735	0.2856	0.1825	0.3801	0.5583
Q and A		0.4584	0.2940	0.1745	**0.4218**$*$	0.5954
Q (Only)	DPH	0.4110	0.2273	0.1419	0.3539	0.5054
Q and A		**0.4640**$*$	**0.2945**$*$	**0.1723**$*$	**0.4182**$*$	**0.5898**$*$
Q (Only)	Hiemstra_LM	**0.4752**$\triangleleft$	**0.2937**$\triangleleft$	**0.1869**$\triangleleft$	0.3838	0.5700
Q and A		0.3854	0.2403	0.1472	0.4075	0.5625
Q (Only)	TF_IDF	0.4611	0.2775	0.1789	0.3814	0.5311
Q and A		0.4817	**0.3142**$*$	0.1829	**0.4191**$*$	**0.5865**$*$

and MRR respectively. BM25 is the only term weighting model to be found to be sensitive to stopword removal as significant improvement in retrieval performance was observed after the removal of stopwords (see Figure 4.13). Table 4.2 and 4.3 provides a comprehensive summary of our experimental results.

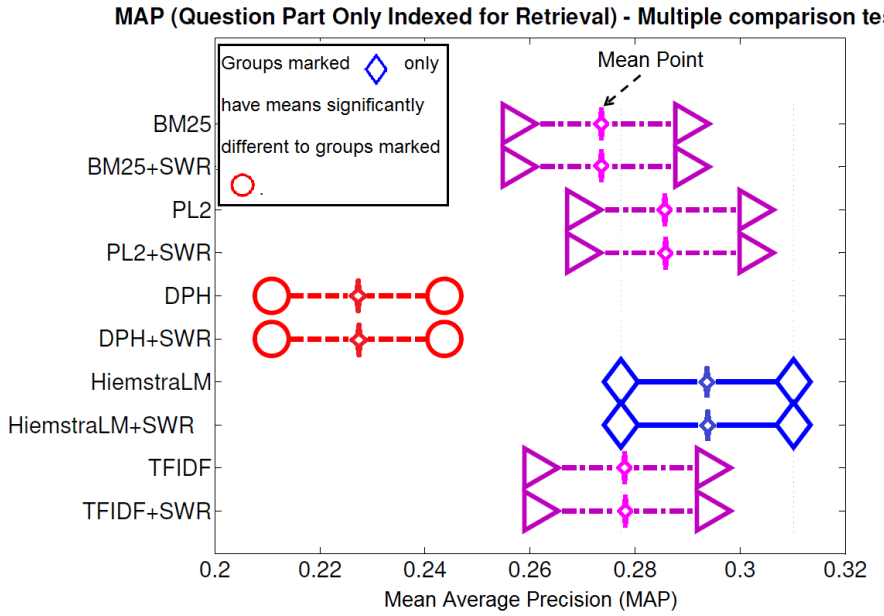


Figure 4.12: The confidence intervals of the MAP means for the 10 different test sets when only the question part is indexed for retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25+SWR (BM25 with stopword removal enabled, PL2+SWR (PL2 with stopword removal enabled).

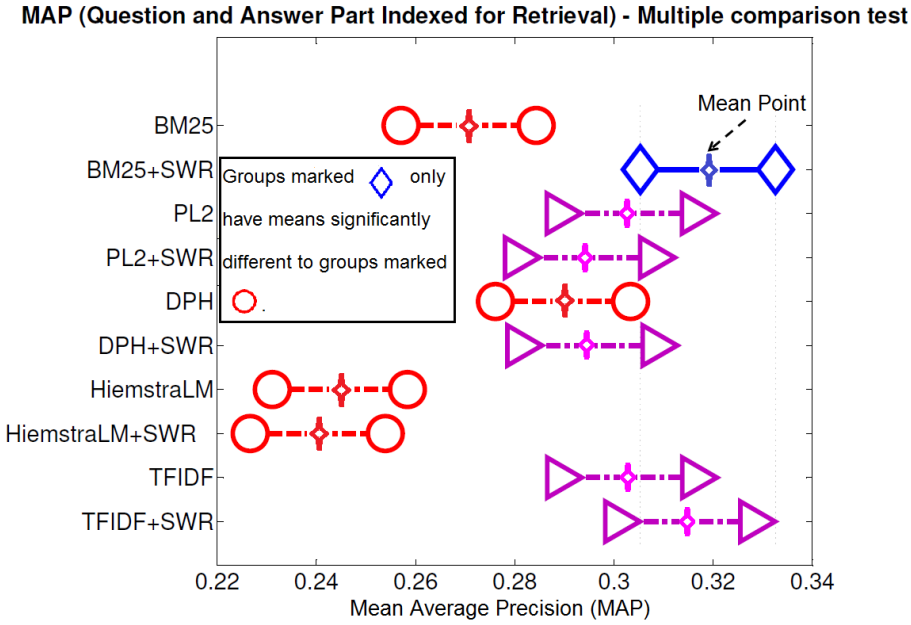


Figure 4.13: The confidence intervals of the MAP means for the 10 different test sets when both the question part and the answer part are indexed for retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25+SWR (BM25 with stopword removal enabled, PL2+SWR (PL2 with stopword removal enabled).

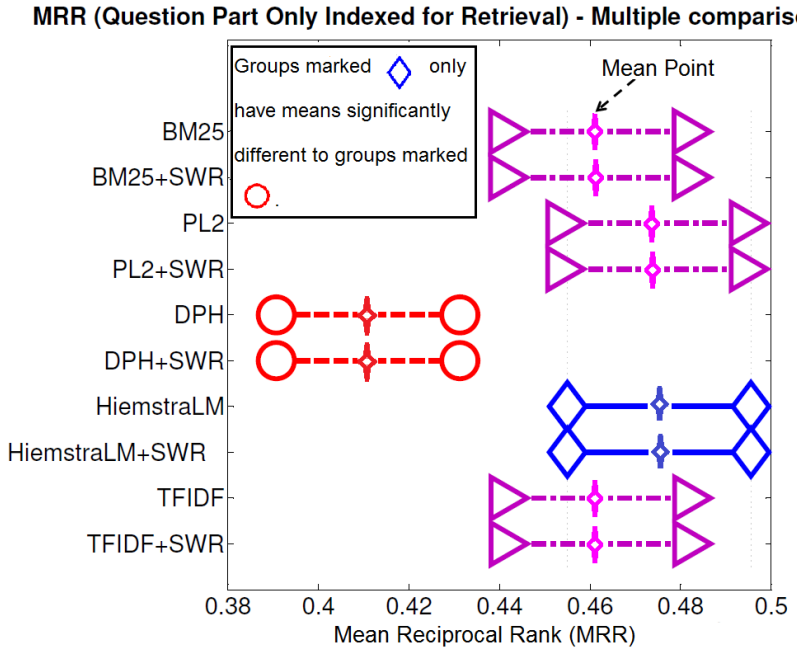


Figure 4.14: The confidence intervals of the MRR means for the 10 different test sets when only the question part is indexed for retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25+SWR (BM25 with stopword removal enabled, PL2+SWR (PL2 with stopword removal enabled).

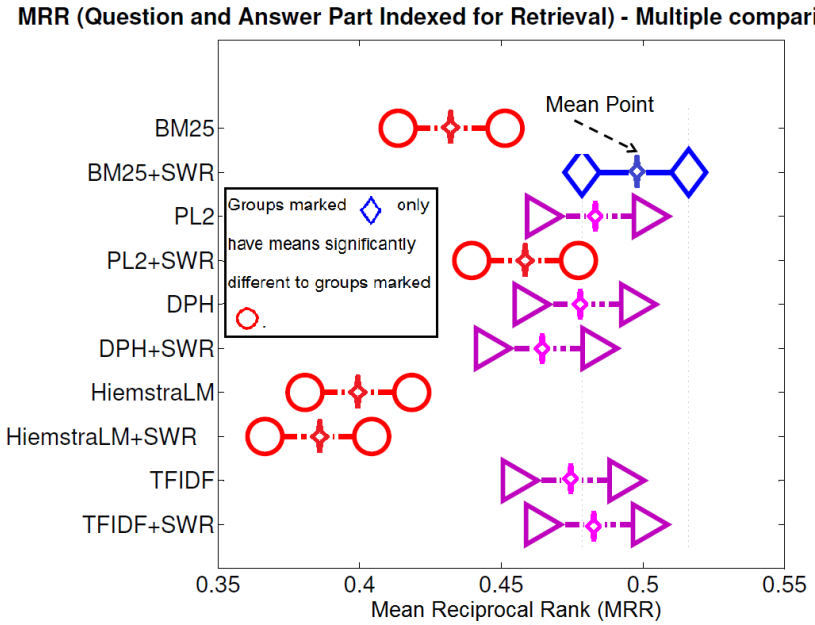


Figure 4.15: The confidence intervals of the MRR means for the 10 different test sets when both the question part and the answer part are indexed for retrieval. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$). BM25+SWR (BM25 with stopword removal enabled, PL2+SWR (PL2 with stopword removal enabled).

4.5.4 Summary

The purpose of the study in Section 4.5 was to determine the term weighing model to use in the baseline FAQ retrieval system for HIV/AIDS. BM25 outperformed all the other term weighing models across all the evaluation measures when stopwords are removed and the whole FAQ document is indexed for retrieval (see Table 4.2 and 4.3). The results of this study indicate that it is important to index the whole FAQ document in order to attain significant improvement in recall. This is mainly attributed to a reduction in term mismatch between the query and the relevant FAQ documents. Including the answer part in the index to be searched increases the vocabulary of the FAQ documents. Therefore, this increases the chances of a match between the SMS query tokens and the FAQ documents tokens.

4.6 Conclusion

The main goal of this chapter was to develop a suitable baseline system for our automated FAQ retrieval system. Four main building blocks were identified for our baseline system as described in Section 4.3. These are the data source (inverted index), the retrieval sub-system, the matching sub-system and the iterative interaction session manager. In Section 4.5, an empirical evaluation was conducted to determine the most appropriate way of representing the FAQ document in the data source. The results of this empirical evaluation suggest that indexing the whole FAQ document for retrieval improves the overall recall. This is very important in the iterative interaction retrieval strategy because users may decide to engage with the system longer to find other similar or relevant FAQ documents. The investigation in Section 4.5 also set out to determine the term weighting models to use in the matching sub-system. It was found that BM25 outperformed all the other term weighing models across all the evaluation measures when stopwords are removed and the whole FAQ document is indexed for retrieval.

Also in this chapter, we carried our an investigation to determine the number of iterations users are willing to tolerate before abandoning the iterative search process that we proposed in Section 4.3. The results of this investigation suggest that a majority of users can tolerate approximately 2 to 3 iterations before abandoning the search process. The bad abandonments statistics in this investigation were subsequently used to develop a novel evaluation metric for measuring user satisfaction. In essence, our new evaluation measure estimates the probability that any random user will be satisfied when using the FAQ retrieval system. As an additional finding, it was shown that the previous search experience of the users has a significant effect on their future behaviour.

In the next chapter, we describe experiments conducted to resolve the term mismatch prob-

lem that is inherent in FAQ retrieval systems as discussed in Section 2.2, (Chapter 2). For these experiments, the whole FAQ documents is indexed for retrieval since it was found from this chapter that it significantly improved recall. The BM25 term weighting model is used for the baseline system because it outperformed all the other term weighting models across all the evaluation measures when stopwords are removed and the whole FAQ document is indexed for retrieval. In addition, we create two other baseline systems that uses PL2 and DPH term weighting models because they performed better than the Hiemstra language model in our empirical evaluation. In our subsequent experiments, we do not use TF-IDF because it is similar to the BM25 term weighting model (Zhai, 2008).

Chapter 5

Resolving Term Mismatch for Search Length Reduction

5.1 Introduction

Previously in Chapter 1 (Section 1.3), we identified the FAQ document collection deficiencies as one of the main aspects to consider when developing automated FAQ retrieval systems. For example, the term mismatch problem between the users' queries and the relevant FAQ documents in the collection. In this chapter, our aim is to alleviate this term mismatch problem. In Table 5.1, we provide examples of the term mismatch problem between the users' queries and the relevant FAQ documents in the collection. Different approaches have been proposed in the literature for addressing this term mismatch problem in FAQ retrieval systems. For instance, in the literature review Chapter 2 (Section 2.2), we saw that these approaches can be characterised as statistical approaches, template-based approaches, Natural Language Processing (NLP) and ontology-based approaches (Sneiders, 2009). Indeed, we discussed how template-based approaches are more effective in resolving this term mismatch problem in small collections. By adding keywords and phrases to FAQ documents, previous studies have shown that the term mismatch problem can be markedly alleviated.

Another approach normally used to alleviate the term mismatch problem is Automatic Query Expansion (AQE). In AQE, the original query is expanded with other words that best capture the actual user intent, or that simply produce a query that is more likely to retrieve relevant documents (Carpineto and Romano, 2012). In Section 5.2, we discuss several query expansion approaches proposed in the literature.

As per our thesis statement, we propose to alleviate this term mismatch problem by enriching the FAQ documents with additional terms from a query log, which are added as a separate field in a field-based model. In our proposed FAQ document enrichment strategy,

Table 5.1: Examples of term mismatch problem between the users' queries and the relevant FAQ document in the collection.

Users' Queries	Relevant FAQ Document
Is an unborn baby at risk of contracting hiv? Does pregnant woman transmit aids to unborn baby?	How can you get infected with HIV? The main ways in which you can get infected with HIV are- By having oral or penetrative sex without a condom. blood-to-blood contact i.e. by sharing sharp objects like razor blades or needles with an infected person, or by coming into contact with an HIV-positive persons blood, through sores or cuts on your body. **from an HIV-positive mother to her child, either in the womb, when giving birth or through breast-feeding.** By way of blood transfusion. However, in Botswana, this risk is low as all blood donations are tested for HIV.
How does hiv/aids affect the person health? How does HIV/AIDS affect people?	What happens after you become infected with HIV? Our bodies are protected from diseases by our immune system. After you become infected with HIV, the virus gradually multiplies inside the body and eventually destroys the body's immune system. This means that the body will not be able to fight off diseases.
How fast can the HIV/AIDS virus weaken the human body? Does hiv virus differ from person to person?	How long does it take for HIV to cause AIDS? There is no set time period, however, about half of the people with HIV develop AIDS within six to ten years after becoming infected with the virus. The onset of AIDS depends on various factors- -how strong your immune system is -your lifestyle (what you eat, how much you exercise and rest, whether you drink alcohol or smoke, etc. -early treatment or prevention of some of the diseases associated with HIV - whether a specially trained AIDS doctor has prescribed you specific medicines that slow down the disease progression by suppressing the virus in your body.

the FAQ documents are enriched with additional terms from the SMS queries for which the true relevant FAQ documents are known. We will show that if this term mismatch problem is alleviated, there will be marked improvement in recall and the probability that any random user will be satisfied. Furthermore, we investigate our proposed approach using two different enrichment strategies. In particular, the Term Occurrence and the Term Frequency enrichment strategies. In Section 5.3, we describe in detail these two different enrichment strategies. The following research questions were identified to help us in our investigation:

Chapter 5-Research Question One (C5-RQ1): Can we improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries for which the true relevant FAQ document are known.

Chapter 5-Research Question Two (C5-RQ2): Can we improve the overall recall and the probability that any random user will be satisfied by taking into consideration the number of times a term occurs in the queries when enriching the FAQ documents.

Chapter 5-Research Question Three (C5-RQ3): Does increasing the number of queries used in enriching the FAQ documents increase the overall recall and the probability that any random user will be satisfied.

Chapter 5-Research Question Four (C5-RQ4): Does the proposed enrichment strategies pro-

duce similar results when deployed with different field-based term weighting models.

The remainder of this chapter is organised as follows:

- Section 5.3 describes our proposed FAQ documents enrichment strategies. In particular, the Term Occurrence and the Term Frequency enrichment strategies.
- In Section 5.4, we provide background information on the field-based term weighting models that we use to evaluate our proposed enrichment strategies.
- In Section 5.5, we describe how we investigate and evaluate our proposed enrichment strategies together with our baseline systems. We also describe the dataset that we use to evaluate our enrichment strategies.
- Section 5.6 presents our results and analysis. This is followed by the discussion and conclusions in Section 5.7.

5.2 Automatic Query Expansion Techniques

Automatic query expansion techniques can be classified into the following five main groups: Linguistic methods, corpus-specific statistical approaches, query-specific statistical approaches, search log analysis and web data (Carpineto and Romano, 2012). These techniques are classified based on the conceptual paradigm used for generating the expansion terms. Figure 5.1 shows a general taxonomy of approaches of query expansion after splitting each groups into a few sub classes. The following is a brief description of each method:

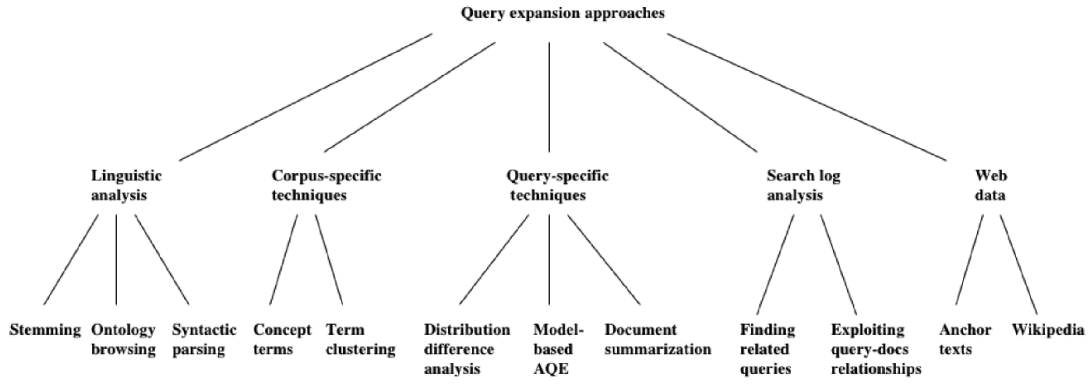


Figure 5.1: A taxanomy of query expansion approaches (Carpineto and Romano, 2012).

- Linguistic Analysis: These are techniques that use external resources such as WordNet, dictionaries and thesauri to generate lexical, syntactic and semantic word relationships to expand or reformulate query terms. Voorhees (1994b) showed that expanding

queries with concepts from WordNet makes little difference in retrieval effectiveness if the original queries are relatively complete descriptions of the information being sought even when the concepts are selected manually. Furthermore, she demonstrated that the retrieval effectiveness of less well developed queries can be significantly improved only if the expansion terms are selected manually. In general, linguistic techniques are often considered less effective than those based on statistical analysis of the document collection or analysis of the top-ranked documents for a given query to discover word relationships (Carpineto and Romano, 2012).

- Corpus-Specific Global Techniques: These are data driven techniques that rely on the statistical relationship between term pairs in a document collection to identify expansion terms. In particular, early techniques used co-occurrence data to identify expansion terms (Minker et al., 1972, Smeaton and van Rijsbergen, 1983). However, this approach has not been very successful (Peat and Willett, 1991). Several authors have reported a degradation in the retrieval effectiveness when the original query terms are expanded with terms that co-occur with the original query terms in the document collection (Minker et al., 1972, Smeaton and van Rijsbergen, 1983). Other corpus-specific approaches that have been found to improve the retrieval effectiveness used techniques such context vectors (Gauch et al., 1999), mutual information (Hu et al., 2006), latent semantic indexing (Park and Ramamohanarao, 2007) and interlinked Wikipedia articles (Milne et al., 2007) to automatically build a similarity thesaurus to aid in expanding the original query. A similarity thesaurus is a matrix that consists of term-term similarities (Qiu and Frei, 1993).

- Query-Specific Local Techniques: These techniques have been found to be more effective than corpus-specific techniques because they generate expansion terms based on the local context provided by the query (Xu and Bruce, 2000). Typically, they generate the expansion terms from the top k retrieved documents. For example, Attar and Fraenkel (1977) deployed a term clustering algorithm on the top retrieved documents for a given query. These term clusters were then used to expand the original query. Several other query-specific methods for generating expansion terms have been proposed, such as local context analysis, which selects expansion terms based on co-occurrence with the query terms within the top-ranked documents (Xu and Bruce, 2000), passage extraction (Xu and Bruce, 1996) and query-expansion using document summaries (Lam-Adesina and Jones, 2001).

- Search Log Analysis: These are query expansion techniques that use external evidence that has been implicitly suggested by Web users to expand the original query (Carpineto and Romano, 2012). There are two main approaches of automatic query expansion based on search logs. In the first approach, the expansion terms are extracted from

external web documents that are related to the original query. In particular, Yin et al. (2009) used two types of external evidence (query logs and snippets) for query expansion. In their evaluation, they found that snippet-based expansion, using the summaries provided by an external search engine, provides significant improvement in retrieval performance. The second approach exploits the relation of queries to retrieved documents to generate expansion terms (Xue et al., 2004). Example of such approaches include but are not limited to extracting terms directly from clicked results or finding queries associated with the same clicks. These methods however, do not perform well for all types of users and search tasks. Some of the problems associated with their use in automatic query expansion include noise, incompleteness, sparseness, and the volatility of Web pages and query (Xue et al., 2004).

- Web Data: Anchor text extracted from search engine logs are another source of data for query expansion (Arguello et al., 2008, Kraft and Zien, 2004). Anchor text are commonly used for AQE because they provide a summary of the destination page (Carpineto and Romano, 2012). Several methods have been proposed for selecting candidate anchor text for query expansion. In particular, Arguello et al. (2008) proposed a method that selects the set of candidates associated with a query by considering only those anchor texts that point to a short set of top-ranked documents from a large set of top-ranked documents. Although the methods that use web data have shown significant improvement in retrieval performance when deployed in an IR system, these methods have not yet been compared with others on a standard collection (Carpineto and Romano, 2012).

Later in Section 5.5.5, we compare our proposed FAQ documents enrichment strategies with the query-specific local techniques because they have been found to be more effective than other query expansion techniques (Carpineto and Romano, 2012).

5.3 FAQ Documents Enrichment Strategies

In Web IR, there is the notion of document fields and this provides a way to incorporate the structure of a document in the retrieval process (Robertson et al., 2004). For example, the contents of different HTML tags (e.g anchor text, title, body) are often used to represent different document fields (Plachouras and Ounis, 2007, Robertson et al., 2004). Earlier work by Macdonald et al. (2006) has shown that combining evidence from different fields in Web retrieval improves the retrieval performance. In this thesis, we propose a new structure for our FAQ documents, where the contents of each FAQ document is separated into different fields. Within this new structure, each FAQ document is divided into two fields, a *QUESTION* and an *ANSWER* field. A third field is introduced, the *FAQLog* field. This newly

introduced field is used for adding additional terms from the SMS queries for which the true relevant FAQ documents are known. Our aim is to incorporate evidence (term frequencies) from these three fields during the retrieval process. The intuition is that, additional terms from previous searches, which are added into the *FAQLog* field will help to alleviate the term mismatch problem.

In order to answer research question *C5-RQ2* we investigate our proposed approach using two different enrichment strategies. First, the FAQ documents will be enriched using all the terms from a query log. In this approach, all the queries from the training set for which the true relevant FAQ documents are known will be added into the newly introduced *FAQLog* field as shown in Table 5.2. In other words, if an FAQ document is known to be relevant to a query, then this query is added to its *FAQLog* field. For the remainder of this chapter, this approach will be referred to as the Term Frequency approach. In the second approach, the FAQ documents will be enriched using term occurrences from a query log. Here, all the unique terms from the training set for which the true relevant FAQ documents are known will be added to the *FAQLog* field as shown in Table 5.3. In other words, only new query terms that do not appear in the *FAQLog* field will be added to that field. For the remainder of this chapter, this approach will be referred to as the Term Occurrence approach.

Table 5.2: Enrichment Using Query Term Frequencies. All the queries from the training set for which the true relevant FAQ documents are known are added into the newly introduced *FAQLog* field.

FIELDS	**CONTENTS of FIELDS**
QUESTION	Does HIV / AIDS affect women differently from men?
ANSWER	No, the virus affects both men and women in exactly the same way i.e. by making the immune system weak, so that it cannot fight off other illnesses.
FAQLog	Is hiv/aids gender based to some extent? between men and women, who are most infected by hiv/aids? who are mainly infected male or female? which gender is mostly affected by the disease?

Table 5.3: Enrichment Using Query Term Occurrence. All the unique terms from the training set for which the true relevant FAQ documents are known will be added to the *FAQLog* field.

FIELDS	**CONTENTS of FIELDS**
QUESTION	Does HIV / AIDS affect women differently from men?
ANSWER	No, the virus affects both men and women in exactly the same way i.e. by making the immune system weak, so that it cannot fight off other illnesses.
FAQLog	is, hiv, aids, gender, based, to, some, extent, between, men, and, women, who, are, most, infected, by, mainly, male, or, female, which, mostly, affected, the, disease

The main difference between the two enrichment strategies is that the Term Frequency approach captures the frequencies with which users use some terms to retrieve some FAQ documents. Hence, some query terms will have a higher term frequency in the *FAQLog* if they were used often by the users. This information can be very useful as it can be used to measure how important a query term is to an FAQ document. In Table 5.2 and Table 5.3, we illustrate the differences between the two enrichment strategies. For example, under the Term Frequency approach (Table 5.2), the term frequencies of the terms *gender* and *infected* in the *FAQLog* field are: *gender=2* and *infected=2*. Under the Term Occurrence approach (Table 5.3) the term frequencies of these terms are 1 because the query terms under this approach can only be added to this field once even if they appear in many queries. Since field-based models rely on term frequencies to calculate the final retrieval score of a relevant document given a query, the two enrichment strategies will always give different retrieval scores. In Section 4.5.2, we investigate the usefulness of each enrichment strategy.

In order for us to be able to answer research question *C5-RQ4*, we use two widely used field-based term weighting models namely, PL2F (Macdonald et al., 2006) and BM25F (Robertson et al., 2004) to evaluate the proposed enrichment strategies. In addition, a third non-parametric document weighting model (DPH) (Amati et al., 2007) will be extended to handle fields (DPHF). This new DPHF will also be used to evaluate the proposed FAQ documents enrichment strategies.

5.4 Background Information on Field-Based Term Weighting Models

In this section, we provide details of the field-based models used to evaluate the proposed enrichment strategies. We start by describing the BM25F field-based weighting model in Section 5.4.1. This is followed by a description of the PL2F field-based weighing model in Section 5.4.2 and DPHF in Section 5.4.3.

5.4.1 The BM25F Weighting Model

BM25F (Robertson et al., 2004) is an extension of the traditional BM25 weighting model that incorporates the structure of a document in scoring. The relevance score of a document d for a given query Q based on the BM25 weighting model is expressed as (Robertson et al., 1996):

$$score_{BM25}(d,Q) = \sum_{t \in Q} w^{(1)} \frac{(k_1+1)tfn}{k_1+tfn} \cdot \frac{(k_3+1)qtf}{k_3+qtf} \tag{5.1}$$

where qtf is the number of occurrences of a given term in the query Q. k_1 and k_3 are parameters of the model with default settings of 1.2 and 1000 respectively. $w^{(1)}$ denotes the idf factor and is given by:

$$w^{(1)} = \log \frac{N - dft + 0.5}{dft + 0.5} \tag{5.2}$$

Where N is the number of documents in the collection and dft is the number of documents in the collection that have a term t. On the other hand, the normalised within document term frequency tfn of the BM25 weighting model in Equation (5.1) is given by:

$$tfn = \frac{tf}{(1-b) + b \cdot \frac{l}{avg\_l}} \tag{5.3}$$

Where tf is the frequency of the query term in the document d and b is the term frequency normalisation hyper-parameter. l is the length of the document d and avg_l is the average document length in the collection.

The BM25F weighting model on the other hand normalises the term frequencies on a per-field basis (Zaragoza et al., 2004). This per-field normalisation applies a linear combination of the normalised term frequencies from different fields based on Equation (5.3) as follows:

$$tfn = \sum_{f} w_f \cdot \frac{tf_f}{(1-b_f) + b_f \cdot \frac{l_f}{avg\_l_f}} \tag{5.4}$$

where w_f is the weight for the field f, tf_f is the frequency of the query term in f^{th} field. l_f represents the number of tokens in the f^{th} field while avg_l_f represents the average length of the f^{th} in the collection . b_f is the term frequency normalisation hyper-parameter of the f^{th} field. The BM25F model proposed by Zaragoza et al. (2004) yields different retrieval scores compared to the BM25F model proposed by Robertson et al. (2004). This is because, in their proposed BM25F model, Robertson et al. (2004) do not normalise term frequencies in a per-field manner.

5.4.2 The DFR PL2F Weighting Model

PL2F is a per-field derivative of the PL2 Divergence from Randomness (DFR) model that applies term frequency normalisation and weighting for a number of different fields in a document (Plachouras and Ounis, 2007). The relevance score of a document d for a given query Q based on the PL2 weighting model is expressed as follows:

$$score_{PL2}(d,Q) = \sum_{t \in Q} \frac{qtfn}{tfn+1} \left(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn) \right) \tag{5.5}$$

where $score(d,Q)$ is the relevance score of a document d for a given query Q. $\lambda = \frac{tfc}{N}$ is the mean and variance of a Poisson distribution, tfc is the frequency of the term t in the collection C while N is the number of documents in the collection. The normalised query term frequency is given by $qtfn = \frac{qtf}{qtf_{max}}$, where qtf_{max} is the maximum query term frequency among the query terms and qtf is the query term frequency. tfn is the Normalisation 2 of

the term frequency tf of the term t in a document d and is expressed as:

$$tfn = tf \cdot \log_2 \left(1 + b\frac{avg\_l}{l}\right), (b > 0) \tag{5.6}$$

In the above expression, l is the length of the document d, avg_l is the average document length in the collection and b is a hyper-parameter. In documents that have fields, the per-field Normalisation 2F extends Normalisation 2 in Equation (5.6) so that tfn becomes a linear combination of the field weights and the normalised term frequency in each field f. This is expressed as:

$$tfn = \sum_f w_f \cdot tfn_f \tag{5.7}$$

where w_f is the weight of the f^{th} field and tf_{nf} represents the normalised term frequency of the query term in the f^{th} field. This is given by Normalisation 2F:

$$tfn_f = tf_f \cdot \log_2 \left(1 + b_f\frac{avg\_l_f}{l_f}\right), (b_f > 0) \tag{5.8}$$

where l_f and avg_l_f are the field length and the average field of the f^{th} field respectively. b_f is a hyper-parameter for each field f and tf_f is the frequency of the query term in the f^{th} field. It is important to note that Normalisation 2 is only used when the whole document is considered as one field while Normalisation 2F applies when the document is divided into fields. Based on the above expressions, the PL2 model from Equation (5.5) can be extended to the PL2F model by substituting tfn with Equation (5.7).

5.4.3 The DFR DPHF Weighting Model

In this thesis, we also extend the parameter-free DPH term weighting model from the Divergence from Randomness (DFR) framework (Amati et al., 2007) so that it can handle fields. We call the new model DPHF. The DPH term weighting model calculates the score of a document d for a given query Q as follows:

$$score_{DPH}(d, Q) = \sum_{t \in Q} qtf \cdot norm \cdot \left(tf \cdot \log((tf \cdot \frac{avg\_l}{l}) \cdot (\frac{N}{tfc})) + 0.5 \cdot \log(2 \cdot \pi \cdot tf \cdot (1 - t_{MLE}))\right) \tag{5.9}$$

where qtf, tf and tfc are the frequencies of the term t in the query Q , in the document d and in the collection C respectively. N is number of documents in the collection C, avg_l is the average length of documents in the collection C and l is the length of the document d. $t_{MLE} = \frac{tf}{l}$ and $norm = \frac{(1-t_{MLE})^2}{tf+1}$.

In this thesis, we adapt the DPH term weighting model to field-based retrieval by following the approach in He and Ounis (2007). In their approach, He and Ounis (2007) extend the

DLH Hyper-Geometric DFR Model using the Laplace Normalization (DLH) (Amati, 2006) term weighting model to handle fields by directly combining the term frequencies in the different fields without normalisation. They called this field-based model DLHF. Hence, in our newly derived DPHF field-based weighting model, the term frequency tf of the whole document will be a linear combination of the term frequencies in the different fields (He and Ounis, 2007). This linear combination of term frequencies is expressed as:

$$tf = \sum_{f} w_f \cdot tf_f \tag{5.10}$$

In the above expression, w_f is the weight of the f^{th} field and tf_f is the frequency of the query term in the f^{th} field of the document. Hence, the DPH weighting model can be adapted to field-based retrieval (DPHF) by substituting the term frequency in the document tf in Equation (5.9) with Equation (5.10).

5.5 Experimental Investigation and our Baseline Systems

In this section, we are investigating whether we can alleviate the term mismatch problem between the users' queries and the relevant FAQ documents in the collection by enriching the FAQ documents with additional terms from a query log. First, in Section 5.5.1 we provide a description of how we created the training and testing set for our evaluation. In Section 5.5.2, we describe how we enrich the FAQ documents in the collection. Section 5.5.3 provides the detail of our experimental setting. This is followed by a description of how the field weights for the field-based term weighting models were optimised in Section 5.5.4. In Section 5.5.5, we outline a series of experiments conducted to answer research questions listed in Section 5.1.

5.5.1 Creating the Training and Testing Sets

We used the 10 different training and testing sets that we created as described in Section 3.3.1 (Chapter 3) to investigate our proposed enrichment strategies. As outlined in Chapter 3, we produced 10 random splits of the 750 matched SMS queries into training set of 600 queries and testing set of 150 queries. These SMS queries were manually corrected for spelling errors so that such a confounding variable does not influence the outcome of these experiments.

5.5.2 FAQ Documents Enrichment

The main contributions of this chapter as described in Section 5.1 is to demonstrate that we can improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries for which the true relevant FAQ document are known (*C5-RQ1*). To answer research question *C5-RQ3* (does increasing the size of the training set improve retrieval performance), we additionally split the 600 training queries into three sets of 200 and incrementally combined them to create training sets of size 200, 400 and 600 queries (hereafter referred to as 200SMSes, 400SMSes and 600SMSes). 400SMSes is therefore a superset of 200SMSes and 600SMSes is a superset of 400SMSes. This process was chosen as it emulates the temporal nature of query collection in a real system. For each train/test split, we created 6 (3 for term frequencies and the other 3 for term occurrences) enriched collections (corresponding to 200SMSes, 400SMSes and 600SMSes) using the two enrichment approaches described in Section. 5.3. In total, we created 60 different enriched FAQ document collections.

In order to infer whether using field-based weighting models does indeed help in the overall retrieval performance in terms of recall and the probability that any random user will be satisfied, the weights for each field were optimised on the training set as shown in Figure 5.2. Optimisation of these field weights is vital as significant gains in relevance can be obtained if the parameters are properly optimised (Robertson and Zaragoza, 2009, Robertson et al., 2004). We used the 10 random splits of the 600 SMS queries of training data for optimising the field weights. The test queries for each train/test split were naturally not used for optimisation of the field weights in order to avoid over-fitting. For each training set, we randomly selected 450 SMS queries and used these to enrich the FAQ documents using our two enrichment strategies proposed in Section 5.3, thus giving us 2 different enriched FAQ document collections for each training set. The remaining 150 SMS queries were left for optimising the field weights. In the next section, we provide our experimental setting.

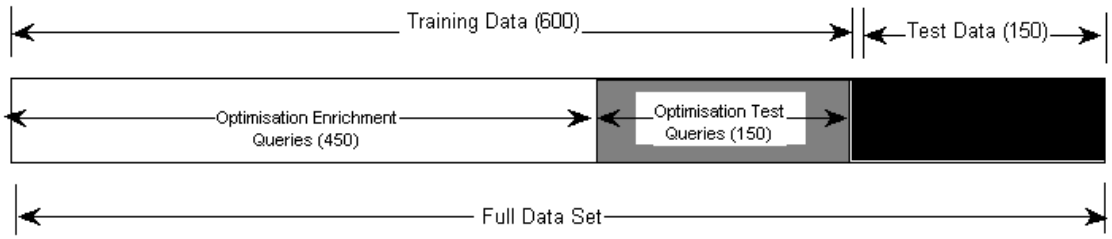


Figure 5.2: Training and testing sets

5.5.3 Experimental setting

For all our experimental evaluation, we used Terrier-3.5[25] (Ounis et al., 2005), an open source IR platform. All the FAQ documents used in this study were first pre-processed before indexing and this involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). We have seen in Chapter 4 that BM25 performs poorly when stopword removal is not enabled compared to other term weighting models. Therefore, in all our experiments, we enabled stopword removal by ignoring the terms that had low IDF when scoring the documents. It was also discovered in Chapter 4 that BM25 yields the best retrieval performance when there is no length normalisation ($b = 0.0$). Hence, in these experiments the normalisation parameter for BM25 was set to 0.0. The same normalisation parameter settings were used for each field (*QUESTION*, *ANSWER* and *FAQLog* fields) when BM25F was deployed. For PL2, the normalisation parameter was set to its default value of $c = 1.0$. Similarly, the normalisation parameter settings for each field was set to 1.0 for the *QUESTION*, *ANSWER* and *FAQLog* fields when PL2F was deployed.

5.5.4 Optimisation of Field Weights

In optimising the field weights, we used the Terrier-3.5 Information Retrieval (IR) platform. First we indexed the enriched collections separately without stopword removal and using the full Porter stemming algorithm. We then performed our optimisation using the Robust Line Search (RLS) strategy as described in (Robertson and Zaragoza, 2009).

For both BM25F, DPHF and PL2F, we performed an initial scan of the field weights parameters w_Q, w_A and w_{QL} (*QUESTION*, *ANSWER* and *FAQLog* fields respectively) to determine the optimal values of these field weights with respect to a higher Mean Average Precision (MAP). In our initial scan, the field weights were varied linearly from 0.0 to 10.0 in steps of 1. Higher MAP values for the first scan were obtained when the *ANSWER* field was set to 1 for most of the collections. For the *QUESTION* and *FAQLog* fields, higher MAP values were obtained when these fields were set to 2 or higher.

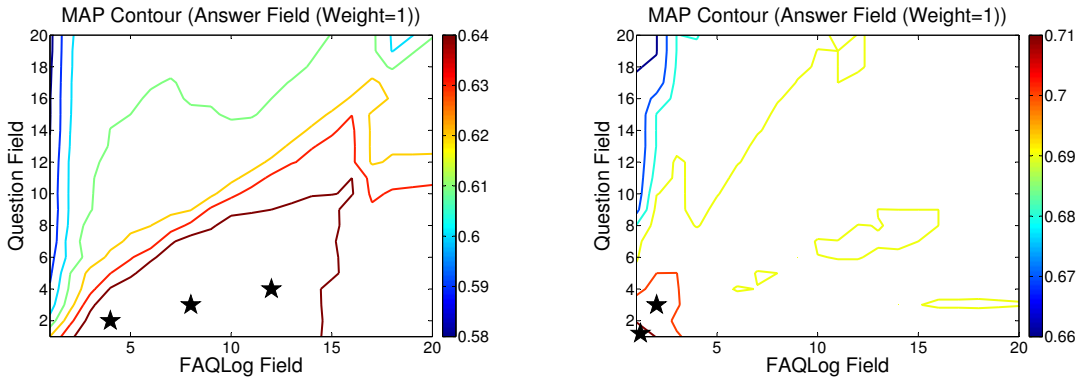
We then set a second starting point for each field weight to $(w_Q = 1.0, w_A = 1.0, w_{QL} = 1.0)$. Because the optimal value of the *ANSWER* field was 1, this field was fixed while the others were varied linearly from 1.0 to 21.0 in steps of 1.0 for the second RLS. The above procedure was repeated for all the 10 random splits of training data. The optimal values of the field weights for these 10 random splits of training data were averaged to arrive at the final values of the field weights to use in testing our enrichments strategies.

Table 5.4 shows the mean and standard deviation of the field weights that we will use in our experimental investigation. It is worth pointing out that these values were averaged taking into consideration that small changes in the parameter values of these models are

Table 5.4: The mean and the standard deviation for the *QUESTION* and *FAQLog* field weights. The *ANSWER* field weight (w_A) was set to 1.0.

Weighting Model	Enrichment Strategy	Mean Field Weights	Standard Deviation
PL2F	Term Occurrence	$w_Q = 14.9, w_{QL} = 14.1$	$stdv_Q = \pm 4.63, stdv_{QL} = \pm 5.05$
	Term Frequency	$w_Q = 1.8, w_{QL} = 16.1$	$stdv_Q = \pm 0.79, stdv_{QL} = \pm 1.66$
BM25F	Term Occurrence	$w_Q = 2.0, w_{QL} = 7.4$	$stdv_Q = \pm 1.05, stdv_{QL} = \pm 4.47$
	Term Frequency	$w_Q = 2.8, w_{QL} = 2.0$	$stdv_Q = \pm 1.30, stdv_{QL} = \pm 0.82$
DPHF	Term Occurrence	$w_Q = 9.3, w_{QL} = 3.9$	$stdv_Q = \pm 1.88, stdv_{QL} = \pm 0.73$
	Term Frequency	$w_Q = 7.3, w_{QL} = 2.8$	$stdv_Q = \pm 7.75, stdv_{QL} = \pm 0.42$

known to produce small changes in the accuracy of relevance (Robertson and Zaragoza, 2009). Our analysis of the various contour plots also show that the mean field weights in Table 5.4 are also within the region of higher MAP values as denoted by ★ in Figure 5.1(a). Since higher MAP values were obtained when the *ANSWER* field was fixed at 1, this contour plots only shows the changes in MAP when the *QUESTION* and the *FAQLog* field are varied. Similarly, the regions of higher MAP values are denoted by ★ in Figure 5.1(b), 5.2(a), 5.2(b) 5.3(a) and 5.3(b), for all the training samples.



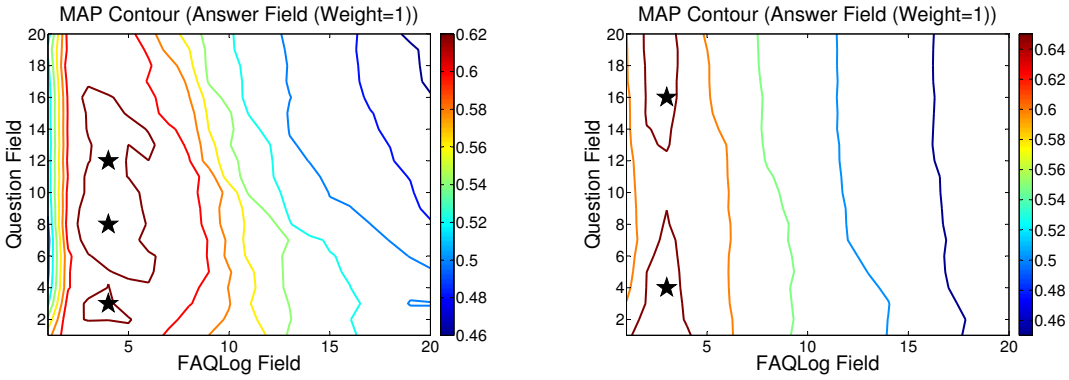
(a) BM25F Term Occurrence

(b) BM25F Term Frequency

Figure 5.3: The ★ denotes the regions with the highest MAP when the answer field w_A is fixed at 1 for BM25F term occurrence and term frequency enrichment strategies.

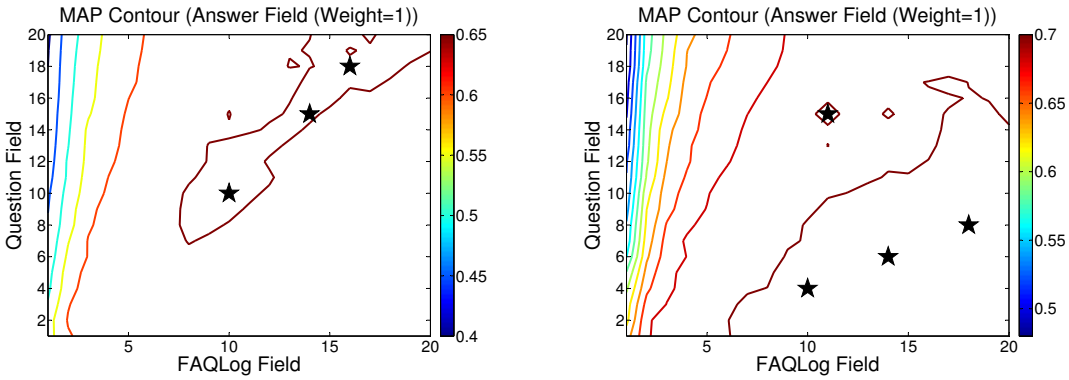
5.5.5 Experimental Outline

EXV1: In this experiment, the proposed enrichment strategies are tested. In particular, we investigate whether we can improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries for which the true relevant FAQ document are known (*C5-RQ1*). A description of how the FAQ documents were enriched using the training set is provided in Section 5.5.2. To carry out this investigation, we used the retrieval settings described in Section 5.5.3. First, the enriched FAQ document collections were indexed using fields so that field-based weighting models



(a) DPHF Term Occurrence (b) DPHF Term Frequency

Figure 5.4: The ★ denotes the regions with the highest MAP when the answer field w_A is fixed at 1 for DPHF term occurrence and term frequency enrichment strategies.



(a) PL2F Term Occurrence (b) PL2F Term Frequency

Figure 5.5: The ★ denotes the regions with the highest MAP when the answer field w_A is fixed at 1 for PL2F term occurrence and term frequency enrichment strategies.

such as BM25F (Robertson et al., 2004), PL2F (Macdonald et al., 2006) and our newly derived DPHF could be used. For each index containing the enriched FAQ documents, we used the associated testing sets to perform retrieval using three different field-based weighting models (BM25F, PL2F and DPHF). For this investigation, all the field weights parameters were intentionally set to 1 ($w_Q = 1, w_A = 1, w_{QL} = 1$), where ($w_Q, w_A$ and w_{QL}) represents the *QUESTION*, *ANSWER* and *FAQLog* field weights respectively. As a baseline, we also created an inverted index with the non-enriched FAQ documents. We then used the 10 different testing sets to perform retrieval using BM25, PL2 and DPH.

EXV2: In this experiment, we investigate whether we can do better by optimising the field weights for the enriched FAQ documents collections. It is well known that significant gain in relevance can be obtained if the field weight parameters are properly optimised (Robertson and Zaragoza, 2009, Robertson et al., 2004). In our investigation, we use *EXV1* as our baseline systems. We then optimise the field weights for all the enriched collections. A

Table 5.5: Examples of some of the web pages that were crawled from the web to use as an external collection in our collection enrichment approach.

Web Page	Uniform Resource Locator (URL)
Avert : AVERTing HIV and AIDS	$http://www.avert.org$
FAQ |AIDS Foundation of South Africa	$http://www.aids.org.za$
What everyone should know about HIV	$http://www.hivaware.org.uk$
AIDS$_\star$gov	$http://www.aids.gov$

description of how the field weights were optimised can be found in Section 5.5.4. We then perform retrieval on these enriched FAQ document collections using the associated testing set with the field weights for BM25F, PL2F and DPHF set to their new optimal values.

EXV3: In experiments *EXV1* and *EXV2* we also investigate the effect of changing the size of the training set (*C5-RQ3*). In carrying out these experiments, three different collections that were enriched with queries of varying sizes were used for each testing set. A description of how these collections were created is provided in Section 5.5.2.

EXV4: To compare our approach with traditional approaches normally used to resolve the term mismatch problem, we used the collection enrichment approach first introduced by Kwok and Chan (1998). In collection enrichment, a high quality external collection is used to expand the original query terms and then retrieves from the local collection using the expanded query (Kwok and Chan, 1998). A local collection refers to the collection from which the final retrieved documents are retrieved. In the collection enrichment approach, we first performed retrieval on an external collection of HIV/AIDS documents, which were crawled from the web on the 28^{th} of January 2013. We crawled web pages that have a strong focus on HIV/AIDS frequently asked questions. Each web page crawled was indexed as a single document. In total, we had 3648 web page documents. For example, from $www.avert.org$, we were able to crawl 259 web documents. We provide examples of some of the domains and pages crawled in Table 5.5. In our collection enrichment approach, we used the Terrier-3.5 Divergence From Randomness (DFR) Bose-Einstein 1 (Bo1) model to select the 10 most informative terms from the top 3 returned documents as expansion terms. These 10 new terms together with the original query terms were used for retrieval on the non enriched FAQ documents collection. The DFR Bo1 model calculates the weight of a term t in the top-ranked documents as follows:

$$w(t) = tfx \cdot \log_2 \frac{1 + P_n(t)}{P_n(t)} + \log_2(1 + P_n(t)) \tag{5.11}$$

$$P_n(t) = \frac{tfc}{N} \tag{5.12}$$

where tfx is the frequency of the query term in the top x ranked documents, tfc is the

frequency of the term t in the collection, and N is the number of documents in the collection. In the next section, we present an analysis of our experimental results.

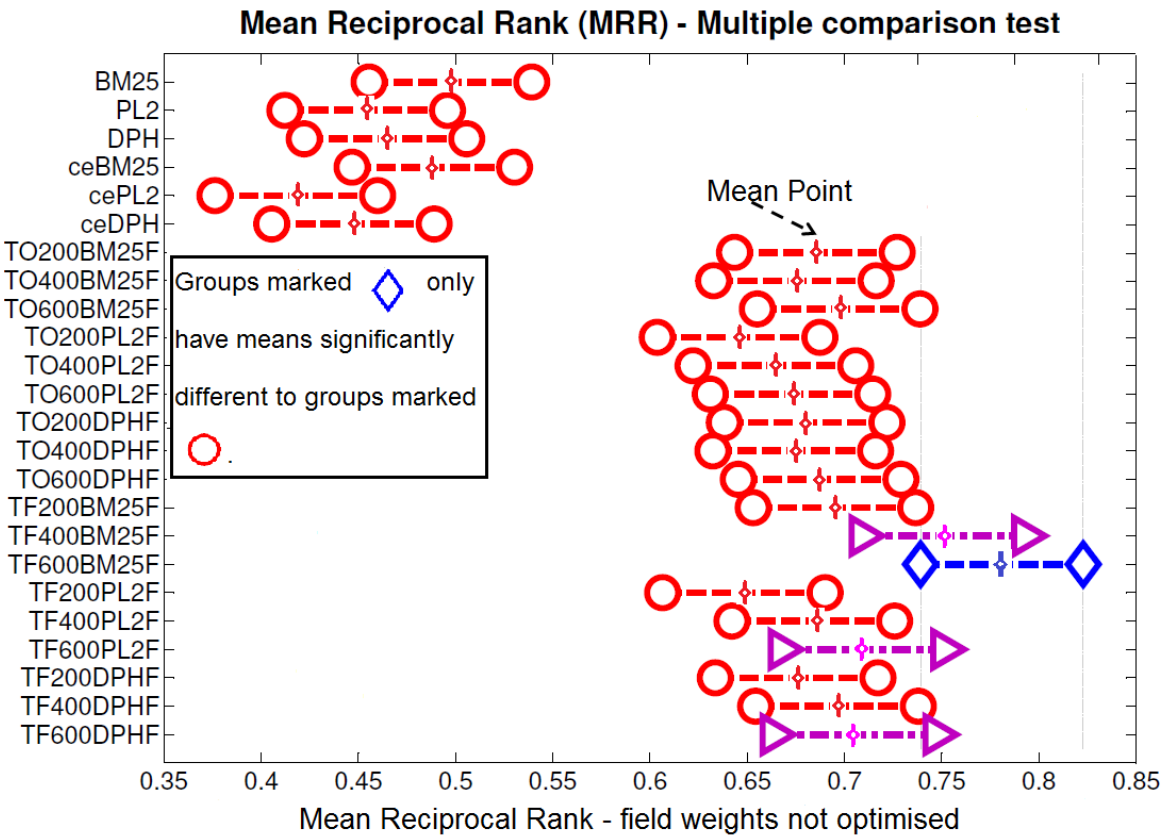
5.6 Experimental Results

In this section, we report on whether we can alleviate the term mismatch problem between the users' queries and the relevant FAQ documents in the collection by enriching the FAQ documents with additional terms from a query log. In particular, in Section 5.6.1, we report on the retrieval effectiveness of our enrichment strategies in terms of MRR, MAP, recall and the probability that any random user will be satisfied when the field weights were not optimised. In Section 5.6.2, we examine the effect of optimising the field weights on the retrieval effectiveness.

5.6.1 Field Weights not Optimised

The first set of our analyses examines the effect of enriching the FAQ documents with additional terms from queries for which the true relevant FAQ documents are known (*C5-RQ1*). To answer this research question, a Multiple Comparison test was used. Our results in Figure 5.6 suggest a significant improvement in the retrieval performance when the FAQ documents are enriched. In this figure, there is no significant difference in the retrieval performance when the confidence intervals between two groups overlap. For example, there is no significant difference in retrieval performance between the groups denoted by $\diamondsuit$ and $\triangleright$. However, when two confidence intervals do not overlap, it means that there is a significant difference in retrieval performance. For example, there is a significant difference in retrieval performance between the groups denoted by $\diamondsuit$ and $\circ$ (Multiple comparison test, $p < 0.05$). Similar results were observed for other evaluation measures as shown in Figures 5.7, 5.8 and 5.9. There was a significant increase (Multiple Comparison test, $p < 0.05$) in recall from around 0.4182 for the non enriched FAQ documents to more than 0.6800 for the enriched FAQ documents (Table 5.6). An increase in recall implies a reduction in term mismatch because previously non-retrieved documents have been retrieved.

Moreover, higher MRR values were obtained when enriching the FAQ documents using the query term frequencies rather than the query term occurrence (*C5-RQ2*). These findings suggests that it is important to take into consideration the number of times a term occurs in the queries when enriching the FAQ documents. An increase in the size of the collection used to enrich the FAQ documents resulted in a slight increase in the average MRR (averaged across the 10 train/test partitions) for both PL2F, DPHF and BM25F (*EXV3*). However, only the increase from 200 to 400 and 200 to 600 training SMS queries was statistically significant



ceBM25	collection enrichment using BM25	cePL2	collection enrichment using PL2
ceDPH	collection enrichment using DPH	TO200BM25F	term occurrence, 200 sms queries using BM25F
TO400BM25F	term occurrence, 400 sms queries using BM25F	TO600BM25F	term occurrence, 600 sms queries using BM25F
TO200PL2F	term occurrence, 200 sms queries using PL2F	TO400PL2F	term occurrence, 400 sms queries using PL2F
TO600PL2F	term occurrence, 600 sms queries using PL2F	TO200DPHF	term occurrence, 200 sms queries using DPHF
TO400DPHF	term occurrence, 400 sms queries using DPHF	TO600DPHF	term occurrence, 600 sms queries using DPHF
TF200BM25F	term frequency, 200 sms queries using BM25F	TF400BM25F	term frequency, 400 sms queries using BM25F
TF600BM25F	term frequency, 600 sms queries using BM25F	TF200PL2F	term frequency, 200 sms queries using PL2F
TF400PL2F	term frequency, 400 sms queries using PL2F	TF600PL2F	term frequency, 600 sms queries using PL2F
TF200DPHF	term frequency, 200 sms queries using DPHF	TF400DPHF	term frequency, 400 sms queries using DPHF
TF600DPHF	term frequency, 600 sms queries using DPHF		

Figure 5.6: The confidence intervals for the MRR means of the different groups when the field weights are not optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

(Multiple Comparison test, $p < 0.05$), suggesting that adding more training SMS queries in the new field does indeed help to alleviate the term mismatch problem. We also evaluated the different systems using the bad abandonment statistics from Chapter 4 and found that there was no significant increase in the probability any random user will be satisfied when increasing the training set size as shown in Figure 5.9. This is despite having a significant increase in MRR when the training set size is increased from 200 to 400 and 600. A plausible explanation on the observed differences is that our estimation of the probability that any random user will be satisfied when using our FAQ retrieval system treats documents ranked at different levels equally. We assume that if a user engages with a system longer to find the relevant answer for an FAQ document, his or her information needs will be equally satisfied

as someone who has found the relevant documents quicker (e.g one iteration). So, it is crucial that our FAQ retrieval system ranks as many relevant FAQ documents as possible within the search length desired by users in order to reduce the rate at which users abandon their search before their information need has been satisfied.

Our approach performed better that our baseline system described in (*EXV4*)[27], which uses a collection enrichment approach (Multiple Comparison test, $p < 0.05$). This is because, the expansion terms were selected automatically from an external collection of HIV/AIDS documents, which may result in some queries being expanded with non-relevant terms.

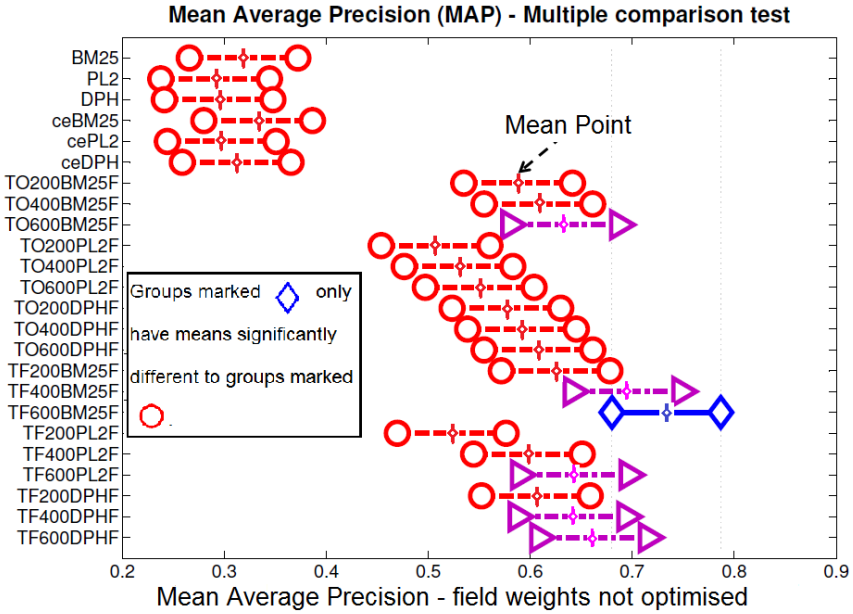


Figure 5.7: The confidence intervals of the MAP means for the different groups when the field weights are not optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

5.6.2 Field Weights Optimised

We also carried out an investigation to determine whether we can further improve the overall recall and the probability that any random user will be satisfied by optimising the field weights of the field-based weighting models (*EXV2*). As can be seen in Table 5.7, we recorded higher recall values ranging from 0.7418 to 0.8878 when the field weights are optimised compared to 0.6827 to 0.8612 when the field weights are not optimised in Table 5.6. It is worth noting that there is a difference in recall because we only carry out our evaluation

[27]In out a preliminary investigation, we also expanded the original queries with synsets from WordNet. These synsets were selected automatically and our preliminary results showed a significant decrease in the retrieval performance. In another query expansion approach, we selected the 10 most informative terms from the top 3 retrieved documents after first-pass retrieval on a local collection as expansion terms using Bose-Einstein 1 (Bo1) model and there was no significant improvement in the retrieval performance.

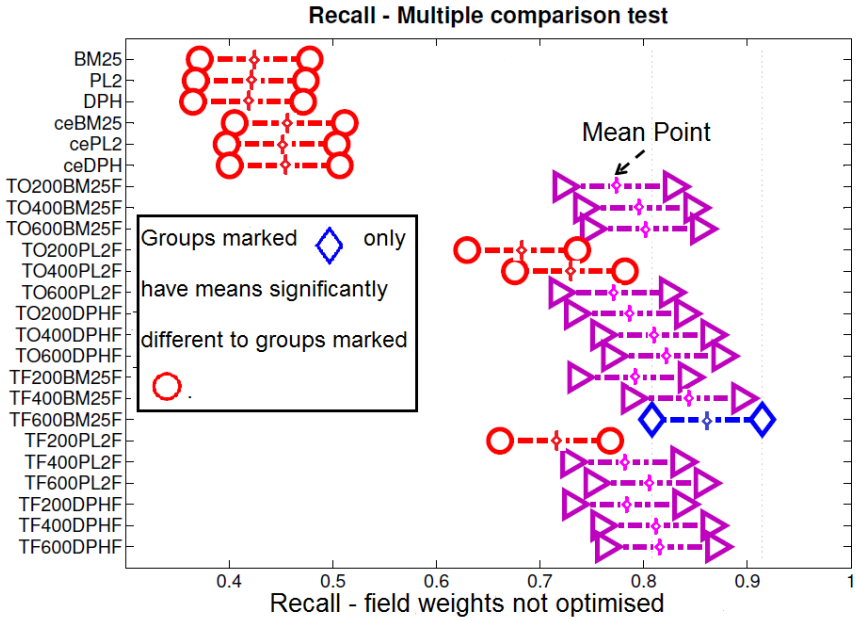


Figure 5.8: The confidence intervals of the rate of recall for the different groups when the field weights are not optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

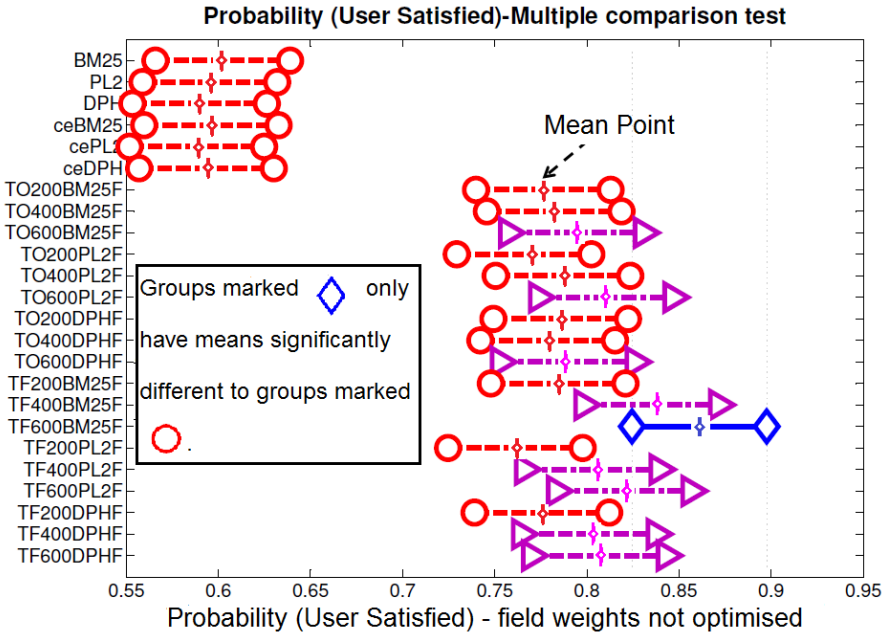


Figure 5.9: The confidence intervals of the probability that any random user will be satisfied for the different groups when the field weights are not optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

only on the top 20 retrieved documents. Similar results were recorded for the other evaluation measures in this study. One plausible explanation of the observed increase in retrieval performance after optimising the field weights is that the fields of high importance (*Question*

Table 5.6: The mean retrieval performance for each collection when field weights are not optimised. There is a significant improvement in the retrieval performance if the FAQ documents are enriched with queries over non enriched FAQ documents, as denoted by $*$ (Multiple comparison test, $p < 0.05$). Also, there is a significant difference between the Term Frequency approach and the Term Occurrence approach, as denoted by $**$ (Multiple comparison test, $p < 0.05$).

Weighting Model	Experiment	Collection	Enrichment	Test Evaluation Measure			
				MRR	MAP	Recall	P(Satisfied)
BM25F	*EXV1*	Q and A	No Enrichment	0.4973	0.3189	0.4246	0.6024
	EXV4		Collection Enrichment	0.4886	0.3332	0.4585	0.5962
	EXV1 and EXV3	Q,A and 200SMS	Term Occurrence	0.6854*	0.5881*	0.7749*	0.7764*
		Q,A and 400SMS		0.6745*	0.6079*	0.7947*	0.7823*
		Q,A and 600SMS		0.6971*	0.6335*	0.8012*	0.7938*
		Q,A and 200SMS	Term Frequency	0.6949*	0.6249*	0.7891*	0.7845*
		Q,A and 400SMS		0.7508*	0.6945*	0.8408*	0.8347*
		Q,A and 600SMS		**0.781****	0.7335*	0.8612*	0.8611*
PL2F	*EXV1*	Q and A	No Enrichment	0.4584	0.2940	0.4218	0.5954
	EXV4		Collection Enrichment	0.4181	0.2975	0.4507	0.5883
	EXV1 and EXV3	Q,A and 200SMS	Term Occurrence	0.6456*	0.5072*	0.6827*	0.7659*
		Q,A and 400SMS		0.6640*	0.5297*	0.7288*	0.7871*
		Q,A and 600SMS		0.6728*	0.5505*	0.771*	0.8100*
		Q,A and 200SMS	Term Frequency	0.6483*	0.5231*	0.7145*	0.7612*
		Q,A and 400SMS		0.684*	0.5978*	0.7823*	0.8025*
		Q,A and 600SMS		0.7092*	0.6429*	0.8047*	0.8198*
DPHF	*EXV1*	Q and A	No Enrichment	0.4640	0.2945	0.4182	0.5898
	EXV4		Collection Enrichment	0.4473	0.3121	0.4536	0.5936
	EXV1 and EXV3	Q,A and 200SMS	Term Occurrence	0.6801*	0.5766*	0.7858*	0.7859*
		Q,A and 400SMS		0.6742*	0.5919*	0.8112*	0.7788*
		Q,A and 600SMS		0.6873*	0.608*	0.8217*	0.7898*
		Q,A and 200SMS	Term Frequency	0.6755*	0.6055*	0.7841*	0.7756*
		Q,A and 400SMS		0.6962*	0.6407*	0.8114*	0.8008*
		Q,A and 600SMS		0.7052*	0.6617*	0.8159*	0.8063*

and *FAQLog* fields) have been assigned field weights of more than one, thus increasing the importance of term frequencies within those fields.

Table 5.7: The mean retrieval performance for each collection when field weights are optimised. There is a significant difference between the Term Frequency approach and the Term Occurrence approach, as denoted by $**$ (Multiple comparison test, $p < 0.05$).

Weighting Model	Experiment	Collection	Enrichment	Test Evaluation Measure			
				MRR	MAP	Recall	P(Satisfied)
BM25F	*EXV2 and EXV3*	Q,A and 200SMS	Term Occurrence	0.7124	0.6486	0.8334	0.7988
		Q,A and 400SMS		0.7089	0.6641	0.8437	0.7971
		Q,A and 600SMS		0.7163	0.6809	0.8466	0.7991
		Q,A and 200SMS	Term Frequency	0.7137	0.657	0.824	0.8002
		Q,A and 400SMS		0.7707	0.7271	0.869	0.8444
		Q,A and 600SMS		**0.795****	0.7583	0.8878	0.8654
PL2F	*EXV2 and EXV3*	Q,A and 200SMS	Term Occurrence	0.6815	0.5663	0.7418	0.7823
		Q,A and 400SMS		0.712	0.6051	0.7788	0.8158
		Q,A and 600SMS		0.7206	0.6291	0.8142	0.8402
		Q,A and 200SMS	Term Frequency	0.6891	0.6144	**0.8073****	0.7844
		Q,A and 400SMS		0.7236	0.674	**0.8554****	0.8177
		Q,A and 600SMS		0.7466	0.7073	0.8735	0.8404
DPHF	*EXV2 and EXV3*	Q,A and 200SMS	Term Occurrence	0.6975	0.6149	0.8309	0.7786
		Q,A and 400SMS		0.7175	0.6381	0.847	0.8213
		Q,A and 600SMS		0.7413	0.6611	0.8683	0.8472
		Q,A and 200SMS	Term Frequency	0.6713	0.6049	0.8109	0.771
		Q,A and 400SMS		0.7243	0.6653	0.857	0.8215
		Q,A and 600SMS		0.7382	0.689	0.8719	0.8422

Another important observation made is that when the field weights are optimised, there is no significant increase in retrieval performance when the training set size is increased from 200

to 400, 200 to 600 and 400 to 600 as illustrate in Figures 5.10 5.11, 5.12 and 5.13 (Multiple comparison test, $p < 0.05$). The only significant difference was observed when BM5F was deployed with the term frequencies enrichment strategies. There are possible explanations for this result. We have seen previously in Section 5.6.1 that when the field weights are not optimised, there is no significant difference in retrieval performance when the training set is increased from 400 to 600. This finding suggest that there might be a point where there is no gain in retrieval performance even when the number of training queries is increased. Hence, when assigning higher field weights, a saturation point might be reached with only 200 SMS queries. This is because higher field weights increases the term frequencies of each field.

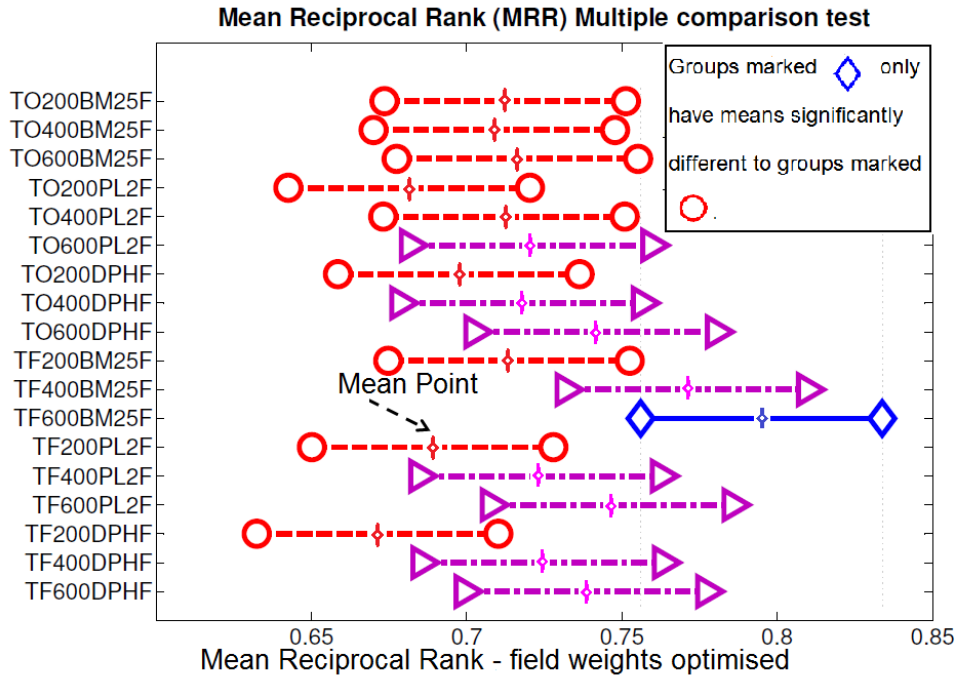


Figure 5.10: The confidence intervals of the MRR means for the different groups when the field weights are optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

5.7 Discussion and Conclusions

In this chapter, we described a field-based approach to reduce the term mismatch problem in our FAQ retrieval system dealing with questions related to HIV and AIDS. Our experiments show that the inclusion of a field derived from the logs of SMS queries for which the true relevant question-answer pair is known substantially improves the recall compared to the collection enrichment approach. An increase in recall verified that the term mismatch did indeed decrease with the proposed approach (See results in Table 5.9 and 5.7). As per our thesis statement, our results in Figure 5.9 and 5.13 suggest that the probability that any

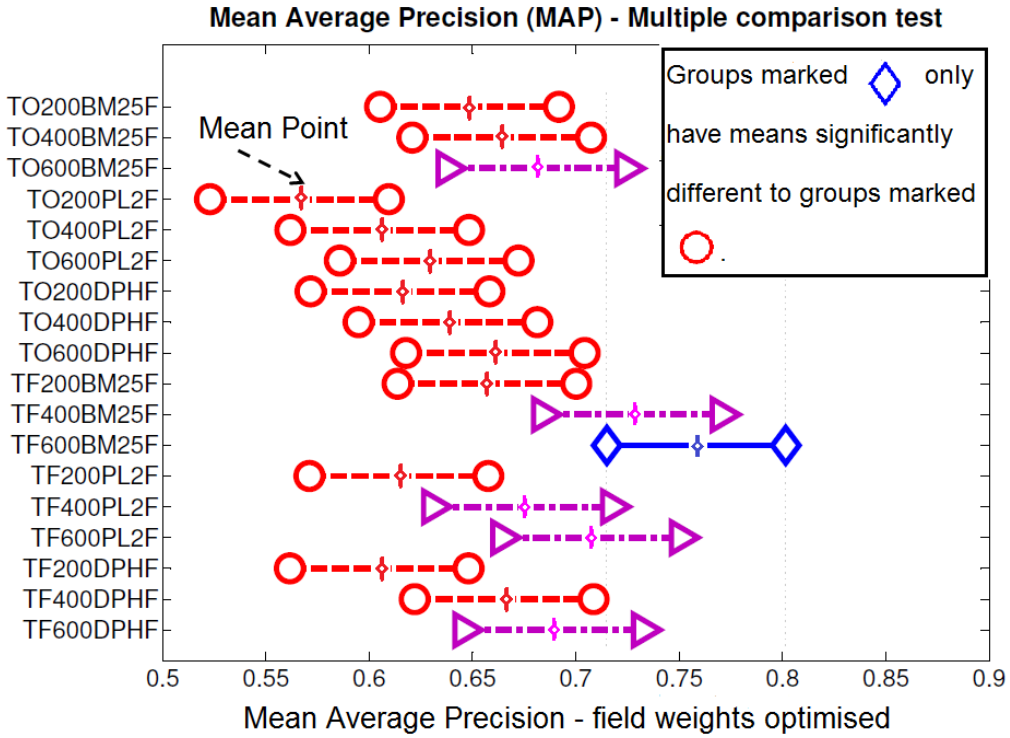


Figure 5.11: The confidence intervals of the MAP means for the different groups when the field weights are optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

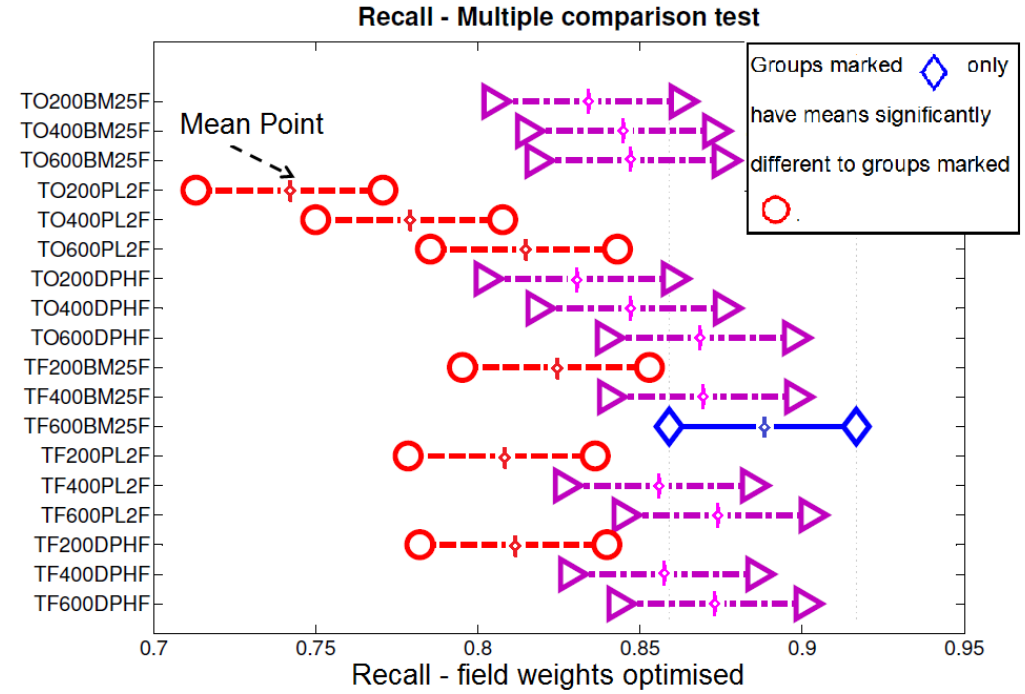


Figure 5.12: The confidence intervals of the rate of recall for the different groups when the field weights are optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

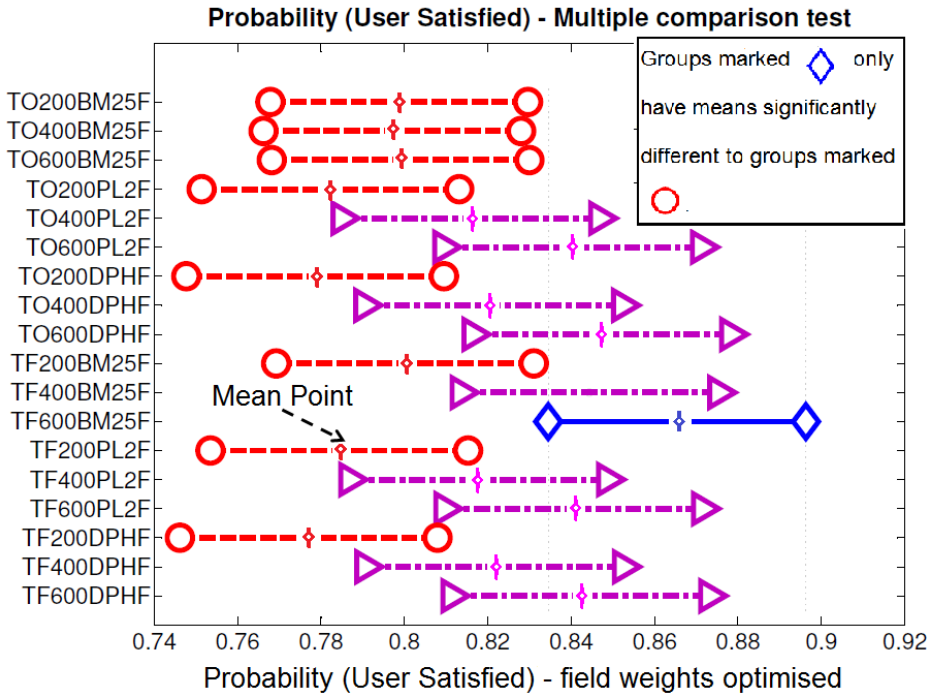


Figure 5.13: The confidence intervals of the probability that any random user will be satisfied for the different groups when the field weights are optimised. There is no significant difference in the retrieval performance when two confidence intervals overlap (Multiple comparison test, $p < 0.05$).

random user will be satisfied when using our FAQ retrieval system can be improved by enriching the FAQ documents with additional terms from a query log. Our results were consistent across all the field-based term weighting models deployed as shown in Table 5.9 and 5.7 (*C5-RQ4*).

In addition, we carried out an investigation to determine how the number of queries used to enrich the FAQ documents affect the retrieval performance (*C5-RQ3*). We saw a statistically significant increase in both recall and the average MRR when the number of queries used to enrich the FAQ documents were increased from 200 to 400 and 200 to 600 when the field weights were not optimised. An increase of training queries from 400 to 600 did not result in statistically significant improvement in MRR and recall.

Two different enrichment strategies were also investigated in this study. The term frequency enrichment approach produced higher MRR and recall values compared to the term occurrence approach. These findings suggests that it is important to take into consideration the number of times a term occurs in the queries when enriching the FAQ documents (*C5-RQ2*). The results also supports the idea that we can further improve the overall retrieval effectiveness of each enrichments strategy by optimising the field weights. In our investigation, we found that the retrieval performance can be improved by assigning a higher field weight to the *QUESTION* and the *FAQLog* field for DPHF and PL2F weighting models. This is be-

cause, these fields contain the same question words and phrases used by information seekers to expressed their on needs. In the next chapter, we carry out a further study with more focus on ranking the FAQ documents based on the number of times they were identified as relevant for a particular query term t_i. In particular, we propose another method for incorporating the term frequencies from a query log into the scoring and ranking of the FAQ documents for each user query. We will carry out our investigation on a non enriched FAQ document collection.

Chapter 6

Ranking the FAQ Documents Based on their Click Popularity Scores

6.1 Introduction and Motivation

In the previous chapter, we addressed the term mismatch problem in order to improve the probability that any random user will be satisfied when using our FAQ retrieval system by enriching the FAQ documents with term frequencies from a query log. In this chapter, we propose another method for incorporating the term frequencies from a query log into the scoring and ranking of the FAQ documents for each user query. Our aim is to address another FAQ document collection deficiency problem that we described in Chapter 1. In particular, the FAQ documents are created by the information supplier in advance and these may not always satisfy the users' information needs. The list of FAQ documents in the FAQ document collection may contain information that is not of interest to users. For example, we have seen in Chapter 3 that users ask questions that are only related to a subset of the FAQ documents in the FAQ document collection. Similar findings were also reported in (Sneiders, 2009).

Even though some FAQ documents are of less interest to users, the term weighting models that we deployed in the previous chapters treat all the FAQ documents equally when calculating their term weights. These term weighing models do not take into account the proportion of clicks (popularity) in the FAQ documents that answered the query term t. A click means that an FAQ document was identified as either relevant or slightly relevant for a given query term t. Incorporating the click popularity score of a query term t on an FAQ document d in the scoring and ranking function will ensure that the FAQ documents that are popular to users are ranked higher than non popular FAQ documents. Modern web search engines deploy algorithms such as PageRank to measure the importance of website pages so that search results could be ordered according to their importance (Brinkmeier, 2006). However, this approach cannot be deployed in our system because we do not have the number and quality of links

for each FAQ document to estimate its importance. The most valuable information we have is our query log and the associated clicks for each query.

In this chapter, we investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users. In particular, we propose to incorporate the click popularity score of a query term t on an FAQ document d into the scoring and ranking of the FAQ documents. Three approaches will be investigated. In the first approach, we propose to modify the Robertson/Sparck Jones (RSJ) weight (Robertson and Zaragoza, 2009) in the BM25 term-weighting model by multiplying it with the click popularity score of a query term t on an FAQ document d. This approach is motivated by previous works on query recommendation (Baeza-Yates et al., 2004, Baeza-Yates, 2005). In their approach, the weights for each term in the query term-weight vectors are calculated by replacing the IDF component with the click popularity (fraction of clicks in the retrieved documents) in the classical TF-IDF term weighting scheme. The click popularity scores in this case are used as a measure of importance for the query terms so that queries that have many clicks associated with URLs are recommended to users at the expense of those with fewer clicks.

Our proposed measure of popularity differs with the one defined in Baeza-Yates et al. (2004) and Baeza-Yates (2005) in three ways: (1) Instead of replacing the RSJ weight with the click popularity score, we multiply it with the click popularity score. (2) We add a smoothing factor to the click popularity score so that for previously unseen query terms, the $w^{(1)}$ reverts back to the RSJ weight. (3) We also differ in the way we compute the popularity score. In their work, Baeza-Yates et al. (2004) defined the popularity as the fraction of the documents returned by the query that captured the attention of users. In our work however, we define the click popularity score of a query term t on an FAQ document d as the fraction of clicks in the FAQ documents that answered the query term t. We believe that this definition of popularity is more suitable in distinguishing the importance of each query term to an FAQ document. In particular, it will be most suitable in ranking documents in a collection that was written by a single author. Documents in such a collection carry the linguistic signature of the same author and are likely to contain similar phrases and terms (Coulthard, 2004). Hence, this property can affect the retrieval performance of systems that rely on term weighting models in their ranking of the relevant documents.

In the second approach, we investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by deploying a click-based document prior in the Language Modelling (LM) approach for Information Retrieval (IR) (Ponte and Croft, 1998, Hiemstra, 1998). Since the document prior in the LM for IR can incorporate ranking preferences which are independent of the user query, we propose a click popularity score that takes into account the proportion of clicks associated with an FAQ document d in the FAQ document collection.

In our third and final approach, we investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by incorporating the click popularity scores of an FAQ document into the scoring process using a learning to rank technique. In our proposed learning to rank approach, we use the click popularity scores that we deployed in the first and second approach as features. The following research questions were identified to help us with our investigation:

Chapter 6-Research Question One (C6-RQ1): Can we improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users for a particular query term t.

Chapter 6-Research Question Two (C6-RQ2): Can we improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users without taking into consideration the query terms associated with those FAQ documents.

The remainder of this chapter is organised as follows:

- In Section 6.2, we describe how we combine the click popularity score of a query term t on an FAQ document d with the BM25 term weighting model when scoring and ranking the FAQ documents.
- We begin Section 6.3 by describing the Language Modeling approach for Information Retrieval. This is followed by a description of how we incorporate our click-based document prior into the scoring and ranking of the FAQ documents.
- In Section 6.4, we describe how we incorporate the click popularity score associated with an FAQ document into the scoring process using a learning to rank technique.
- In Section 6.5, we describe how we investigate and evaluate our baseline systems together with our proposed ranking functions that take into account the click popularity scores associated with an FAQ document. We also describe the dataset that we use in our experimental investigation and evaluation.
- Section 6.6 presents our results and analysis. This is followed by the discussion and conclusions in Section 6.6.

6.2 Combining the Click Popularity Score with the BM25 Term Weighting Model

As provided earlier in Chapter 5, the relevance score of a document d for a given query Q based on the BM25 term weighting model is expressed as:

$$score_{BM25}(d, Q) = \sum_{t \in Q} w^{(1)} \cdot \frac{(k_1 + 1)tfn}{k_1 + tfn} \cdot \frac{(k_3 + 1)qtf}{k_3 + qtf}. \tag{6.1}$$

where qtf is the number of occurrences of a given term t in the query Q. k_1 and k_3 are parameters of the model. tfn is the normalised within document term frequency. $w^{(1)}$ denotes the RSJ weight, which is an IDF factor. In our first approach, we want to ensure that the FAQ documents are ranked according to how often they have been previously identified as relevant by users for a particular query term t by multiplying the standard RSJ term weight ($w^{(1)}$) with the query term click popularity score *(C6-RQ1)*. In particular, for any given query Q, with terms t_i .... t_n, and any FAQ document d, we define the click popularity score of a query term t on an FAQ document d as the proportion of clicks in the FAQ documents that answered the query term t. This click popularity score is given by:

$$clickPop(t, d) = \frac{c(h_t, d) + 0.5}{\sum_{d_k \in C} c(h_t, d_k) + 0.5}, \tag{6.2}$$

where $c(h_t, d)$ is the count of clicks h_t in an FAQ document d, which was identified as relevant by a user who has issued a query with term t. $c(h_t, d_k)$ is the count of clicks h_t in any FAQ document d_k in the collection C, which was identified as relevant by a user who has issued a query with term t. We add a smoothing factor of 0.5 to ensure that documents without clicks are not completely ignored in the scoring process. Hence, all the FAQ documents containing this new query term will be treated equally as they will all be having a click popularity score $clickPop(t, d)$ of 1. The final score of an FAQ document d for any given query Q is computed as follows:

$$score(d, Q)_{productClickPop} = \sum_{t \in Q} clickPop(t, d) \cdot w^{(1)} \cdot \frac{(k_1 + 1)tfn}{k_1 + tfn} \cdot \frac{(k_3 + 1)qtf}{k_3 + qtf}. \tag{6.3}$$

6.3 Incorporating a Click-Based Document Prior in the Language Modelling for Information Retrieval

The Language Modeling approach for Information Retrieval was first introduced by Ponte and Croft (1998) in their work on the query likelihood scoring method. In their proposed method, the basic idea is to estimate a probabilistic language model for each document, and then rank documents by the likelihood of the query according to the estimated probabilistic language model of each document (Ponte and Croft, 1998, Lavrenko and Croft, 2001, Zhai, 2008). The query likelihood scoring function can be derived from computing the probability of a document given a query, $Pr(d|Q)$, using Bayes' rule so that ranking is proportional to the query likelihood given by (Losada and Azzopardi, 2008):

$$Pr(d|Q) = \frac{Pr(Q|d)Pr(d)}{Pr(Q)}, \tag{6.4}$$

where $Pr(Q)$ is a constant, which can be ignored since it does not affect the ranking. The prior probability of a document $Pr(d)$ can also be assumed to be uniform across all documents. Later in this section, we propose a non-uniform document prior.

The earlier approach proposed by Ponte and Croft uses the multiple Bernoulli model and it has not been popular because it ignores the term frequencies (Zhai, 2008). Under this model, the presence and absence of terms in assumed to be independent of that of other terms. In this work, we deploy a multinomial model, which was proposed by Zhai and Lafferty (2004). In a multinomial model, every word occurence is assumed to be independent, including the multiple occurrence of the same term (Zhai and Lafferty, 2004, Zhai, 2008). Hence, assuming the query terms are independent conditioned on the document:

$$Pr(Q|d) = \prod_{t \in Q} Pr(t|d), \tag{6.5}$$

where $Pr(t|d)$ is the probability of a term t given a document d. The main problem with this model is that any document that does not contain at least one query term is not returned (zero probabilities). To overcome this, several smoothing methods have been proposed. In particular, Bayesian smoothing with a Dirichlet Prior and Jelinek-Mercer (JM) smoothing (Losada and Azzopardi, 2008, Zhai and Lafferty, 2001, 2004). In this thesis, we use Bayesian smoothing with a Dirichlet Prior which is given by:

$$Pr(t|d) = \frac{tf + \mu Pr(t|C)}{l + \mu}, \tag{6.6}$$

where tf is the frequency of term t in document d, l is the total number of terms in docu-

ment d, μ is the parameter for smoothing and $Pr(t|C)$ is the probability of term t occurring in the collection. In this thesis, we set the smoothing parameter (μ) to its default value of 2500. $Pr(t|C)$ is given by $\frac{tfc}{token_c}$, where tfc is the frequency of term t in the collection C and $token_c$ is the number of tokens in the collection C. Substituting (Equation (6.6)) into (Equation (6.5)), applying logarithms and rearranging terms, the retrieval score of a document d given a query Q can be reduced to the following Equation (Losada and Azzopardi, 2008, Zhai and Lafferty, 2001, 2004):

$$score(d,Q) = \prod_{t \in Q} Pr(t|d) \propto \sum_{t \in Q} \log\left(1 + \frac{tf}{\mu \cdot \frac{tfc}{token_c}}\right) + \log\left(\frac{\mu}{l+\mu}\right). \tag{6.7}$$

In this thesis, we call the scoring function DirichletLM. In order to answer research question *C6-RQ2* (Can we improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users.), we deploy non-uniform document prior probabilities in (Equation (6.7)). We derive these probabilities from the proportion of clicks associated with each FAQ document d in the FAQ document collection. We compute the prior probability of each FAQ document according to the following expression:

$$Pr(d) = \frac{c(h,d) + 0.5}{\sum_{d_k \in C}\left(c(h,d_k) + 0.5\right)}, \tag{6.8}$$

where $c(h,d)$ is the number of clicks from previous searches associated with an FAQ document d. $c(h,d_k)$ is the number of clicks from previous searches associated with any other FAQ documents d_k in the collection C. We add a smoothing factor of 0.5 to ensure that documents without clicks are not completely ignored in the scoring and ranking process. Therefore, the score of a document d with a click-based prior probability $Pr(d)$ becomes:

$$score(d,Q) = \frac{c(h,d)+0.5}{\sum_{d_k \in C} c(h,d_k)+0.5} \cdot \sum_{t \in Q} \log\left(1 + \frac{tf}{\mu \cdot \frac{tfc}{token_c}}\right) + \log\left(\frac{\mu}{l+\mu}\right). \tag{6.9}$$

6.4 Incorporating the Click Popularity Score in a Learning to Rank Approach

In our third and final approach, we investigate whether we can improve the probability that any random user of our FAQ retrieval system will be satisfied by incorporating the click popularity scores of an FAQ document into the scoring process using a learning to rank technique. These are algorithms that use machine learning techniques to learn an appropriate combination of features into an effective ranking model (Liu, 2009). The idea behind using

learning to rank is that we can re-rank a sample of the top-ranked documents for a given query using the learned model before returning the results to the user. In general, the steps for learning an effective ranking model are as follows (Macdonald et al., 2013a,b):

1. Top K retrieval: Using a set of training queries that have relevance assessment, retrieve a sample of k documents using an initial weighting model such as BM25.

2. Feature extraction: For each document in the retrieved sample, extract a set of features. These features can either be query-dependent (term weighting models, term dependence models) or query-independent (click count, fraction of stopwords). The feature vector for each document is labelled according to the already existing relevance judgements.

3. Learning: Learn an effective ranking model by deploying an effective leaning to rank technique on the feature vectors of the top k documents.

This learned model can be deployed in a retrieval setting as follows:

4. Top K retrieval: For each unseen query, the top k documents are retrieved using the same retrieval strategy as in step (1)

5. Feature extraction: A set of features are extracted for each document in the sample of k documents. These features should be the same as those extracted in step (2).

6. Re-rank the documents: Re-rank the documents for the query by applying a learned model on every feature vector of the documents in the sample. The final ranking of the documents are obtained by sorting the predicted scores in descending order.

Learning to rank techniques are often classified as either listwise, pointwise or pairwise, depending on their loss function (Liu, 2009). Listwise approaches learn the ranking model on a set of documents associated with a query by optimising an IR evaluation measure such as MAP. Pointwise approaches on the other hand do not take into account the inter-dependency between the documents when learning the ranking model. They model ranking as regression, classification and ordinal regression. Therefore, pointwise approaches use regression loss, classification loss and regression loss as their loss function. Pairwise ranking techniques model ranking as pairwise classification. They use the classification loss on a pair of documents as a loss function. In this work, we will deploy two state-of-the-art listwise approaches. Prior work has indicated listwise approaches are often effective compared to the other approaches (Liu, 2009). In particular, we will deploy Coordinate Ascent (Metzler and Croft, 2007), which is a linear-based learner and LambdaMART (Burges et al., 2007), which is a tree-based learner. A linear-based learner yields a model that linearly combines

the feature values (Metzler and Croft, 2007, Burges et al., 2007, Macdonald et al., 2013b). The final score of a document d for any given query Q, for a linear leaner is given by:

$$score_(d, Q) = \sum_{f} \alpha_i \cdot f_{i,d} \tag{6.10}$$

where α_i is the weight of the i_{th} feature and $f_{i,d}$ is the value/score of the i_{th} feature for the document d.

On the other hand, a tree-based learner builds a set of regression trees T. The final score of a document d is obtained by traversing the nodes of a particular tree t, according to the decisions based on the vector of feature values of the document f_d (Burges et al., 2007, Macdonald et al., 2013b). The leaf node of the tree traversed represents the final score of the document d. This can be expressed as:

$$score_(d, Q) = \sum_{t \in T} t(f_d) \tag{6.11}$$

In our proposed learning to rank approach, we use the click popularity score of a query term t on an FAQ document and the click popularity score of an FAQ document as features. We also use several term weighting models as features. In Section 6.5.3 (Table 6.1), we provide a list of features that we use in our investigation.

6.5 Experimental Investigation and our Baseline Systems

In this section, we describe how we investigate and evaluate our proposed ranking functions that take into account the click popularity score of a query term t (C6-RQ1) on an FAQ document d and the click popularity score of an FAQ document d (C6-RQ2). First we provide a description of the dataset used in our investigation in Section 6.5.1. In Section 6.5.2, we describe how the click popularity score of a query t on an FAQ document d is computed from the dataset. In this section, we also describe how the click popularity score of an FAQ document is computed. Section 6.5.3 provides a description of the features that we deploy in our learning to rank approach. In Section 6.5.4 we provide our experimental setting.

6.5.1 Creating the Training and Testing Sets

We used the 10 different training and testing sets that we created as described in Section 3.3.1 (Chapter 3) to investigate our proposed scoring and ranking functions. As outlined in Chapter 3, we produced 10 random splits of the 750 matched SMS queries into a training set

of 600 queries and a testing set of 150 queries. In this chapter, we use the training set to compute the click popularity scores and the testing set to evaluate our proposed scoring and ranking functions.

6.5.2 Computing the Click Popularity Scores

The main contributions of this chapter as described in Section 6.1 is to investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users for a particular query term t (C6-RQ1). In particular, we used the 10 different training sets of 600 SMS queries that we created as described in Section 6.5.1 to compute the click popularity score of each query term t on an FAQ document d using Equation (6.2). The click popularity scores of each query term t on an FAQ document are deployed in the BM25 term weighting model as shown in Equation (6.3).

The other main contribution as described in Section 6.1 is to investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking the FAQ documents according to how often they have been previously identified as relevant by users without taking into consideration the query terms associated with those FAQ documents (C6-RQ2). In particular, we used the 10 different training sets of 600 SMS queries that we created as described in Section 6.5.1 to compute the proportion of hits/clicks (prior probability) for each FAQ document d using Equation (6.8). We deploy these click-based prior probabilities in Equation (6.9), which is based on the Language Modeling approach for Information Retrieval.

6.5.3 Features for our Learning to Rank Technique

In this section, we describe the features used in our learning to rank techniques. We used BM25 term weighting model to retrieve a sample of FAQ documents for each query in the training, testing and validation set. Since, the collection being searched is small, we retrieved all the FAQ documents that matched each query. Several query-dependent and query-independent features were extracted from this sample of documents. In particular, we used 6 different query-dependent weighting models and three field-based weighting models as features as shown in Table 6.1. For the field-based weighting models, we extracted features for both the question and answer field. The other set of features incorporate the click popularity score of a query into the query-dependent weighting models. These are described in Table 6.1. We use 10 different training sets of 600 SMS queries and their corresponding testing sets of 150 SMS queries that we created as described in Section 6.5.1 to create features for our learning to rank technique. Only 150 SMS queries in each training set were

randomly selected to generate features for training a model for learning to rank. Similarly, 150 SMS queries in the training set were randomly selected to generate features for validating the learned model. The 600 SMS queries in the training set were also used to compute the click popularity score of each query term t on an FAQ document d and the proportion of hits/clicks (prior probability) for each FAQ document d.

Table 6.1: All query-dependent (QD) and query-independent (QI) features used in this work.

Features	Type	Total
Weighting models (BM25, PL2, DPH, DLH, TF-IDF and DirichletLM)	QD	6
Field-based weighing models,Question and answer field (BM25F, PL2F and DPHF)	QD	6
DirichletLM with proportion of clicks/hits $Pr(d)$ as priors for an FAQ document d	QD&QI	1
FAQ document $score(d, Q)_{productClickPop}$ based on BM25 score with a modified RSJ term weight (multiplied the click Popularity score with the RJS term weight)	QD&QI	1
Total		14

6.5.4 Experimental Setting

FAQ Retrieval Platform: For all our experimental evaluation, we used Terrier-3.5[25] (Ounis et al., 2005), an open source IR platform. All the FAQ documents used in this study were first pre-processed before indexing and this involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). We have seen in Chapter 4 that BM25 does not perform well compared to other term weighting models when stopwords are not removed. Therefore, in all our experiments, we enabled stopword removal by ignoring the terms that had low IDF when scoring the documents. It was also observed in Chapter 4 that BM25 yields the best retrieval performance when there is no length normalisation ($b = 0.0$). Hence, in these experiments the normalisation parameter for BM25 was set to 0.0.

Training Learning to Rank Techniques: For our learning to rank approach, we used RankLib[28], a library of learning to rank algorithms. To train and test LambdaMART and Coordinate Ascent, we used the default RankLib parameter values of the algorithms. In all our experiments, we used MAP as the objective function (Macdonald et al., 2013a).

6.6 An Analysis of Experimental Results

In order to answer research questions *C6-RQ1* and *C6-RQ2*, we performed a paired t-test on the mean retrieval performance for all the test collections. In Table 6.2, we see a significant improvement in the retrieval performance when the FAQ documents are ranked according to how often they have been previously marked relevant by users for a particular query term t (*C6-RQ1*). In particular, a comparison between the baseline BM25 and the modified RSJ weight was made using a paired t-test and it was observed that the modified RSJ weight

[28]http://people.cs.umass.edu/ vdang/ranklib.html

Table 6.2: The mean retrieval performance for each collection (all FAQ documents that matched queries terms retrieved). RSJ weight gives a significant improvement on the MRR, MAP and the probability that any random user will be satisfied when using our FAQ retrieval system, as denoted by $*$ (paired t-test, $p < 0.05$). Also, there is a significant improvement in the retrieval performance in terms of MAP and MRR when a click based document prior is used in the Language Modelling for Information Retrieval, as denoted by $\diamond$ (paired t-test, $p < 0.05$). LambdaMART and Coordinate Ascent gives a significant improvement on the MRR, MAP, and the probability that any random user will be satisfied when using our FAQ retrieval system, as denoted by $\circledast$ (paired t-test, $p < 0.05$).

Weighting Model	Experiment	Collection	Test Evaluation Measure			
			MRR	MAP	Recall	P(Satisfied)
BM25	*Baseline*	Q and A	0.4973	0.3356	0.8237	0.6149
	Modified RSJ weight	Q,A and 600SMS	**0.6875**$*$	**0.5513**$*$	0.8237	**0.7620** $*$
DirichletLM	*Uniform Prior*	Q and A	0.4647	0.3247	0.8237	0.6053
	proportion of clicks	Q,A and 600SMS	**0.5139**$\diamond$	**0.3983**$\diamond$	0.8237	0.5951
LamdaMART	*Uniform Prior &*	Q,A and 600SMS	0.7011	**0.5873**$\circledast$	0.8237	0.7894
Coordinate Ascent	*proportion of clicks*	Q,A and 600SMS	**0.7243**$\circledast$	0.5729	0.8237	**0.7942**$\circledast$

gives a significant improvement on the probability that any random user will be satisfied when using our FAQ retrieval system, as denoted by $*$ in Table 6.2 (paired t-test, $p < 0.05$).

In our second approach however, there is no improvement on the probability that any random user will be satisfied when the FAQ documents are ranked according to how often they have been previously identified as relevant by users. In contrast, there is a significant improvement in retrieval performance in terms of MAP and MRR as denoted by $\diamond$ in Table 6.2 (paired t-test, $p < 0.05$). A closer examination of our results suggest that our new ranking function that uses a click-based document prior in the Language Modelling approach for Information Retrieval improved the retrieval performance for some queries and it degraded the retrieval performance for other queries.

Furthermore, we see a significant improvement in the retrieval performance when we incorporate the click popularity scores of an FAQ document into the scoring and ranking process using learning to rank techniques, as denoted by $\circledast$ (paired t-test, $p < 0.05$). In particular, there is a significant improvement in the probability that any random user will be satisfied when using our FAQ retrieval system when Coordinate Ascent is deployed compared to when non-learning to rank techniques such as BM25 and DirichletLM are deployed.

6.7 Discussion and Conclusions

In this chapter, we proposed three different approaches for ranking documents based on their click popularity scores. In the first approach we rank documents based on how often they have been previously marked relevant by users for a particular query term *t* (*C6-RQ1*). In the second approach we rank documents based on how often they have been identified as relevant

by users, without taking into account the query terms associated with those documents (*C6-RQ2*). In the third approach, we investigate whether we can improve the probability that any random user will be satisfied by incorporating the click popularity scores of an FAQ document into the scoring and ranking process using learning to rank techniques.

Our results suggest that we can improve the probability that any random user will be satisfied by ranking the FAQ documents based on how often they have been previously marked relevant by users for a particular query term t (*C6-RQ1*). However, there is no significant improvement on the probability that any random user will be satisfied when the FAQ documents are ranked based on how often they have been previously identified as relevant by users, without taking into consideration the query terms associated with those documents (*C6-RQ2*). Closer examination of our results suggest that our second approach degrades retrieval performance for some queries and it improves retrieval performance for some queries. Furthermore, when we incorporate the click popularity scores of an FAQ document into the scoring and ranking process using a learning to rank technique, we see a significant improvement in the retrieval performance over non-learning to rank techniques.

These findings may help us to further improve the retrieval performance of an FAQ retrieval system that uses a field-based approach to resolve the term mismatch problem between the user query and the relevant FAQ documents. Although we found such as system to be very effective in retrieving the relevant FAQ documents as discussed in Chapter 5, we postulate that we can further improve the retrieval performance of such a system by ranking the enriched FAQ documents according to how often they have been previously marked relevant by users for a particular query term t.

Chapter 7

Detecting Missing Content Queries

7.1 Introduction

In Chapter 1 (Section 1.2), we described the information source used to build the semi-automated FAQ retrieval system for HIV/AIDS in this thesis. In particular, this information source is made up of pre-stored sets of FAQ documents (FAQ document collection) to be searched by users. However, since the FAQ document collection to being searched is very small, it is possible that some user queries might not have the relevant FAQ documents in the FAQ document collection. Recall from Chapter 2 that these queries are referred to as *MCQs*. In this chapter, we use our query log to build a binary classifier for detecting these *MCQs*. Our aim is to deploy this binary classifier in our FAQ retrieval system so that when the *MCQs* are detected by the system, the FAQ manager (as discussed earlier in Chapter 4, (Section 4.3.4)) is automatically notified so that he/she can step in to help in the question answering process. The system will also notify the user that he/she will be shortly contacted by the FAQ manager.

Later in Chapter 8, we deploy this binary classifier in our FAQ retrieval system and investigate its impact on the probability that any random user will be satisfied when using our FAQ retrieval system. Before building such a classifier, we first empirically evaluate different feature sets in order to determine the set of features that can build a model that yields the best classification performance. We carry out our empirical evaluation using several feature sets generated from a query log before and after retrieval by the FAQ retrieval system. These feature sets were previously deployed in previous works as described in Chapter 2 (Section 2.3.5). The following research questions were identified for our empirical evaluation:

Chapter 7-Research Question One (C7-RQ1): Which set of features produce the best classification performance when classifying *MCQs* and *non-MCQs*?

Chapter 7-Research Question Two (C7-RQ2): Does combining different feature sets produce

a better classification performance compared to any individual feature set?

Chapter 7-Research Question Three (C7-RQ3): Does increasing the size of the training set for the *MCQs* and the *non-MCQs* yield a better classification performance?

In addition, we use the FIRE2012 SMS-Based FAQ retrieval task dataset in order to determine how our findings generalises to other datasets. The remainder of this chapter is organised as follows:

- In Section 7.2, we describe how we identified the *MCQs* and the *non-MCQs*.
- In Section 7.3, we describe how we create the training and testing instances for detecting *MCQs* and *non-MCQs*.
- Section 7.4 presents our experimental setting.
- In Section 7.5, we present our results and analysis followed by discussion and conclusions in Section 7.6.

7.2 Identifying MCQs and non-MCQs

Recall that in Chapter 3, we described a user study conducted in Botswana where 85 participants were recruited to provide SMS queries on the general topic of HIV/AIDS. Having provided the SMS queries, they then used a web-based interface to find the relevant FAQ documents from the FAQ document collection for these SMS queries. This provided us with SMS queries linked to the appropriate FAQ documents in the collection. In total, 957 SMS queries were collected of which 750 could be matched to an FAQ document in the collection (*non-MCQs*). The remaining 207 *MCQs* did not match anything in the collection. In this chapter, we investigate how to detect these *MCQs*. In order to determine how our findings will generalise to other datasets, we used a second dataset of 707 SMS queries (540 *non-MCQs* and 167 *MCQs*) that we randomly selected from the FIRE2012 English Monolingual SMS query dataset. This dataset had 4476 SMS queries. We selected only a fraction of these SMS queries to use in our experimental evaluation because we had to manually correct them for spelling errors.

A key difference between the two datasets (HIV/AIDS and FIRE2012 datasets) is that the FIRE2012 dataset covers several topics (Railways, telecommunication, health, career counselling and general knowledge etc.) while the HIV/AIDS dataset only has one topic, HIV/AIDS. Also, the *MCQs* for the HIV/AIDS dataset are the on-topic (related to HIV/AIDS only) while the *MCQs* for the FIRE2012 dataset has both the on-topic and the off-topic *MCQs*. The on-topic *MCQs* are those that are related to some of the topics in the collection being searched.

On the other hand, the off-topic *MCQs* are those that are not related to any topic in the collection being searched. Both the HIV/AIDS and the FIRE2012 SMS queries were manually corrected for spelling errors so that such a confounding variable does not influence the outcome of our experiments.

7.3 Creating the Feature Sets for Detecting MCQs and non-MCQs

In this Section, we provide details on how we create the feature sets for our empirical evaluation using the HIV/AIDS and the FIRE2012 SMS query log that we categorised as described in Section 7.2. Seven different feature sets were created for our empirical evaluation to answer the aforementioned research questions *C7-RQ1*, *C7-RQ2* and *C7-RQ3*. We provide details of the seven feature sets in Section 7.3.1, 7.3.1 and 7.3.2.

7.3.1 Feature Sets for Answering C7-RQ1

In order to determine the type of features that produce the best classification performance when classifying *MCQs* and *non-MCQs* (*C7-RQ1*), we created three different feature sets: *Query Strings (QS)*, *features generated from the retrieval scores and word overlap information (RSWO)* and *Query Difficulty Predictors (QDP)*. *RSWO* and *QDP* were previously deployed in Leveling (2012) and Hogan et al. (2011) to build a binary classifier that can detect *MCQs* and *non-MCQs*. In this thesis, we also introduce a third feature set, the query strings (*QS*) for our empirical evaluation. A description of all the feature sets is provided below.

QS: Instances in this feature set were represented by a vector of word counts from the text contained in the query strings. We used this feature set to investigate whether we can use information provided by the words in the user query to detect the *MCQs* and the *non-MCQs*. Below is an example of an instance, first represented as a query string and then as a vector of attributes representing word count information of this query string.

Query String : what does aids stand for?

Word Count : 23 1,159 1,212 1,488 1,591 1.

In our example above, the attributes in this vector are separated by commas and each attribute is made up of two parts, the attribute number, and the word count information. For example the attribute *"23 1"* denotes that the term $what$ is attribute number 23 in the string vector and this term only appear once.

RSWO: For this feature set, the training and testing instances were created using the approach proposed by Leveling (2012). In particular, numeric attributes generated during the retrieval phase of the FAQ documents by the *non-MCQs* and *MCQs* were used in this feature set. This feature set used the retrieval scores, number of retrieved documents and word overlap information to measure how well the user query matched the retrieved documents. For each query, we performed retrieval on the FAQ Retrieval Platform described in Section 7.4.1 to extract attributes for identifying *non-MCQs* and *MCQs*. The following is a description of the features generated from the retrieval scores and word overlap information (*RSWO*):

- The result set size (number of retrieved FAQ documents) [1 Feature].
- The raw BM25 scores for the top 5 retrieved documents [5 Features].
- The percentage difference between consecutive BM25 scores of the top five retrieved documents [4 features].
- The normalised BM25 scores for the top 5 retrieved documents. This is given as the sum of all IDF scores for all the query terms given a documents as proposed by Ferguson et al. (2011). The equation for calculating the IDF factor is given in Chapter 5 (Equation (5.2)) [5 features].
- The term overlap between the query and the top 5 documents as given in Equation (7.1), where $m(d, Q)$ represent the number of terms that appear in both the query Q and document d normalised by the query length $|Q|$ [5 features].

$$w\_overlap(d, Q) = \frac{m(d, Q)}{|Q|}. \tag{7.1}$$

QDP: For this feature set, the training and testing instances were created using eight different query difficulty predictors. Query difficulty predictors are normally used to predict whether a query will have a high average precision given retrieval from a particular collection, or low average precision (Hauff et al., 2008). Seven of the query difficulty predictors used in this study were pre-retrieval predictors and these were : Average Pointwise Mutual Information (AvPMI) (Hauff et al., 2008), Simplified Clarity Score (SCS) (He and Ounis, 2004), Average Inverse Collection Term Frequency (AvICTF) (He and Ounis, 2006), Average Inverse Document Frequency (AvIDF) (Hauff et al., 2008) and the derivatives of the similarity score between collection and query (SumSCQ, AvSCQ, MaxSCQ) (Zhao et al., 2008). One post-retrieval predictor was used, the Clarity Score (CS) (Cronen-Townsend et al., 2002). For each query, the FAQ Retrieval Platform described in Section 7.4.1 was used to generate the score for each query difficulty estimation predictor. The following is a description of the features used as query difficulty predictors (*QDP*):

- Average Pointwise Mutual Information (AvPMI): This pre-retrieval predictor measures the average mutual information of two query terms in the collection, averaged over all query terms. It is given by Equation (7.2), where $Pr(t_1, t_2)$ is the probability that the two terms t_1 and t_2 occur together in a document. $Pr(t_1)$ and $Pr(t_2)$ are the probabilities that the terms t_1 and t_2 occur in the collection. For example, $Pr(t_1)$ is given by $\frac{tfc}{token_c}$, where tfc is the frequency of term t_1 in the collection C and $token_c$ is the number of tokens in the collection C.

$$AvPMI(Q) = \frac{1}{|(t_1, t_2)|} \sum_{(t_1,t_2)\in Q} \log_2 \left(\frac{Pr(t_1, t_2)}{Pr(t_1) \cdot Pr(t_2)} \right). \tag{7.2}$$

- Simplified Clarity Score (SCS): This pre-retrieval predictor measures the Kullback-Leibler divergence of the query language model from the collection language model. It is given by Equation (7.3), where $P(t|Q)$ is simply the relative frequency of term t in the query Q, which is given by $\frac{qtf}{ql}$. qtf is the number of occurrences of a query term in the query and ql represents the length of the query. $Pr(t)$ is the probability that term t occur in the collection C and is given by $\frac{tfc}{token_c}$.

$$SCS(Q) = \sum_{t\in Q} Pr(t|Q) \cdot \log \left(\frac{Pr(t|Q)}{Pr(t)} \right). \tag{7.3}$$

- Average Inverse Collection Term Frequency (AvICTF): This pre-retrieval predictor measures the relative importance of a query term and is given by Equation (7.4), where ql is the query length, tfc is the frequency of a query term t in the collection C and $token_c$ is the number of tokens in the whole collection.

$$AvICTF(Q) = \frac{\log_2 \prod_{t\in Q} \frac{token_c}{tfc}}{ql}. \tag{7.4}$$

- Averaged Inverse Document Frequency (AvIDF): This pre-retrieval predictor takes the average IDF over all query terms as given in Equation (7.5), where Q is a query of length ql, N is the number of documents in the collection and dft is the number of documents containing a query term t.

$$AvIDF(Q) = \frac{1}{ql} \sum_{t\in Q} \log \frac{N}{dft}. \tag{7.5}$$

- Summed Collection Query Similarity (SumSCQ): This pre-retrieval predictor defines the similarity score between the query Q of length ql and the collection C as given by Equation (7.6). tfc is the frequency of a query term t in the collection C and dft is the

number of documents containing a query term t.

$$SumSCQ(Q) = \sum_{t \in Q} (1 + \ln(tfc)) \cdot \ln\left(1 + \frac{N}{dft}\right). \tag{7.6}$$

- Averaged Collection Query Similarity (AvSCQ): This is the average similarity over all the query terms and is expressed as:

$$AvSCQ(Q) = \frac{1}{ql} \cdot SumSCQ(Q). \tag{7.7}$$

- Maximum Collection Query Similarity (MaxSCQ): This pre-retrieval predictor relies on the maximum collection query similarity score over all query terms. This is expressed as:

$$MaxSCQ(Q) = max\left[\forall_{t \in Q}(1 + \ln(tf_c)) \cdot \ln\left(1 + \tfrac{N}{dft}\right)\right]. \tag{7.8}$$

- Clarity Score (CS): This post-retrieval predictor as defined by Cronen-Townsend et al. (2002) measures the Kullback-Leibler divergence between the language model of the result set, $Pr(\cdot|R)$, and the language model of the entire collection $Pr(\cdot|C)$, as given by Equation (7.9), where $V(C)$ is the set of terms in the entire collection.

$$CS(Q) = \sum_{t \in V(C)} Pr(t|Q) \cdot \log\left(\frac{Pr(t|Q)}{Pr(t)}\right). \tag{7.9}$$

$Pr(t)$ is the probability that term t occur in the collection and it was previously given as $\frac{tfc}{token_c}$ in Equation (7.3), where tfc is the frequency of term t in the collection and $token_c$ is the number of tokens in the collection. On the other hand, the query language model is computed by summing over all the documents in the retrieved set and is given by:

$$Pr(t|Q) = \sum_{d \in R} Pr(t|d) \cdot Pr(d|Q), \tag{7.10}$$

where d is a document in the result set R for a given query Q. The probability of a term t in document d can be estimated by using a document language model:

$$Pr(t|d) = \lambda \cdot Pr_{ml}(t|d) + (1 - \lambda)Pr(t), \tag{7.11}$$

$Pr_{ml}(t|d)$ is given by $\frac{tf}{l}$, where tf is the frequency of term t in document d and l is the length of the documents d. $Pr(t)$ is defined as in Equation (7.2). λ is a free parameter between 0 and 1.

Using Bayesian inversion, we obtain the probability of a document d given a query Q as given by Equation (7.12).

$$Pr(d|Q) = \frac{Pr(d) \cdot Pr(Q|d)}{Pr(Q)}, \tag{7.12}$$

In Equation (7.12), $Pr(Q)$ can be ignored because it is the same for all documents. $Pr(d)$ is the prior probability of a document d and $Pr(Q|d)$ can be computed as:

$$Pr(Q|d) = \prod_{t \in Q} Pr(t|d), \tag{7.13}$$

where the probability of term t given a document d, $Pr(t|d)$ is computed as in Equation (7.11).

7.3.2 Feature Sets for Answering C7-RQ2

Combined Feature Sets: We created four additional feature sets by combining the above feature sets *(QS, RSWO and QDF)* in order to answer research question, *C7-RQ2* (Does combining different feature sets produce a better classification performance compared to classifying using any individual feature set). The feature sets were simply combined by concatenating the corresponding instances. These four additional feature sets were: *QS+RSWO, QS+QDP, RSWO+QDP and QS+RSWO+QDP*.

7.3.3 Feature Sets for Answering C7-RQ3

In our empirical evaluation, we also investigate whether increasing the size of the training set for the *MCQs* and the *non-MCQs* yield a better classification performance (*C7-RQ3*). To answer this research question, we randomly split the feature set *(QS+RSWO+QDP)* 10 times into training and testing sets. For each training/testing split, we created two training sets, one containing 50% of the data (instances) and the other containing 75% of the data. The training set with 75% of the data was the superset of the training set with 50% of the data. The remaining 25% of the data was made the testing set. In total, we had 20 different training sets, 10 containing 50% of the data and the other 10 containing 75% of the data. The training data with 50% of the data shared the same testing set with its superset containing 75% of the data.

7.4 Experimental Setting

We begin Section 7.4.1 by describing the FAQ Retrieval Platform used for generating the features for the training and testing instances, followed by a description on how we train and classify the *non-MCQs* and the *MCQs* in Section 7.4.2.

7.4.1 FAQ Retrieval Platform

For our experimental evaluation, we used the Terrier-3.5[25] (Ounis et al., 2005), Information Retrieval (IR) platform with BM25 term weighting model. All the HIV/AIDS and the FIRE2012 FAQ documents used in this study were first pre-processed before indexing and this involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). We enabled stopword removal by ignoring the terms that had low IDF when scoring the documents. Indeed, all the terms with term frequency higher than the number of the FAQ documents (205) were considered to be low IDF terms. In Chapter 4, it was discovered that BM25 yields the best retrieval performance when there is no length normalisation ($b = 0.0$). Hence, in these experiments the normalisation parameter for BM25 was set to 0.0.

7.4.2 Training and Classifying Missing Content and Non-Missing Content Queries

Three different classifiers in WEKA, namely: Naive Bayes (John and Langley, 1995), RandomForest (Breiman, 2001) and C-Support Vector Classification (C-SVC) (Chang and Lin, 2011) were deployed in our empirical evaluation. These classifiers were chosen as they are popular and represent three broad families of classification methods. Naive Bayes is a probabilistic classifier and it assumes independence of features. C-SVC is a kernel based classifier and Random Forest is tree-based and it uses a combination of classifiers. Evidence from previous works suggest that Random Forest and Support Vector Classifiers achieve excellent performance compared to Naive Bayes across a wide variety of binary classification problems and evaluation metrics (Caruana and Niculescu-Mizil, 2006). We used three classifiers on the labelled feature sets created in Section 7.3 to train and classify *non-MCQs* and *MCQs*. For each feature set, we created 10 random splits of training and testing sets. For each training/testing split, each training set was made up of 75% of the data while the remaining 25% of the data was for testing. All the feature values in these training and testing sets were scaled between -1 and 1 as given by Equation (7.14).

$$fv_s = 2 \cdot \left(\frac{fv - min(fv)}{max(fv) - min(fv)} \right) - 1, \tag{7.14}$$

where fv is the original feature value and fv_s is the scaled/normalised feature value. Different kernels were used for C-SVC. A linear kernel was used for the feature sets with a large number of features (String) and a Radial Basis Function (RBF) kernel was used on the feature sets with few features. RBF kernels are suitable for non-linearly mapping samples into higher dimensional spaces so that they can handle the case when the relation between class labels and features is not linear. However, if the number of features is large (e.g String), there is no need to map data to a higher dimensional space because this will not improve the classification performance (Hsu et al., 2010). The regularization parameter C and the kernel parameter γ for the RBF kernel were chosen through a grid-search strategy. This involved performing a 10-fold cross validation on the training data (see Figure 7.1) with various pairs of (C, γ) and selecting the pair that gave the best classification accuracy. The same grid-search strategy was deployed to select the parameters for Random Forest. The C and the γ parameter for the RBF kernel were set to 1.0 and 0.9 respectively while the C parameter for the linear kernel was set to 0.7. For Random Forest, we set the number of trees to 10 for each feature set while the number of random features for creating the trees varied and were 5 and 10 for $fSet3$ and $fSet2$ respectively and 30 when using $fSet1$. In this empirical evaluation, we will define the $non - MCQs$ as the positive class and the $MCQs$ as the negative class. Table 7.1 shows a confusion matrix for the outcome of this two class problem.

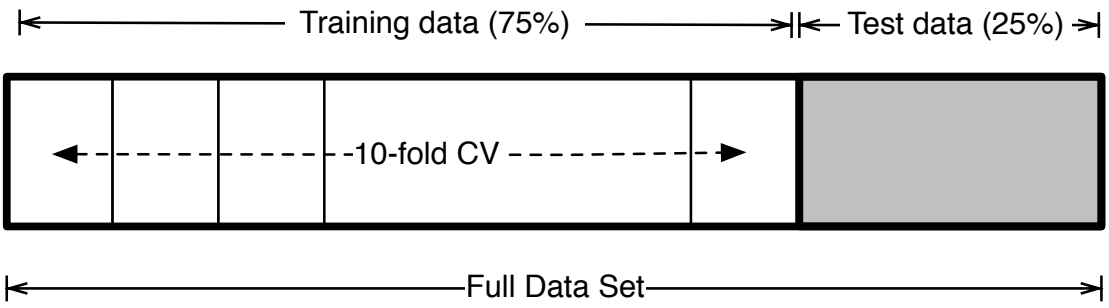


Figure 7.1: Training and testing sets

Table 7.1: Confusion matrix for a 2-class problem

		Predicted Class	
		non-MCQs (+ve)	*MCQs* (-ve)
Actual Class	*non-MCQs* (+ve)	True Positive (TP)	False Negative (FN)
	MCQs (-ve)	False Positive (FP)	True Negative (TN)

Table 7.2: The mean (for the 10 random splits) classification accuracy of all the feature sets. Significantly higher classification accuracy for the query string QS as compared to $RSWO$ and QDP, as denoted by $*$ and $\diamond$ (paired t-test, $p < 0.05$). Also there is a significant improvement when combining the QS with the other feature sets as denoted by $\circledast$ and $\triangleleft$ (paired t-test, $p < 0.05$). All the values depicted, range from 0 to 1 except the accuracy which is expressed as a percentage.

Dataset	Feature Set	Classifier	Sensitivity	Specificity	Accuracy(%)	ROC area	Kappa
$HIV/AIDS$	QS	NB	0.935	0.546	**85.06**$*$	**0.84**$*$	**0.522**$*$
		RF	0.983	0.406	**85.79**$*$	**0.857**$*$	**0.481**$*$
		C-SVC	0.959	0.454	**84.95**$*$	**0.833**$*$	**0.4812**$*$
$FIRE2012$	QS	NB	0.957	0.431	**83.31**$\diamond$	**0.807**$\diamond$	**0.457**$\diamond$
		RF	0.974	0.341	82.41	0.782	0.3935
		C-SVC	0.956	0.449	**83.59**$\diamond$	**0.832**$\diamond$	**0.4709**$\diamond$
$HIV/AIDS$	$RSWO$	NB	0.953	0.058	75.97	0.604	0.016
		RF	0.935	0.121	75.86	0.639	0.072
		C-SVC	0.999	0.005	78.37	0.502	0.0055
$FIRE2012$	$RSWO$	NB	0.836	0.593	77.86	0.767	0.4107
		RF	0.937	0.443	82.09	0.793	0.4333
		C-SVC	0.941	0.437	82.23	0.811	0.43382
$HIV/AIDS$	QDP	NB	0.891	0.473	**80.04**$\star$	**0.796**$\star$	**0.3821**$\star$
		RF	0.937	0.348	**80.98**$\star$	**0.777**$\star$	**0.337**$\star$
		C-SVC	0.969	0.251	**81.40**$\star$	**0.748**$\star$	**0.2867**$\star$
$FIRE2012$	QDP	NB	0.95	0.311	79.97	0.748	0.3199
		RF	0.958	0.389	82.37	0.737	0.4146
		C-SVC	0.978	0.380	82.63	0.759	0.4619
$HIV/AIDS$	$QS + RSWO$	NB	0.923	0.304	**78.89**$\circledast$	**0.694**$\circledast$	**0.2672**$\circledast$
		RF	0.989	0.271	**83.39**$\circledast$	**0.813**$\circledast$	**0.3465**$\circledast$
		C-SVC	0.952	0.449	**84.33**$\circledast$	**0.841**$\circledast$	**0.4647**$\circledast$
$FIRE2012$	$QS + RSWO$	NB	0.876	0.581	**80.62**$\triangleleft$	**0.771**$\triangleleft$	**0.4596**$\triangleleft$
		RF	0.981	0.329	82.74	0.836	0.3939
		C-SVC	0.961	0.479	**84.72**$\triangleleft$	**0.857**$\triangleleft$	**0.5097**$\triangleleft$
$HIV/AIDS$	$QS + QDP$	NB	0.900	0.507	**81.50**$\circledast$	**0.828**$\circledast$	**0.4274**$\circledast$
		RF	0.975	0.425	**85.89**$\circledast$	**0.903**$\circledast$	**0.4837**$\circledast$
		C-SVC	0.953	0.493	**85.37**$\circledast$	**0.866**$\circledast$	**0.5083**$\circledast$
$FIRE2012$	$QS + QDP$	NB	0.954	0.347	81.05	0.717	0.3643
		RF	0.989	0.383	**84.58**$\triangleleft$	**0.83**$\triangleleft$	**0.4655**$\triangleleft$
		C-SVC	0.948	0.521	**84.72**$\triangleleft$	**0.735**$\triangleleft$	**0.5256**$\triangleleft$
$HIV/AIDS$	$RSWO + QDP$	NB	0.903	0.435	80.15	0.774	0.2672
		RF	0.944	0.324	80.98	0.774	0.3230
		C-SVC	0.959	0.271	80.77	0.776	0.2854
$FIRE2012$	$RSWO + QDP$	NB	0.893	0.563	**81.52**$\bullet$	**0.807**$\bullet$	**0.4705**$\bullet$
		RF	0.948	0.479	**83.78**$\bullet$	**0.812**$\bullet$	**0.4869**$\bullet$
		C-SVC	0.954	0.593	**86.88**$\bullet$	**0.862**$\bullet$	**0.6001**$\bullet$
$HIV/AIDS$	$QS + RSWO + QDP$	NB	0.919	0.464	**82.03**$\circledast$	**0.804**$\circledast$	**0.4191**$\circledast$
		RF	0.979	0.314	**83.49**$\circledast$	**0.887**$\circledast$	**0.3754**$\circledast$
		C-SVC	0.948	0.502	**85.16**$\circledast$	**0.871**$\circledast$	**0.5072**$\circledast$
$FIRE2012$	$QS + RSWO + QDP$	NB	0.913	0.545	**82.60**$\triangleleft$	**0.794**$\triangleleft$	**0.4871**$\triangleleft$
		RF	0.972	0.413	**84.02**$\triangleleft$	**0.864**$\triangleleft$	**0.4653**$\triangleleft$
		C-SVC	0.965	0.515	85.86	0.866	0.5504

7.5 Experimental Results and Analysis

Table 7.2, summarises the overall classification accuracy for all the feature sets. The sensitivity measures the proportion of the actual positive instances (recall for TP) correctly classified as *non-MCQs*. The specificity measures the proportion of the actual negatives instances (re-

call for TN) correctly classified as *MCQs*. It can be seen from Table 7.2 that the different feature sets yielded fairly reasonable recall rates for the TP instances. In particular, the recall rates for TP (sensitivity) ranges from 0.891 to 0.999 for the HIV/AIDS dataset and and 0.836 to 0.989 for the FIRE2012 dataset. To put these values into perspective, these translate to between 668 and 749 correctly classified instances from a total of 750 instances for the HIV/AIDS dataset. In contrast, our classifiers did not perform well for the TN instances. Fairly low recall rates (specificity) for the TN instances were observed. Depending on the feature set and the classifier used, the specificity ranged from 0.005 to 0.546 for the HIV/AIDS dataset and from 0.311 to 0.593 for the FIRE2012 dataset. These values translate to between 1 and 113 correctly classified TN instances from a total of 207 TN instances for the HIV/AIDS and between 52 and 99 from a total of 167 for the FIRE2012 dataset. Our empirical evaluation suggests that all the feature sets performed well for the *non-MCQs* (TP instances). For the *MCQs* (TN instances), the best performing feature set only yielded roughly 50% classification. When we compare our results with previous works, we observe that our classifiers performed fairly poorly in the detection of *MCQs*. One plausible explanation on this dissimilarly is that our dataset was fairly unbalanced. The majority class for our classifiers was the *non-MCQs* while in previous studies it was the *MCQs*.

To answer research question *C7-RQ1* (Which type of features produce the best classification performance when classifying *MCQs* and *non-MCQs*), we used a paired t-test to analyse the classification accuracy between the following 10 random splits, (*QS* and *RSWO*), (*QS* and *QDP*), and (*RSWO* and *QDP*). The actual query strings (*QS*) provided a significantly higher classification accuracy (paired t-test, $p < 0.05$) compared to the other feature sets as denoted by $*$ for the HIV/AIDS dataset and $\diamond$ for the FIRE2012 dataset. These results show that it is important to use the actual words contained in the user query to detect the *MCQs* and the *non-MCQs*. Also observed were significantly higher (paired t-test, $p < 0.05$) Kappa statistic and ROC area (AUC) for the *QS*. The kappa statistic measures the agreement of prediction with the true class and a value of 1 signifies total agreement and a value of 0 signifies total disagreement. The ROC area on the other-hand signifies the overall ability of the classifier to identify *MCQs* and *non-MCQs*. The best classifier has an area of 1.0 and a classifier with an area of 0.5 or lower is considered ineffective.

A comparison between the features generated from the retrieval scores and word overlap information (*RSWO*), and the features generated by using the eight different query difficulty predictors (*QDP*) was also made using a paired t-test. It was observed that query difficulty predictors (*QDP*) give a better classification accuracy for the HIV/AIDS dataset as denoted by $\star$ in Table 7.2. No significant difference in classification accuracy was observed between *RSWO* and *QDP* for the FIRE2012 dataset. This disparity between the HIV/AIDS and the FIRE2012 dataset when we compare the classification accuracy between *RSWO* and *QDP* suggest that the retrieval scores and word overlap information (used in *RSWO*) are not good

discriminators for the on-topic *MCQs*. Although *RSWO* did not perform well for the on-topic *MCQs* (TN instances, HIV/AIDS dataset), it performed well for the off-topic *MCQs* (TN instances FIRE2012 dataset) as depicted by higher specificity values.

There was a significantly higher classification accuracy observed when the query strings (*QS*) were combined with the other feature sets (research question *C7-RQ2*). This is denoted by $\circledast$ and $\triangleleft$ in Table 7.2 for the HIV/AIDS and FIRE2012 dataset respectively (paired $t-test < 0.05$ for ((*QS*+*RSWO*) and *RSWO*), ((*QS*+*QDP*) and *QDP*) and ((*QS*+*RSWO*+*QDP*) and (*RSWO*+*QDP*)). Similar findings were observed when the query difficulty predictors (*QDP*) were combined with features generated from the retrieval scores and word overlap information (*RSWO*) as denoted by $\bullet$, (paired $t-test < 0.05$ for ((*RSWO*+*QDP*) and *QDP*))

A paired t-test was used to analyse whether increasing the size of the training instances increases the classification accuracy (research question *C7-RQ3*). The results, as shown in Table 7.3, indicate that there is a significant difference in classification accuracy as denoted by $*$ (paired t-test, $p < 0.05$) when the training set in increased by 25% from the original 50% of the data to 75% of the data.

Table 7.3: The overall classification accuracy for ($QS + RSWO + QDP$). One training set contains 50% of the data (instance) and the other contains 75% of the data. There is a significant improvement in the classification accuracy when the size of the training instances is increased, as denoted by $*$ (paired t-test, $p < 0.05$). All the values depicted, range from 0 to 1 except the accuracy which is expressed as a percentage.

Dataset	Feature Set	Training Set Size	Classifier	Sensitivity	Specificity	Accuracy(%)	ROC area	Kappa
$HIV/AIDS$	$QS + RSWO + QDP$	50%	NB	0.924	0.454	82.24	0.816	0.4192
			RF	0.976	0.271	82.34	0.86	0.3213
			C-SVC	0.96	0.478	85.58	0.889	0.5075
$FIRE2012$	$QS + RSWO + QDP$	50%	NB	0.898	0.473	79.77	0.757	0.3984
			RF	0.972	0.305	82.47	0.837	0.3509
			C-SVC	0.943	0.462	82.88	0.832	0.4598
$HIV/AIDS$	$QS + RSWO + QDP$	75%	NB	0.923	0.493	**82.98**$*$	**0.822**$*$	**0.4526**$*$
			RF	0.983	0.266	**82.75**$*$	**0.884**$*$	**0.3281**$*$
			C-SVC	0.956	0.56	**87.04**$*$	0.886	**0.5747**$*$
$FIRE2012$	$QS + RSWO + QDP$	75%	NB	0.92	0.539	**83.02**$*$	**0.798**$*$	**0.494**$*$
			RF	0.972	0.443	**84.72**$*$	**0.882**$*$	**0.4952**$*$
			C-SVC	0.967	0.497	**85.57**$*$	**0.85**$*$	**0.537**$*$

7.6 Conclusions

In this chapter, we built a binary classifier for detecting *MCQs* and *non-MCQs* in our FAQ retrieval system. Before building the classifier, we conducted an empirical evaluation to determine the set of features that can build a model that yields the best classification performance. Several research questions were addressed to achieve the above goal. Our result suggest that the most important feature set (research question *C7-RQ1*) for building our classifier is the actual query string (*QS*), which is a set of attributes representing word count

information from the text contained in the query strings. The query strings (QS) provided a significantly higher classification accuracy (paired t-test, $p < 0.05$) compared to the other feature sets across the different classifiers as denoted by $*$ in Table 7.2. It also emerged from this study that the classification accuracy of a classifier built using features generated from the retrieval scores and word overlap information (*RSWO*) and those generated by the different query difficulty predictors (*QDP*) can be improved further by combining these feature sets with the actual query strings (*QS*) (research question *C7-RQ2*), in particular *QDP* (feature sets generated by query difficulty predictors). In future, we will investigate better ways on how to combine these feature sets, in order to improve the classification accuracy.

In addition, we also investigate whether increasing the training set size would yield a better classification accuracy (research question *C7-RQ3*). A significant increase in accuracy, ROC area and Kappa statistic was observed when the training set was increased by 25%. These results suggest that we should collect more data in future to improve the performance of our classifier. The other finding to emerge from this study is that some feature sets work best for some datasets and perform poorly on other datasets. As our results suggest in Table 7.2, features generated from the retrieval scores and word overlap information (*RSWO*) do not perform well when the *MCQs* are on-topic (*MCQs* related to the FAQ document collection) as in the case of the HIV/AIDS dataset. This feature set does however perform well when the *MCQs* are off-topic (*MCQs* not related to the FAQ document collection) as in the case of the FIRE2012 dataset. However, the query strings ((*QS*) and the query difficulty predictors (*QDP*) perform well across these different collections.

Based on these findings, in Chapter 8, we deploy a C-SVC based binary classifier that uses the combined feature sets: query strings (*QS*), features generated from the retrieval scores and word overlap information (*RSWO*) and query difficulty predictors (*QDP*) in our FAQ retrieval system to detect missing content queries. We chose this classifier and the combined feature sets because they yielded the best classification performance as shown in Table 7.2. In Chapter 8, we also investigate the impact of deploying the missing content queries detection system on user satisfaction.

Chapter 8

Testing the Generality of our Previous Results and Findings

8.1 Introduction

In this chapter, we test whether our previous results and findings generalise on a second dataset. Recall that in Chapter 1, we discussed the main aspects to consider when developing an automated FAQ retrieval system. Four main aspects were identified. These are handling noisy text, the FAQ document collection deficiency problems (no relevant information and the term mismatch problem), the search result presentation problem on low-end mobile phone devices and handling cross-lingual and bilingual queries. In this thesis, we addressed two of these aspects: the FAQ document collection deficiency problems and the search result presentation problem on low-end mobile phones in order to improve the probability that any random user will be satisfied when using our FAQ retrieval system. In this chapter, our aim is to combine and evaluate the different subsystems that we developed to address the aforementioned problems in our FAQ retrieval system. In particular, we investigate whether the previous results generalise on other datasets, including when we combine our sub-systems. Also, we investigate the impact of the missing content queries detection system that we developed in Chapter 7 on the probability that any random user will be satisfied when using our FAQ retrieval system. In addition, we also investigate whether we can improve the retrieval performance and the probability that any random user will be satisfied by correcting spelling errors. In our empirical investigation and evaluation, we use an additional dataset that we created as described in Section 8.2 for testing and we use the previous dataset that we created as described in Chapter 3 for training. The rationale for using an independent test set is to investigate how well our previous results generalise on other datasets. The following research questions were identified to help us with our investigation.

Chapter 8-Research Question One (C8-RQ1): Do the previous results generalise on a second dataset, including when we combine our subsystems.

Chapter 8-Research Question Two (C8-RQ2): What impact does the missing content queries detection system have on the probability that any random user will be satisfied when using our FAQ retrieval system?

Chapter 8-Research Question Three (C8-RQ3): Does correcting spelling errors help improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system?

The remainder of this chapter is organised as follows:

- In Section 8.2, we describe the datasets used in our empirical investigation and evaluation. In particular, we describe how we created the second dataset for testing our FAQ retrieval system. We also describe how we created another dataset for training our FAQ retrieval system.
- In Section 8.3, we describe our experimental investigation. In particular, we describe how we combine and evaluate the different subsystems that we developed to address the FAQ document collection deficiency problems and the result presentation problem on low-end mobile phones. We also investigate whether we can improve the retrieval performance and the probability that any random user will be satisfied by correcting spelling errors.
- In Section 8.5, we present our results and analysis. This is followed by the discussion and conclusions in Section 8.6.

8.2 Creating the Training and Testing Sets

8.2.1 Creating the Testing Set

We conducted another user study in Botswana form 1^{st} November 2013 to 30^{th} January 2014 to create a second dataset to use in assessing how well our previous results generalise on other datasets, including when we combine our sub-systems. This study was granted the University of Glasgow ethics approval and was allocated the ethics project reference number: CSE01286. We recruited 39 participants in the city of Gaborone at the main bus rank. In total, there were 21 males and 18 females who took part in this study. Their ages ranged from 18 to 46. Only people who owned low-end mobile phones were allowed to take part in our user study.

In this study, the participants were asked to use their mobile phones to type and send at least 10 SMS queries on the general topic of HIV/AIDS to the mobile phone number that we provided. Just like in our earlier study in Chapter 3, participants were not shown the

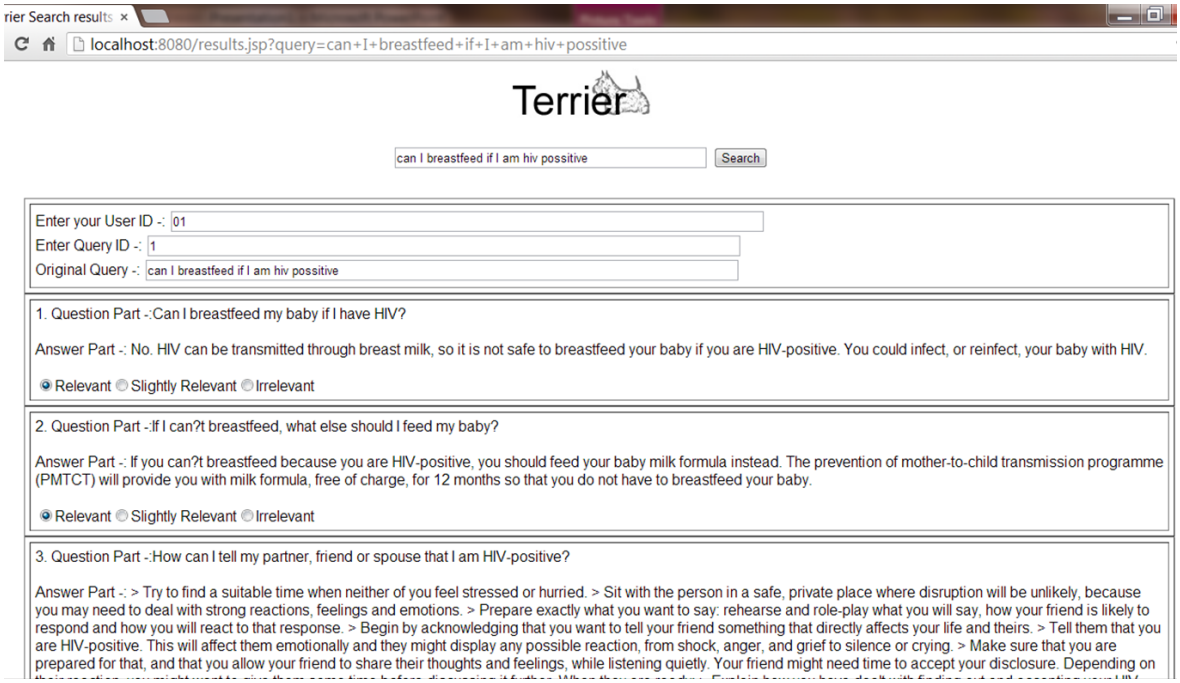


Figure 8.1: The web-based HIV/AIDS FAQ retrieval system used for collecting query relevance information.

FAQ documents in the information source for the FAQ retrieval system. After submitting the SMS queries, the participants were asked to use a spell checker to correct spelling errors. We also kept a record of the original SMS queries with spelling errors. Participant were then presented with a web-based FAQ retrieval system for HIV/AIDS and were asked to use the SMS queries, which were corrected for spelling errors to search for the relevant FAQ documents. For each SMS query, the system ranked and retrieved the top 5 FAQ documents and the participants were asked to judge them as either relevant, slightly relevant or irrelevant as shown in Figure 8.1. We only asked participants to judge up to 5 retrieved documents because in our earlier study in Chapter 4 (Section 4.4), we learnt that participants tolerate up to 3 iterations.

Table 8.1: The total number of SMS queries with relevant and non-relevant FAQ documents in the top 5 retrieved documents.

	Number
Number of collected SMS queries	441
Number of SMS queries with relevant FAQ documents	328
Number of SMS queries with non-relevant FAQ documents	113
Number of FAQs that matches the SMS queries	158
Number of FAQs that did not match any SMS queries	47

Our web-based FAQ retrieval system used a term weighting model that produced the best recall in our earlier work in Thuma et al. (2013). In particular, we used PL2F term weighting

Table 8.2: Distribution of judgements in the top 5 retrieved FAQ documents for the 441 SMS queries.

	Number
Total number of FAQ documents retrieved	2205
Number of FAQ documents judged relevant	518
Number of FAQ documents judged slightly relevant	388
Number of FAQ documents judged non-relevant	1299

model to score and rank the HIV/AIDS FAQ documents, which were enriched using 600 SMS queries from one of the training sets that we created in Chapter 3. However, in Chapter 5, we later found out that BM25F outperforms PL2F when the documents lengths are not normalised. Hence, we deploy BM25F in our future evaluation. Also, we deploy the Term Frequency enrichment strategy because it outperformed the Term Occurrence enrichment strategy in our earlier study in Chapter 5 (see results in Section 5.6). Table 8.1 shows our query log analysis. In total, we collected 441 SMS queries. 328 of these SMS queries had relevant and slightly relevant FAQ documents in the top 5 retrieved documents. 113 of the SMS queries did not have a relevant FAQ document in the top 5 retrieved documents. A closer examination of the SMS queries that did not have a relevant FAQ document in the top 5 retrieved documents suggests that a majority of them were *MCQs*. Examples of these *MCQs* are:

- Is it true that HIV has been manufactured in the lab?
- In which year was aids discovered?
- Which continent is mostly affected by aids?
- In which region do you find the most deadly virus?
- How much does a month's supply of ARVs cost at local pharmacies?
- Has it ever occurred that someone who was on treatment tested negative after prolonged use?

We also conducted a further analysis of the click-through data to determine the number of FAQ documents judged relevant, slightly relevant and non-relevant. Table 8.2 provides a summary of these judgements. In total, 2205 FAQ documents were judged for the 441 SMS queries. 518 of these FAQ documents were judged relevant while 388 were judged slightly relevant. The remaining 1299 were judged as non-relevant. We used these judgements to create a query relevance file for future evaluation of our system. Similar to our earlier approach in Chapter 3, all the FAQ documents that were judged slightly relevant were considered relevant when creating our query relevance file. One main limitation of our query relevance file

is that it lacks completeness as only the top 5 FAQ documents were judged. It is possible that some relevant FAQ documents were not in the top 5 retrieved FAQ documents and were not judged by the participants. In future, we will use several systems to create a pool of the top FAQ documents to be judged by users.

8.2.2 Creating the Training Set

We used one of the different training sets that we created as described in Section 3.3.1 (Chapter 3) for our empirical evaluation to investigate the effect of combining the different subsystems on user satisfaction. As outlined in Chapter 3, we produced 10 random splits of the 750 matched SMS queries into training sets of 600 SMS queries and testing sets of 150 queries. In order to answer research question C8-RQ2, we used from our query log (Chapter 3) the 207 SMS queries that did not match anything in the collection (*MCQs*) together with the 750 matched SMS queries (*non-MCQs*) to create the training instances for our binary classifier for detecting *MCQs*.

8.3 Combining and Evaluating the Different Sub-Systems

8.3.1 Using a Field-Based Approach to Rank the Enriched FAQ Documents based on their Click Popularity Scores.

One of the main contributions in this chapter is to investigate whether the previous results generalise on other datasets, including when we combine our sub-systems (*C8-RQ1*). In order to answer this research question, we combine the approach that we proposed in Chapter 6 for ranking the FAQ documents based on how often they have been previously identified as relevant by users for a particular query term t with our other approach that we proposed in Chapter 5 for enriching the FAQ documents with additional terms from a query log in order to resolve the term mismatch problem. In our investigation, we use one of the training set of 600 SMS queries that we created as described in Section 8.2 to enrich our FAQ documents using the Term Frequency enrichment strategy that we proposed in Chapter 5 (Section 5.3). We only use the Term Frequency enrichment strategy because earlier in Chapter 5, it outperformed the Term Occurrence enrichment approach. We also used this training set of 600 SMS queries to compute the click popularity score of each query term t on an FAQ document d using Equation (6.2). The click popularity scores of each query term t on an FAQ document are deployed in the BM25F term weighting model as shown in Equation (8.1) below. We use the testing set that we created in Section 8.2.1 in our investigation.

$$score(d,Q)_{BM25FClickPop} = \sum_{t \in Q} clickPop(t,d) \cdot w^{(1)} \cdot \frac{(k_1+1)tfn}{k_1+tfn} \cdot \frac{(k_3+1)qtf}{k_3+qtf}. \quad (8.1)$$

8.3.2 Effects of the Missing Content Query Detection System On User Satisfaction

In this chapter, we also investigate the impact of the missing content queries (*MCQs*) detection system on the probability that any random user will be satisfied when using our FAQ retrieval system (*C8-RQ2*). In our investigation, we deploy the best binary classifier from Chapter 7, which is C-Support Vector Classification (C-SVC) based and uses the combined feature sets: query strings (*QS*), features generated from the retrieval scores and word overlap information (*RSWO*) and query difficulty predictors (*QDP*). Recall from Chapter 7 that the training and testing instances in *QS* were represented by a vector of word count from the text contained in the query strings. On the other hand, the training and testing instances in *RSWO* were generated during the retrieval phase of the FAQ documents by the *non-MCQs* and *MCQs*. For *QDP*, the training and testing instances were created using eight different query difficulty estimation predictors. We use the training and testing sets that we created in Section 8.2 to create the feature sets for the *non-MCQs* and *MCQs*.

8.3.3 Effects of Noisy SMS Queries on User Satisfaction

Recall from Chapter 1 that we identified correcting noisy SMS queries as one of the aspects to consider when developing an FAQ retrieval system. In this thesis however, we did not focus on this aspect. Instead, we manually corrected all the SMS queries for spelling errors. In this chapter, we investigate whether we can improve the retrieval performance and the probability that any random user will be satisfied by correcting spelling errors (*C8-RQ3*). In particular, we evaluate our FAQ retrieval system using two different versions of our testing set that we created as described in Section 8.2.1. The first testing set contained the original SMS queries from the participants, which were not manually corrected for spelling errors. The second testing set contained the same set of SMS queries, which were manually corrected for spelling errors.

8.4 Experimental Setting

8.4.1 FAQ Retrieval Platform

For all our experimental evaluation, we used Terrier-3.5[25] (Ounis et al., 2005), an open source IR platform. All the FAQ documents used in this study were first pre-processed before indexing and this involved tokenising the text and stemming each token using the full Porter stemming algorithm (Porter, 1997). We have seen in Chapter 4 that BM25F yields the best retrieval performance when the field weights are set to ($w_Q = 3, w_A = 1, w_{QL} = 2$), where ($w_Q, w_A$ and w_{QL}) represents the *QUESTION*, *ANSWER* and *FAQLog* field weights respectively. Therefore, in all our experiments, we set all the field weights to ($w_Q = 3, w_A = 1, w_{QL} = 2$). It was also discovered in Chapter 4 that BM25 yields the best retrieval performance when there is no length normalisation ($b = 0.0$). A possible explanation for this is that all the FAQ documents in the collection are roughly of the same length. Hence, in these experiments the normalisation parameter for BM25F was set to 0.0.

8.4.2 Training and Classifying Missing Content and Non-Missing Content Queries

In our empirical investigation and evaluation, we deployed the C-SVC (Chang and Lin, 2011) classifier in WEKA because earlier in Chapter 7 (see results in Section 7.5), it yielded the best classification accuracy compared to Naive Bayes (John and Langley, 1995) and RandomForest (Breiman, 2001). We deploy this classifier on the feature sets for the training and testing sets that we described in Section 7.3 to train and classify *non-MCQs* and *MCQs*. All the feature values for these training and testing sets were scaled between -1 and 1 using Equation (7.14), which was previously given in Chapter 7 (see Section 7.4.2).

8.5 Results and Analysis

8.5.1 Using a Field-Based Approach to Rank the Enriched FAQ Documents based on their Click Popularity Scores.

In this chapter, we first investigate whether we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by combining the approach that we proposed in Chapter 6 for ranking the FAQ documents based on how often they have been previously identified as relevant by users for a particular query term t with our other approach that we proposed in Chapter 5 for enriching the FAQ documents with additional

Table 8.3: The mean retrieval performance for each collection when the click popularity scores are used to rank the FAQ document on an enriched FAQ document collection.

Weighting Model	Click Popularity	Collection	Test Evaluation Measure			
			MRR	MAP	Recall	P(Satisfied)
BM25F	*no click popularity*	non-enriched	0.4123	0.2988	0.6807	0.5128
	no click popularity	enriched	**0.5393**$*$	**0.4857**$*$	**0.9150**$*$	**0.6104** $*$
	with click popularity	enriched	0.4761	0.4157	0.8427	0.5606

terms from a query log in order to resolve the term mismatch problem. Table 8.3 summaries the results of our investigation. Our results in Table 8.3 show a significant improvement in retrieval performance in terms of *MRR*, *MAP* and *Recall* when the FAQ documents are enriched as denoted by $*$ (paired t-test, $p < 0.05$). Similarly, there is a significant improvement in the probability that any random user will be satisfied when using our FAQ retrieval system (*P(Satisfied)*). We use the bad abandonment statistics together with our proposed evaluation measure in Chapter 4 (Section 4.4) to estimate the probability that any random user will be satisfied when using our current system. However, we see a significant decrease in retrieval performance and the probability that any random user will be satisfied when we deploy the click popularity scores in the BM25F term weighting model. For example, there is a decrease in the probability that any random user will be satisfied when using our FAQ retrieval system from 0.6104 to 0.5606. Our results also suggest that our approach for resolving the term mismatch problem by enriching the FAQ documents with terms from a query log also generalise well on other datasets as we can see a significant improvement in recall from 0.6807 for the non-enriched collection to 0.9150 for the enriched collection.

Recall that in Section 8.2.1, we used PL2F to create a pool of 2205 FAQ documents to be judged by participants. In total, 906 of these FAQ documents were identified as either relevant and slightly relevant by the participants who took part in this study. This 906 FAQ documents were included in our query relevance file. In this study however, when we deployed BM25F term weighting model to retrieve the FAQ documents, we noticed that 77 FAQ documents that were included in the query relevance file did not appear in the top 5 retrieved document, resulting in 91.50% recall (829/906). Similarly, there was a total of 235 unjudged FAQ documents in the top 5 retrieved documents when we used BM25F term weighting model. Some of these unjudged FAQ documents might be relevant to the user queries and should have been included in the query relevance file. In future, we will use several term weighting models to increase a pool of FAQ documents to be judged by users.

Table 8.4: The mean probability that any random user will be satisfied when the missing content queries detection system is deployed in our FAQ retrieval system.

Retrieval Platform	Collection	Classification Accuracy(%)	P(Satisfied)
BM25F only	*enriched*	N/A	0.6104
BM25F with MCQs Detection	*enriched*	83.90	**0.7817** $\diamond$

8.5.2 Effects of the Missing Content Query Detection System On User Satisfaction

Another aspect that we investigate in this chapter is the effect of the missing content queries (*MCQs*) detection system on the probability that any random user will be satisfied when using our FAQ retrieval system (*C8-RQ2*). To answer this research question, we conduct our investigation using our best retrieval platform from Section 8.5.1. In particular, we deploy our *MCQs* detection system on an FAQ retrieval system that uses BM25F without the click popularity scores on an enriched FAQ document collection. Our *MCQs* detection system yielded a classification accuracy of 83.90%. The performance of our classifier also generalise well on this new dataset (*C8-RQ1*) since our best classifier earlier in Chapter 7 (Section 7.5) yielded roughly the same classification accuracy. For example, 312 out of 328 of the SMS queries that had the relevant FAQ documents in the top 5 retrieved documents were correctly classified as *non-MCQs*. Also, 58 out of 113 of the SMS queries that did not have the relevant FAQ documents in the top 5 retrieved documents were correctly classified *MCQs*. Our results in Table 8.4 show a significant improvement on the probability that any random user will be satisfied when we deploy the *MCQs* detection subsystem in our FAQ retrieval system as denoted by $\diamond$ (paired t-test, $p < 0.05$). In our evaluation, we only considered those queries that were classified as *non-MCQs*. Since our FAQ retrieval system does not engage the user in the quesion answering process when the SMS queries are classified as *MCQs*, these were not included in our evaluation.

8.5.3 Effects of Noisy SMS Queries on User Satisfaction

Table 8.5: The mean retrieval performance for each collection when the SMS queries are corrected for spelling errors.

Retrieval Platform	Test Queries	Collection	Test Evaluation Measure			
			MRR	MAP	Recall	P(Satisfied)
BM25F only	*Noisy SMS Queries*	non-enriched	0.3568	0.2618	0.6030	0.4518
	Clean SMS Queries	non-enriched	**0.4123**$\circledast$	**0.2988**$\circledast$	**0.6807**$\circledast$	**0.5128** $\circledast$
BM25F with MCQs Detection	*Noisy SMS Queries*	enriched	0.5933	0.5179	0.8055	0.6808
	Clean SMS Queries	enriched	**0.6356**$\triangleleft$	**0.5732**$\triangleleft$	**0.9145**$\triangleleft$	**0.7817** $\triangleleft$

The other aspect that we investigate is whether we can improve the retrieval performance and

the probability that any random user will be satisfied by correcting spelling errors (*C8-RQ3*). We carried out our evaluation using two different testing sets, one with SMS queries that were manually corrected for spelling errors and the other with SMS queries that were not corrected for spelling errors. Two different retrieval platforms were used in our evaluation. The first retrieval platform only deployed BM25F term weighting model on a non-enriched FAQ document collection. The second retrieval platform deployed BM25F term weighting model with the *MCQs* detection subsystem on an enriched FAQ document collection. Our results in Table 8.5 show a significant improvement in retrieval performance in terms of *MRR*, *MAP* and *Recall* when noisy SMS queries are corrected for spelling errors as denoted by $\circledast$ and $\triangleleft$ (paired t-test, $p < 0.05$). Similarly, we see a significant improvement in the probability that any random user will be satisfied when using our FAQ retrieval system.

8.6 Discussion and Conclusions

In this chapter, we used a different testing set to investigate whether the previous results generalise on other datasets, including when we combine our sub-systems. Recall that in Chapter 5, we saw a significant improvement in retrieval performance in terms of *MRR*, *MAP* and *Recall*, and an improvement in the probability that any random user will be satisfied when we enriched the FAQ documents with terms from a query log in order to resolve the term mismatch problem between the user's queries and the relevant FAQ document in the collection. Similar results were also observed in Table 8.5.1, suggesting that our enrichment strategy generalise well on other datasets (*C8-RQ1*). In Chapter 6, we saw a significant improvement in retrieval performance and the probability that any random user will be satisfied when we rank the FAQ documents based on how often they have been previously identified as relevant for a particular query term t. However, our results in Table 8.5.1 show a significant decrease in retrieval performance and the probability that any random user will be satisfied when we deploy the click popularity score on an enriched collection. These results suggest that our method of incorporating the click popularity scores does not generalise well on different collections.

Earlier in Chapter 7, we developed a *MCQs* detection subsystem in order to filter out those queries that do not have the relevant FAQ documents in the FAQ document collection. In this chapter, we also investigate the effect of this *MCQs* detection subsystem on the probability that any random user will be satisfied (*C8-RQ2*). In particular, we investigate whether deploying this *MCQs* detection subsystem will reduce the number of unnecessary iterations with the users by informing them when there are no relevant FAQ documents in the FAQ document collection. In our evaluation, we saw a significant increase in the probability that any random user will be satisfied when the *MCQs* detection subsystem is deployed because

51.3% of the *MCQs* were correctly classified, hence reducing unnecessary iterations. Our results suggest that we can further improve the probability that any random user will be satisfied by improving the classification accuracy of our *MCQs* detection subsystem.

The other aspect that we investigate in this thesis is whether we can improve the retrieval performance and the probability that any random user will be satisfied by correcting spelling errors (*C8-RQ3*). Our results in Table 8.5 suggests that such a subsystem is crucial as there was a significant improvement in retrieval performance and the probability that any random user will be satisfied when the SMS queries were manually corrected for spelling errors. In future, we will develop an automatic spelling correction subsystem for our FAQ retrieval system. In Chapter 9, we provide a summary of this thesis, detailing all the achievements made and future directions for research.

Chapter 9

Conclusions and Future Work

9.1 Thesis Contributions and Conclusions

This thesis proposed a semi-automated FAQ retrieval system for HIV/AIDS, which is designed to allow users of low-end mobile phones to be able to search an automated FAQ retrieval system through SMS messages. In order to improve the probability that any random user will be satisfied when using our FAQ retrieval system, the thesis proposed to use information from previous searches to alleviate the FAQ document collection deficiency problems (see Section 1.3.2) in our FAQ retrieval system so that users are presented with the correct FAQ documents after a few iterations. Our main objective is to shorten the number of iterations (search length) between the users and our FAQ retrieval system so that users do not abandon the iterative search process before their information need has been satisfied. The remainder of this section discusses the contributions and conclusions of this thesis.The main contributions of this thesis are as follows:

- In Chapter 3, we presented the dataset that we used to validate our thesis statement. In particular, we conduct a user study to gather queries and their relevance judgements from potential users of our FAQ retrieval system in Botswana (Section 3.2). We later use this query log and the query relevance judgements in conjunction with our FAQ document collection to develop a test collection to facilitate our evaluation of the various subsystems that we develop in this thesis (Section 3.3). Furthermore, we conduct another user study with an on-line community of users in order to validate the quality of the query relevance judgements that we collected from our earlier study in Botswana (see Section 3.4).

- In Chapter 4, we describe our baseline iterative FAQ retrieval system. The main building blocks (subsystems) of our FAQ retrieval system are described (Section 4.3). A full study to investigate the number of iterations that users are willing to tolerate before

abandoning this iterative search process was conducted (*Chapter 4-Research Question One (C4-RQ1)*), with the aim of using the bad abandonment statistics from this study for future evaluation of our FAQ retrieval system. In particular, we used this bad abandonment statistics to come up with an evaluation measure for estimating the probability that any random user will be satisfied when using our FAQ retrieval system (Section 4.4.6). Our results suggest that a majority of users tolerate up to 3 iterations (see Section 4.4.5). These results suggest that we can improve the probability that any random user will be satisfied when using our FAQ retrieval system by ranking as many FAQ documents as possible amongst the top 3 FAQ documents.

In addition, we thoroughly investigate whether the search length of previous searches influence the search length of subsequent searches (*Chapter 4-Research Question Two (C4-RQ2)*). In our investigation, we observed that participants who were initially exposed to a shorter search length abandoned quicker when they were given a test set with a longer search length, suggesting that previous searches can significantly influence subsequent behaviour (C4-RQ2).

Moreover, we carried out an empirical investigation and evaluation to determine the most appropriate way of representing the FAQ documents in the data source (Section 4.5). In particular, we investigated whether we can improve the overall retrieval performance by indexing the question part only (*Chapter 4-Research Question Three (C4-RQ3)*). Our results suggest that indexing both the question and the answer part can help improve the overall retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system (see Section 4.5.3). We also used various term weighting models (BM25, PL2, DPH, TF-IDF and Hiemstra languge mode) to investigate whether stopword removal can improve the overall retrieval performance (*Chapter 4-Research Question Four (C4-RQ4)*). This empirical evaluation also enabled us to determine a suitable baseline term weighting model to use in our FAQ retrieval system. Our results suggest that we can improve the overall retrieval performance and the probability that any random user will be satisfied by removing stopwords (see Section 4.5.3). BM25 was chosen in our baseline FAQ retrieval system because it yielded the best retrieval performance compared to other term weighting models when stopwords are removed (see Section 4.5.3). It also performed generally well across the different evaluation measures.

- The first FAQ document collection deficiency problem was addressed in Chapter 5, namely the term mismatch problem. A novel template-based approach that uses queries from a query log for which the true relevant FAQ document are known to enrich the FAQ documents with additional terms in order to alleviate the term mismatch problem was proposed (see Section 5.3, *Chapter 5-Research Question One (C5-RQ1)*). These

terms are added as a separate field in a field-based model using our proposed enrichment strategies, namely the Term Frequency and the Term Occurrence enrichment strategies (see Section 5.3). The Term Occurrence approach ignores the number of times a term occurs in the queries when enriching the FAQ documents while the Term Frequency consider the number of times a term occurs in the queries when enriching the FAQ documents. We thoroughly investigate our enrichment strategies using three different field-based models (BM25F, PL2F and DPHF) in order to determine whether we can improve the overall recall and the probability that any random user will be satisfied when we take into consideration the number of times a term occurs in the queries when enriching the FAQ documents (*Chapter 5-Research Question Two (C5-RQ2)*). Moreover, we investigate whether increasing the number of queries used in enriching the FAQ documents increases the overall recall and the probability that any random user will be satisfied (*Chapter 5-Research Question Three (C5-RQ3)*).

Our results show that we can improve the overall recall and the probability that any random user will be satisfied by enriching the FAQ documents with additional terms from queries for which the true relevant FAQ document are known (see Section 5.6, (C5-RQ1)). An increase in recall suggest that the term mismatch problem has been resolved. When we compare our two different enrichment strategies, we found that the Term Frequency enrichment approach produced higher MRR and recall values compared to the Term Occurrence approach (see Section 5.6). Our findings suggests that it is important to take into consideration the number of times a term occurs in the queries when enriching the FAQ documents (*Chapter 5-Research Question Two (C5-RQ2)*). In addition, we found that increasing the number of queries used in enriching the FAQ documents increases the overall recall and the probability that any random user will be satisfied across the different field-based term weighting models (BM25F, PL2F and DPHF) that we deployed in this study (see Section 5.6, *Chapter 5-Research Question Three (C5-RQ3)*).

- In Chapter 6, we address another FAQ document collection deficiency problem by proposing a new ranking function that selectively ranks the FAQ documents based on how often they have been previously identified as relevant by users for a particular query term t. Previous works (Sneiders, 2009) and our query log analysis (see Section 3.2.3) suggest that users are only interested in a subset of the FAQ documents in the FAQ document collection. Our aim in this chapter is to ensure that the FAQ documents that are popular to users and share query terms with the less popular FAQ documents are always ranked higher. In our proposed approach, we modify the RJS term weight of the BM25 term weighting model by multiplying it with the click popularity score of a query term t on an FAQ document d (see Section 6.2). In addition, we proposed a second approach for ranking the FAQ documents based on how often

they have been previously identified as relevant by users, without taking into consideration the query terms associated with those documents (see Section 6.3). Moreover, we proposed a third method of ranking the FAQ documents by incorporating the click popularity scores of an FAQ document into the scoring process using a learning to rank technique (see Section 6.4).

In our evaluation, we found that we can improve the probability that any random user will be satisfied by ranking the FAQ documents based on how often they have been previously identified as relevant by users for a particular query term t (see Section 6.6, *Chapter 6-Research Question One (C6-RQ1)*). However, there was no improvement on user satisfaction when the FAQ documents are ranked based on how often they have been previously identified as relevant by users without taking into consideration the query terms associated with those documents (see Section 6.6, *Chapter 6-Research Question Two (C6-RQ2)*). Furthermore, we see a significant improvement in the retrieval performance when we incorporate the click popularity scores of an FAQ document into the scoring process using a learning to rank technique.

- In Chapter 7, we empirically evaluated several feature sets in order to determine the best combination of features for building a model that yields the best classification accuracy in identifying the MCQs and the non-MCQs. The different features sets used in this study are described in Section 7.3. We experimentally examined the classification accuracy of the individual feature sets and our result show that, classifiers which use feature sets represented by a vector of word counts from the text contained in the query strings are often more effective compared to other feature sets (see Section 7.5, *Chapter 7-Research Question One (C7-RQ1)*). We also found that classifiers that combined all the features sets were more effective compared to when we use the individual feature sets (see Section 7.5, Chapter 7-Research Question Two (C7-RQ2)). Furthermore, our results show that increasing the training set size of our classifiers can help improve the classification accuracy (*Chapter 7-Research Question Three (C7-RQ3)*).

- In Chapter 8, we investigate whether the previous results and findings generalise on a second dataset, including when we combine our sub-systems (*Chapter 8-Research Question One (C8-RQ1)*). In particular, we develop and train our subsystems using the previous dataset created in Chapter 3. We then create another testing set to evaluate our various subsystems, which were trained using our old dataset (see Section 8.2.1). Our results show comparable results with our earlier results in Chapter 5, 6 and 7, suggesting that our previous results and findings generalise on a new dataset (C8-RQ1). Similarly, when we combine the missing content queries detection subsystem and our enrichment approach for resolving the term mismatch problem, we observed a significant improvement in the probability that any random user will be satisfied when

using our FAQ retrieval system (see Section 8.5.2, *Chapter 8-Research Question Two (C8-RQ2)*). However, incorporating the click popularity score of a query term t on an FAQ document d on an enriched collection degrades the retrieval performance and the probability that any random user will be satisfied (see Section 8.5.1). In addition, we also investigated whether correcting spelling errors can help improve the retrieval performance and the probability that any random user will be satisfied when using our FAQ retrieval system (*Chapter 8-Research Question Three (C8-RQ3)*). Our results show a significant improvement in retrieval performance and the probability that any random user will be satisfied when the SMS queries are corrected for spelling errors (C8-RQ3).

9.2 Directions for Future Work

This section discusses several directions for future work related to, or stemming from this thesis.

Developing a Stemmer for Setswana Language The main objective of the IHISM Project is develop an FAQ retrieval system that could respond to any user query that is written in either English or Setswana. In this thesis however, we developed a mono-lingual FAQ retrieval system that could respond to English SMS queries. One of the reasons why we did not develop an FAQ retrieval system that could respond to Setswana queries is that there is no stemmer for Setswana language. Therefore, future work could be directed towards developing a stemmer for Setswana language to use in our multi-lingual FAQ retrieval system, which can respond to both English and Setswana queries.

Normalising SMS queries: In Chapter 1 (Section 1.3.1), we identified spelling errors as one of the aspects to consider when developing an SMS-Based FAQ retrieval system. Indeed, In Chapter 8 (Section 8.5.3), we discovered that correcting spelling errors can help improve the retrieval performance and user satisfaction in our FAQ retrieval system. In Chapter (Section 2.3.5), we also conducted a literatures review on the SMS normalisation techniques, with a particular focus on the FIRE SMS-Based FAQ retrieval task. However, we did not conduct an empirical evaluation to determine the effectiveness of the different SMS normalisation techniques proposed. Similarly, although there was a standard dataset that was provided to evaluate the overall retrieval performance of the participating systems at the FIRE evaluation forum, no formal evaluation was conducted to assess the effectiveness of the different SMS normalisation techniques proposed. Therefore, future works can be directed towards evaluating the various SMS normalisation techniques on a common FAQ retrieval platform. Our aim is to conduct a thorough empirical evaluation of the different SMS normalisation techniques to use in our FAQ retrieval system before we deploy the system to be used by the general public in Botswana.

Evaluating the usability of the different result presentation and search strategies: In this thesis, we proposed an iterative retrieval search strategy for SMS-Based FAQ retrieval without assessing the usability of our proposed strategy and how the results are presented to users. Therefore, future work can be directed towards evaluating the usability of the different result presentation and search strategies for SMS-Based FAQ retrieval. In particular, we identified the following result presentation and search strategies to be considered for future investigation:

- Iterative interaction search strategy that we proposed in Chapter 4 (Section 4.3). At each iteration, only one question-answer pair is returned and displayed to the user. The

main disadvantage of this approach is that, it is likely that users may iterate longer with a system if the relevant FAQ document is lowly ranked.

- Returning a ranked list of the question parts of the top 5 ranked FAQ documents for each user query. From this ranked list, a user selects the one he or she believe is related to the submitted query to retrieve the answer part. The main advantage of this approach is that it is likely to reduce the number of iterations between the user and the FAQ retrieval system if a user can identify lowly ranked FAQ documents as related to the original query. However, since only the question part is displayed, it is likely that some users may ignore some questions returned even though they are related to the original query because of the lexical difference between the query and the relevant question part.

- Returning a ranked list of the top 5 ranked FAQ documents for each user query. The main disadvantage of this approach in low-end devices as earlier discussed in Chapter 1 (Section 1.3.3) is that users may find it difficult to navigate and identify the relevant FAQ document from several SMS messages that are returned to the user for each user query.

Appendix A

Selection of SMS Queries

Table A.1: Selection of non-Missing Content Queries provided by participants in Botswana.

non-Missing Content Queries
how is tuberculosis related to the hiv infection?
can you be infected by sharing razors
how can you prevent yourself from getting aids
which diet is good for HIV/AIDS patient
can aids kill oneself after an immediate infection
which treatment is given to people who are affected with hiv/aids?
Is it necessary to take supplements e.g Vitamins alongside ARV's
If i skip a day without taking my ARVs what do i do?
Is it true that women are more at risk of getting HIV than men?
can you still have children if you are HIV positive?
Where can people get access of condoms in our country?

Table A.2: Selection of Missing Content Queries provided by participants in Botswana.

Missing Content Queries
where did the first person to get infected get the virus from
was HIV/AIDS testing that common when it was first discovered
how many children were saved by the introduction of PMTCT
How doe circumcision reduce the chances of HIV and AIDS infection?
Where did HIV originate from?
is it true that there is AIDS CURE in uganda
which age group is mostly infected by aids
why can we forced to get tested
can hiv get through witchcraft
is there still a stigma in relation to aids
is hiv aids a disease for both animal and human

Bibliography

Adesina, A., Agbele, K., Abidoye, A., and Nyongesa, H. (2014). Text messaging and retrieval techniques for a mobile health information system. *Journal of Information Science*, pages 1–13.

Adesina, A. and Nyongesa, H. (2013). A Mobile-Health Information Access System. In Volkwyn, R., editor, *Proceedings of Southern African Telecommunication Networks and Applications Conference*, pages 191–, Stellenbosch, South Africa.

Alonso, O. and Baeza-Yates, R. (2011). Design and implementation of relevance assessments using crowdsourcing. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 153–164, Berlin, Heidelberg. Springer-Verlag.

Alonso, O. and Mizzaro, S. (2012). Using crowdsourcing for trec relevance assessment. *Information Processing and Management*, 48(6):1053–1066.

Amati, G. (2006). Frequentist and bayesian approach to information retrieval. In *Proceedings of the 28th European Conference on Advances in Information Retrieval*, ECIR'06, pages 13–24, Berlin, Heidelberg. Springer-Verlag.

Amati, G., Ambrosi, E., Bianchi, M., Gaibisso, C., and Gambosi, G. (2007). FUB, IASI-CNR and University of Tor Vergata at TREC 2007 Blog Track. In *Proceedings of the 16th Text REtrieval Conference (TREC-2007)*, pages 1–10, Gaithersburg, Md., USA. Text REtrieval Conference (TREC).

Anderson, G., Asare, S., Ayalew, Y., Garg, D., Gopolang, B., Masizana-Katongo, A., Mogotlhwane, O., Mpoeleng, D., and Nyongesa, H. (2007a). Towards a Bilingual SMS Parser for HIV and AIDS Information Retrieval in Botswana. In *Proceedings of the International Conference on Information and Communication Technologies and Development*, pages 1–5, Los Alamitos, CA, USA. IEEE Xplore.

Anderson, G., Asare, S., Eyitayo, A., Eyitayo, O., Mpoeleng, D., Nkgau, T., Nyongesa, H., and Ogwu, F. (2007b). Information Storage and Retrieval Techniques for Mobile Healthcare. In *Proceedings of the Advanced Communication Technology International Conference*, volume 2, pages 1096–1100, Los Alamitos, CA, USA. IEEE Xplore.

Andrenucci, A. and Sneiders, E. (2005). Automated Question Answering: Review of the Main Approaches. In *Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2 - Volume 02*, pages 514–519, Washington, DC, USA. IEEE Computer Society.

Arguello, J., Elsas, J., Callan, J., and Carbonell, J. (2008). Document representation and query expansion models for blog recommendation. In *Proc. of the 2nd International Conference on Weblogs and Social Media (ICWSM)*.

Athenikos, S. and Han, H. (2010). Biomedical question answering: A survey. *Computer Methods and Programs in Biomedicine*, 99(1):1–24.

Attar, R. and Fraenkel, A. (1977). Local feedback in full-text retrieval systems. *Journal of the ACM*, 24(3):397–417.

Aw, A., Zhang, M., Xiao, J., and Su, J. (2006). A Phrase-Based Statistical Model for SMS Text Normalization. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 33–40, Stroudsburg, PA, USA. Association for Computational Linguistics.

Baeza-Yates, R. (2005). Applications of web query mining. In *Proceedings of the 27th European Conference on Advances in Information Retrieval Research*, ECIR'05, pages 7–22, Berlin, Heidelberg. Springer-Verlag.

Baeza-Yates, R., Hurtado, C., and Mendoza, M. (2004). Query recommendation using query logs in search engines. In *Proceedings of the 2004 International Conference on Current Trends in Database Technology*, EDBT'04, pages 588–596, Berlin, Heidelberg. Springer-Verlag.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). Modern Information Retrieval. *Addison Wesley*.

Berger, A., Caruana, R., Cohn, D., Freitag, D., and Mittal, V. (2000). Bridging the Lexical Chasm: Statistical Approaches to Answer-finding. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 192–199, New York, NY, USA. ACM.

Bhattacharya, S., Tran, H., and Srinivasan, P. (2013). Data-Driven Methods for SMS-Based FAQ Retrieval. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 104–118, Berlin, Heidelberg. Springer-Verlag.

Bollinger, L. and Stover, J. (1999). Economic Impact of AIDS in Botswana. *Policy Project*, pages 1–11.

Bornman, E. (2012). The Mobile Phone in Africa : Has it Become a Highway to the Information Society or not? *Contemporary Educational Technology*, 3(4):278–292.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Brill, E., Lin, J., Banko, M., Dumais, S., and Ng, A. (2001). Data-Intensive Question Answering. In *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*, pages 183–189, Gaithersburg, MD, USA. National Institute of Standards and Technology (NIST).

Brinkmeier, M. (2006). Pagerank revisited. *ACM Transactions on Internet Technology*, 6(3):282–301.

Brown, P., Pietra, V. D., Pietra, S. D., and Mercer, R. (1993). The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics - Special Issue on Using Large Corpora*, 19(2):263–311.

Burges, C., Ragno, R., and Le, Q. (2007). Learning to rank with non-smooth cost functions. In *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA.

Burke, R., Hammond, K., V. Kulyukin, S. L., Tomuro, N., and Schoenberg, S. (1997). Question Answering from Frequently Asked Question Files Experiences with the FAQ FINDER System. *AI Magazine*, 18(2):57 – 66.

Byun, J., Lee, S., Song, Y., and Rim, H. (2008). Two Phase Model for SMS Text Messages Refinement. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 7–8, California, USA. AAAI Press.

Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 44(1):1:1–1:50.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 161–168, New York, NY, USA. ACM.

Casellas, N., Casanovas, P., Vallbé, J.-J., Poblet, M., Blázquez, M., Contreras, J., López-Cobo, J., and Benjamins, V. (2007). Semantic Enhancement for Legal Information Retrieval: Iuriservice Performance. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, pages 49–57, New York, NY, USA. ACM.

Cerny, V. (1985). Thermodynamical Approach to the Traveling Salesman Problem: An efficient Simulation Algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for Support Vector Machine. *ACM Transactions on Intelligent Systems Technology (TIST)*, 2(3):27:1–27:27.

Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P. (2009). Expected Reciprocal Rank for Graded Relevance. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 621–630, New York, NY, USA. ACM.

Choudhury, M., Saraf, R., Jain, V., Mukherjee, A., Sarkar, S., and Basu, A. (2007). Investigation and Modeling of the Structure of Texting Language. *International Journal on Document Analysis and Recognition*, 10(3):157–174.

Chuklin, A. and Serdyukov, P. (2012). Good Abandonments in Factoid Queries. In *Proceedings of the 21st International Conference Companion on World Wide Web*, pages 483–484, New York, NY, USA. ACM.

Cleverdon, C. and Kean, M. (1968). Factors determining the performance of indexing systems. Aslib Cranfield Research Project, Cranfield, England.

Clough, P. and Sanderson, M. (2013). Evaluating the performance of information retrieval systems using test collection. *Information Research*, 18(2).

Contractor, D., Faruquie, T., and Subramaniam, L. (2010). Unsupervised Cleansing of Noisy Text. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 189–196, Stroudsburg, PA, USA. Association for Computational Linguistics.

Contractor, D., Subramaniam, L., Deepak, P., and Mittal, A. (2013). Text Retrieval Using SMS Queries: Datasets and Overview of FIRE 2011 Track on SMS-Based FAQ Retrieval. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 86–99, Berlin, Heidelberg. Springer-Verlag.

Cooper, W. (1968). Expected search length: A single measure of retrieval effectiveness based on the weak ordering action of retrieval systems. *American Documentation*, 19(1):30–41.

Coulthard, M. (2004). Author identification, idiolect, and linguistic uniqueness. *Applied Linguistics*, 25(4):431–447.

Cronen-Townsend, S., Zhou, Y., and Croft, W. (2002). Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 299–306, New York, NY, USA. ACM.

Daelemans, W., Zavrel, J., Sloot, K., and Bosch, A. (2002). TiMBL: Tilburg Memory-Based Learner - version 4.3 - Reference Guide.

Dai, L., Liu, B., Xia, Y., and Wu, S. (2008). Measuring semantic similarity between words using hownet. In *Computer Science and Information Technology, 2008. ICCSIT '08. International Conference on*, pages 601–605.

Dang, H., Lin, J., and Kelly, D. (2006). Overview of the trec 2006 question answering track 99. In Voorhees, E. M. and Buckland, L. P., editors, *TREC*, volume Special Publication 500-272. National Institute of Standards and Technology (NIST).

Deloitte (2012). Sub-Saharan Africa Mobile Observatory 2012. *GSMA Intelligence (GSMA),UK, Report*, pages 1 – 288.

Diriye, A., White, R., Buscher, G., and Dumais, S. (2012). Leaving So Soon?: Understanding and Predicting Web Search Abandonment Rationales. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 1025–1034, New York, NY, USA. ACM.

Dixon, S., McDonald, S., and Roberts, J. (2002). The impact of HIV and AIDS on Africa's Economic Development. *BMJ: British Medical Journal*, 324(7331):232–234.

Dong, Z. and Dong, Q. (2006). *Hownet And the Computation of Meaning*. World Scientific Publishing Co., Inc., River Edge, NJ, USA.

Donner, J. (2008). Research Approaches to Mobile Use in the Developing World: A Review of the Literature. *The Information Society*, 24(3):140–159.

Duineveld, A., Stoter, R., Weiden, M., Kenepa, B., and Benjamins, V. (2000). WonderTools? A Comparative Study of Ontological Engineering Tools. *International Journal of Human-Computer Studies*, 52(6):1111 – 1133.

Esuli, A., Fagni, T., and Sebastiani, F. (2008). Boosting Multi-label Hierarchical Text Categorization. *Information Retrieval*, 11(4):287–313.

Fang, H. (2008). A Re-Examination of Query Expansion Using Lexical Resources. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, pages 139–147, Stroudsburg, PA, USA. Association for Computer Linguistics.

Fang, W., Guifa, T., Lisheng, R., and JianBin, M. (2008). Research on Mechanism of Agricultural FAQ Retrieval Based on Ontology. In *Proceedings of Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing,*, pages 955–958.

Ferguson, P., O'Hare, N., Lanagan, J., Smeaton, A., McCarthy, K., Phelan, O., and Smyth, B. (2011). CALRITY at the TREC 2011 Microblog Track. In *Proceedings of the 20th TREC Conference*, pages 1–6, Gaithersburg, Md., USA. Text REtrieval Conference (TREC).

Findlater, L., Balakrishnan, R., and Toyama, K. (2009). Comparing semiliterate and illiterate users' ability to transition from audio+text to text-only interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 1751–1760, New York, NY, USA. ACM.

Fleiss, J., Levin, B., and Paik, M. (2004). *Statistical Methods for Rates and Proportions*. Wiley Series in Probability and Statistics. Wiley.

Frakes, W. and Baeza-Yates, R. (1992). Information Retrieval: Data Structures and Algorithms. *Prentice-Hall*.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 28(2):337–407.

Frøkjær, E., Hertzum, M., and Hornbæk, K. (2000). Measuring usability: Are effectiveness, efficiency, and satisfaction really correlated? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, pages 345–352, New York, NY, USA. ACM.

Gauch, S., Wang, J., and Rachakonda, S. (1999). A corpus analysis approach for automatic query expansion and its extension to multiple databases. *ACM Transactions on Information Systems*, 17(3):250–269.

GSMA-Intelligence (2013). Sub-Saharan Africa Mobile Economy 2013. *GSMA Intelligence (GSMA),UK, Report*, pages 1 – 90.

Gupta, A. (2013). Mapping SMSes to Plain Text FAQs. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 157–162, Berlin, Heidelberg. Springer-Verlag.

Hammond, K., Burke, R., Martin, C., and Lytinen, S. (1995). FAQ Finder: A Case-Based Approach to Knowledge Navigation. In *Proceedings of the 1995 IEEE Conference on Artificial Intelligence Applications*, pages 80–86, Los Alamitos, CA, USA. IEEE Computer Society.

Harman, D. (2010). Is the Cranfield Paradigm Outdated? In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1–1, New York, NY, USA. ACM.

Hauff, C., Murdock, V., and Baeza-Yates, R. (2008). Improved query difficulty prediction for the web. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 439–448, New York, NY, USA. ACM.

He, B. and Ounis, I. (2004). Inferring Query Performance Using Pre-retrieval Predictors. In *Proceedings of the String Processing and Information Retrieval*, pages 43–54, Berlin, Heidelberg. Springer-Verlag.

He, B. and Ounis, I. (2006). Query Performance Prediction. *Information Systems*, 31(7):585–594.

He, B. and Ounis, I. (2007). Combining Fields for Query Expansion and Adaptive Query Expansion. *Information Processing and Management: an International Journal*, 43(5):1294–1307.

Hiemstra, D. (1998). A linguistically motivated probabilistic model of information retrieval. In *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries*, ECDL '98, pages 569–584, London, UK, UK. Springer-Verlag.

Hiemstra, D. (2001). Using Language Models or Information Retrieval. *University of Twente,The Netherlands, PhD Thesis*, pages 1 – 178.

Hirschman, L. and Gaizauskas, R. (2001). Natural Language Question Answering: The View from Here. *Natural Language Engineering*, 7(4):275–300.

Hogan, D., Leveling, J., Wang, H., Ferguson, P., and Gurrin, C. (2011). DCU@FIRE 2011: SMS-based FAQ retrieval. In *FIRE 2011, 3rd Workshop of the Forum for Information Retrieval Evaluation, 2-4 December, IIT Bombay*, pages 34–42.

Hsu, C.-W., Chang, C.-C., and C.-J., L. (2010). A Practical Guide to Support Vector Classification.

Hu, J., Deng, W., and Guo, J. (2006). Improving retrieval performance by global analysis. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 2, pages 703–706.

Huang, J., White, R., and Dumais, S. (2011). No Clicks, No Problem: Using Cursor Movements to Understand and Improve Search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1225–1234, New York, NY, USA. ACM.

Huang, X. and Soergel, D. (2004). Relevance judges understanding of topical relevance types: An explication of an enriched concept of topical relevance. In *Proceedings of the Annual Meeting of the American Society for Information Science and Technology*, pages 156–167. John Wiley and Sons, Inc.

Jardine, N. and Rijsbergen, C. V. (1971). The Use of Hierarchic Clustering in Information Retrieval. *Information Storage and Retrieval*, 7(5):217 – 240.

Jeon, J. (2007). Searching Question and Answer Archives. *University of Massachusetts,USA, PhD Thesis*, pages 1 – 131.

Jeon, J., Croft, W., and Lee, J. (2005). Finding Similar Questions in Large Question and Answer Archives. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pages 84–90, New York, NY, USA. ACM.

John, G. and Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Kandala, N.-B., Campbell, E., Rakgoasi, S., Madi-Segwagwe, B., and Fako, T. (2012). The Geography of HIV/AIDS Prevalence Rates in Botswana. *HIV/AIDS - Research and Palliative Care*, 2012(4):95102.

Kantor, P. and Voorhees, E. (2000). The trec-5 confusion track: Comparing retrieval methods for scanned text. *Information Retrieval*, 2(2-3):165–176.

Katz, B. (1997). Annotating the World Wide Web using Natural Language. In *Proceedings of the 5th RIAO Conference on Computer Assisted Information Searching on the Internet*, pages 136–159.

Katz, B., Felshin, S., Yuret, D., Ibrahim, A., Lin, J., Marton, G., McFarland, A., and Temelkuran, B. (2002). Omnibase: Uniform Access to Heterogeneous Data for Question Answering. In *Proceedings of the 7th International Workshop on Applications of Natural Language to Information Systems*, pages 230–234, Berlin, Heidelberg. Springer-Verlag.

Kazai, G. (2011). In search of quality in crowdsourcing for search engine evaluation. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 165–176, Berlin, Heidelberg. Springer-Verlag.

Kelly, D. (2009). Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, 3(1–2):1–224.

Kent, A., Berry, M., Luehrs, F., and Perry, J. (1955). Machine literature searching viii. operational criteria for designing information retrieval systems. *American Documentation*, 6(2):93–101.

Kim, H., Lee, H., and Seo, J. (2007). A Reliable FAQ Retrieval System Using a Query Log Classification Technique Based on Latent Semantic Analysis. *Information Processing and Management*, 43(2):420 – 430.

Kim, H. and Seo, J. (2006). High-Performance FAQ Retrieval Using an Automatic Clustering Method of Query Logs. *Information Processing and Management*, 42(3):650–661.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598):671–680.

Kothari, G., Negi, S., Faruquie, T., Chakaravarthy, V., and Subramaniam, L. (2009). SMS Based Interface for FAQ Retrieval. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, pages 852–860, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koumpouri, A. and Simaki, V. (2012). Queries Without Clicks: Evaluating Retrieval Effectiveness Based on User Feedback. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1133–1134, New York, NY, USA. ACM.

Koza, J. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.

Kraft, R. and Zien, J. (2004). Mining anchor text for query refinement. In *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, pages 666–674, New York, NY, USA. ACM.

Kwok, K. and Chan, M. (1998). Improving two-stage ad-hoc retrieval for short queries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 250–256, New York, NY, USA. ACM.

Lam-Adesina, A. and Jones, G. (2001). Applying summarization techniques for term selection in relevance feedback. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 1–9, New York, NY, USA. ACM.

Landis, J. and Koch, G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Lavrenko, V. and Croft, W. (2001). Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 120–127, New York, NY, USA. ACM.

Lee, D.-G., Rim, H.-C., and Yook, D. (2007). Automatic Word Spacing Using Probabilistic Models Based on Character N-Grams. *IEEE Intelligent Systems*, 22(1):28–35.

Leveling, J. (2012). On the Effect of Stopword Removal for SMS-Based FAQ Retrieval. In *Proceedings of the 17th International Conference on Applications of Natural Language Processing and Information Systems*, pages 128–139, Berlin, Heidelberg. Springer-Verlag.

Li, J., Huffman, S., and Tokuda, A. (2009). Good Abandonment in Mobile and PC Internet Search. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–50, New York, NY, USA. ACM.

Liu, T.-Y. (2009). Learning to Rank for Information Retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331.

Losada, D. and Azzopardi, L. (2008). An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11(2):109–138.

Macdonald, C., Plachouras, V., He, B., Lioma, C., and Ounis, I. (2006). University of Glasgow at WebCLEF 2005: Experiments in Per-field Normalisation and Language Specific Stemming. In *Proceedings of the 6th International Conference on Cross-Language Evalution Forum: Accessing Multilingual Information Repositories*, pages 898–907, Berlin, Heidelberg. Springer-Verlag.

Macdonald, C., Santos, R., and Ounis, I. (2013a). The whens and hows of learning to rank for web search. *Information Retrieval*, 16(5):584–628.

Macdonald, C., Santos, R., Ounis, I., and He, B. (2013b). About learning models with multiple query-dependent features. *ACM Transactions on Information Systems (TOIS)*, 31(3):11:1–11:39.

Manning, C., Raghavan, P., and Schtze, H. (2008). Introduction to Information Retrieval. *Cambridge City Press*.

Medhi, I., Ratan, A., and Toyama, K. (2009). Mobile-Banking Adoption and Usage by Low-Literate, Low-Income Users in the Developing World. In *Proceedings of the 3rd International Conference on Internationalization, Design and Global Development: Held As Part of HCI International*, pages 485–494, Berlin, Heidelberg. Springer-Verlag.

Metzler, D. and Croft, W. (2005). Analysis of Statistical Question Classification for Fact-Based Questions. *Information Retrieval*, 8(3):481–504.

Metzler, D. and Croft, W. (2007). Linear feature-based models for information retrieval. *Information Retrieval*, 10(3):257–274.

Miller, G. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.

Milne, D., Witten, I., and Nichols, D. (2007). A knowledge-based search engine powered by wikipedia. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 445–454, New York, NY, USA. ACM.

Minker, J., Wilson, G., and Zimmerman, B. (1972). An evaluation of query expansion by the addition of clustered terms for a document retrieval system. *Information Storage and Retrieval*, 8(6):329 – 348.

Mollá, D. and Vicedo, J. (2007). Question answering in restricted domains: An overview. *Computational Linguistics.*, 33(1):41–61.

Molomo, B. (2009). The Second National Stategic Framework for HIV and AIDS: 2010-2016. *National AIDS Coordinating Agency*, pages 1–41.

Moreo, A., Eisman, E., Castro, J., and Zurita, J. (2013). Learning Regular Expressions to Template-Based FAQ Retrieval Systems. *Knowledge-Based Systems*, 53(0):108 – 128.

Moreo, A., Navarro, M., Castro, J., and Zurita, J. (2012a). A High-Performance FAQ Retrieval Method Using Minimal Differentiator Expressions. *Knowledge-Based Systems*, 36:9–20.

Moreo, A., Romero, M., Castro, J., and Zurita, J. (2012b). FAQtory: A Framework to Provide High-Quality FAQ Retrieval Systems. *Expert Systems with Applications*, 39(14):11525 – 11534.

Oard, D. and Dorr, B. (1996). A Survey of Multilingual Text Retrieval. Technical report, University of Maryland at College Park, College Park, MD, USA.

Oleksandr, K. and Marie-Francine, M. (2011). A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412 – 5434.

Otlogetswe, T. (2008). Corpus design for Setswana lexicography. *University of Pretoria,SA, PhD Thesis*, pages 1 – 288.

Ounis, I., Amati, G., V., P., He, B., Macdonald, C., and Johnson (2005). Terrier Information Retrieval Platform. In *Proceedings of the 27th European Conference on IR Research*, volume 3408 of *Lecture Notes in Computer Science*, pages 517–519, Berlin, Heidelberg. Springer-Verlag.

Park, L. and Ramamohanarao, K. (2007). Query expansion using a collection dependent probabilistic latent semantic thesaurus. In *Advances in Knowledge Discovery and Data Mining*, volume 4426 of *Lecture Notes in Computer Science*, pages 224–235. Springer Berlin Heidelberg.

Peat, H. and Willett, P. (1991). The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42(5):378–383.

Plachouras, V. and Ounis, I. (2007). Multinomial Randomness Models for Retrieval with Document Fields. In *Proceedings of the 29th European Conference on IR Research*, pages 28–39, Berlin, Heidelberg. Springer-Verlag.

Ponte, J. and Croft, W. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 275–281, New York, NY, USA. ACM.

Porter, M. (1997). An Algorithm for Suffix Stripping. *Readings in Information Retrieval*, 14(3):313–316.

Prager, J. (2006). Open-domain question: Answering. *Foundations and Trends in Information Retrieval*, 1(2):91–231.

Qiu, Y. and Frei, H.-P. (1993). Concept based query expansion. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 160–169, New York, NY, USA. ACM.

Ravichandran, D. and Hovy, E. (2002). Learning Surface Text Patterns for a Question Answering System. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 41–47, Stroudsburg, PA, USA. Association for Computational Linguistics.

Robertson, S. (2004). Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. *Journal of Documentation*, 60(5):503–520.

Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., and Gatford, M. (1996). Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*, pages 1–18, Gaithersburg, Md., USA. Text REtrieval Conference (TREC).

Robertson, S. and Zaragoza, H. (2009). The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Robertson, S., Zaragoza, H., and Taylor, M. (2004). Simple BM25 Extension to Multiple Weighted Fields. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, pages 42–49, New York, NY, USA. ACM.

Romero, M., Moreo, A., and J.L., C. (2013). A Cloud of FAQ: A Highly-Precise FAQ Retrieval System for the Web 2.0. *Knowledge-Based Systems*, 49(0):81 – 96.

Salton, G. (1968). *Automatic Information Organization and Retrieval.* McGraw Hill Text.

Sanderson, M. (2010). Test Collection Based Evaluation of Information Retrieval Systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375.

Shaikh, A., Jain, M., Rawat, M., Shah, R., and Kumar, M. (2013). Improving accuracy of sms based faq retrieval system. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 142–156, Berlin, Heidelberg. Springer-Verlag.

Shivhre, N. (2013). SMS Based FAQ Retrieval. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 131–141, Berlin, Heidelberg. Springer-Verlag.

Silvestri, F. (2010). Mining query logs: Turning search usage data into knowledge. *Foundations and Trends in Information Retrieval*, 4(1—2):1–174.

Smeaton, A. and van Rijsbergen, C. (1983). The retrieval effects of query expansion on a feedback document retrieval system. *The Computer Journal*, 26(3):239–246.

Sneiders, E. (1999). Automated FAQ Answering: Continued Experience with Shallow Language Understanding. Question Answering Systems. In *Proceedings of the Association for the Advancement of Artificial Intelligence Fall Symposium*, pages 97–107, California, USA. AAAI Press.

Sneiders, E. (2002a). Automated Question Answering: Template-Based Approach. *Royal Institute of Technology, Sweden, PhD Thesis*, pages 1 – 264.

Sneiders, E. (2002b). Automated Question Answering Using Question Templates That Cover the Conceptual Model of the Database. In *Proceedings of the 6th International Conference on Applications of Natural Language to Information Systems-Revised Papers*, pages 235–239, Berlin, Heidelberg. Springer-Verlag.

Sneiders, E. (2009). Automated FAQ Answering with Question-specific Knowledge Representation for Web Self-service. In *Proceedings of the 2Nd Conference on Human System Interactions*, pages 295–302, Piscataway, NJ, USA. IEEE Press.

Sorokin, A. and Forsyth, D. (2008). Utility data annotation with amazon mechanical turk. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, pages 1–8.

Soubbotin, M. and Soubbotin, S. (2001). Patterns of Potential Answer Expressions as Clues to the Right Answer. In *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*, page 175182, Gaithersburg, MD, USA. National Institute of Standards and Technology (NIST).

Sparck Jones, K. and Willett, P., editors (1997). *Readings in Information Retrieval*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Stamou, S. and Efthimiadis, E. (2010). Interpreting User Inactivity on Search Results. In *Proceedings of the 32nd European Conference on Advances in Information Retrieval*, pages 100–113, Berlin, Heidelberg. Springer-Verlag.

Thuma, E., Rogers, S., and I., O. (2013). Exploiting Query Logs and Field-Based Models to Address Term Mismatch in an HIV/AIDS FAQ Retrieval System. In *Proceedings of the 18th International Conference on Applications of Natural Language to Information Systems*, volume 7934 of *Lecture Notes in Computer Science*, pages 77–89, Berlin, Heidelberg. Springer-Verlag.

Tombros, A., Villa, R., and Van Rijsbergen, C. (2002). The Effectiveness of Query-Specific Hierarchic Clustering in Information Retrieval. *Information Processing and Management*, 38(4):559–582.

Van Rijsbergen, C. (1979). Information Retrieval, 2nd Edition. *Butterworths*.

Vilario, D., Pinto, D., Len, S., Castillo, E., and Tovar, M. (2013). Two Models for the SMS-Based FAQ Retrieval Task of FIRE 2011. In *Multilingual Information Access in South Asian Languages*, volume 7536 of *Lecture Notes in Computer Science*, pages 175–183. Springer-Verlag, Berlin, Heidelberg.

Voorhees, E. (1994a). On expanding query vectors with lexically related words. In *The Second Text Retrieval Conference (TREC 2)*, pages 223–231. NIST Special Publication 500-215,.

Voorhees, E. (1994b). Query Expansion Using Lexical-semantic Relations. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 61–69, New York, NY, USA. Springer-Verlag New York, Inc.

Voorhees, E. (2001). Overview of the Question Answering Track. In *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*, page 157165, Gaithersburg, MD, USA. National Institute of Standards and Technology (NIST).

Voorhees, E. (2002). The Philosophy of Information Retrieval Evaluation. In *Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, pages 355–370, London, UK, UK. Springer-Verlag.

Vuurens, J. and de Vries, A. (2012). Obtaining high-quality relevance judgments using crowdsourcing. *IEEE Internet Computing*, 16(5):20–27.

Whitehead, S. (1995). Auto-FAQ: An Experiment in Cyberspace Leveraging. *Computer Networks ISDN Systems*, 28(1-2):137–146.

Whitla, P. (2009). Crowdsourcing and Its Applicat ion in Marketing Activities. *Contemporary Management Research*, 5(1):15–28.

Willett, P. (1988). Recent Trends in Hierarchic Document Clustering: a Critical Review. *Information Processing and Management*, 24(5):577–597.

Wu, C.-H., Yeh, J.-F., and Chen, M.-J. (2005). Domain-specific FAQ retrieval using independent aspects. *Transactions on Asian Language Information Processing (TALIP)*, 4(1):1–17.

Xu, J. and Bruce, C. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '96, pages 4–11, New York, NY, USA. ACM.

Xu, J. and Bruce, C. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18(1):79–112.

Xue, G.-R., Zeng, H.-J., Chen, Z., Yu, Y., Ma, W.-Y., Xi, W., and Fan, W. (2004). Optimizing web search using web click-through data. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, pages 118–126, New York, NY, USA. ACM.

Xue, X., Jeon, J., and Croft, W. (2008). Retrieval Models for Question and Answer Archives. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 475–482, New York, NY, USA. ACM.

Yang, S.-Y. (2006). An Ontology-supported Information Management Agent with Solution Integration and Proxy. In *Proceedings of the 10th WSEAS International Conference on Computers*, pages 1027–1032, Stevens Point, Wisconsin, USA. World Scientific and Engineering Academy and Society (WSEAS).

Yang, S.-Y. (2007a). An Ontological Multi-Agent System for Web FAQ Query. In *Proceedings of Sixth International Conference on Machine Learning and Cybernetics*, volume 5, pages 2964–2969. IEEE Xplore Digital Library.

Yang, S.-Y. (2007b). An Ontology-Supported User Modeling Technique with Query Templates for Interface Agents. In *Proceedings of the 2007 Annual Conference on International Conference on Computer Engineering and Applications*, pages 556–561, Stevens Point, Wisconsin, USA. World Scientific and Engineering Academy and Society (WSEAS).

Yang, S.-Y. (2008). An Ontological Website Models-Supported Search Agent for Web Services. *Expert Systems with Applications: An International Journal*, 35(4):2056–2073.

Yang, S.-Y., Chuang, F.-C., and Ho, C.-S. (2007). Ontology-Supported FAQ Processing and Ranking Techniques. *Journal of Intelligent Information Systems*, 28(3):233–251.

Yang, S.-Y., Hsu, C.-L., Lee, D.-L., and Deng, L. (2008). FAQ-master: An Ontological Multi-agent System for Web FAQ Services. *WSEAS Transactions on Information Science and Applications*, 5(3):221–228.

Yin, Z., Shokouhi, M., and Craswell, N. (2009). Query expansion using external evidence. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pages 362–374, Berlin, Heidelberg. Springer-Verlag.

Yom-Tov, E., Fine, S., Carmel, D., and Darlow, A. (2005). Learning to Estimate Query Difficulty: Including Applications to Missing Content Detection and Distributed Information Retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA. ACM.

Zaragoza, H., Craswell, N., Taylor, M., Saria, S., and Robertson, S. (2004). Microsoft Cambridge at TREC 13: Web and HARD tracks. In *Text REtrieval Conference (TREC-13)*.

Zhai, C. (2008). Statistical Language Models for Information Retrieval A Critical Review. *Foundations and Trends in Information Retrieval*, 2(3):137–213.

Zhai, C. and Lafferty, J. (2001). A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 334–342, New York, NY, USA. ACM.

Zhai, C. and Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22(2):179–214.

Zhan, J. and Chen, Y. (2011). Research of Chinese Word Sense Disambiguation Based on HowNet. In *Emerging Research in Artificial Intelligence and Computational Intelligence*,

volume 237 of *Communications in Computer and Information Science*, pages 477–482. Springer Berlin Heidelberg.

Zhang, D. and Lee, W. (2003). Question Classification Using Support Vector Machines. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 26–32, New York, NY, USA. ACM.

Zhang, M. and Dodgson, M. (2007). High-Tech Entrepreneurship in Asia : Innovation, Industry and Institutional Dynamics in Mobile Payments. *Edward Elgar Publishing*, pages 1–315.

Zhang, Z. and Nasraoui, O. (2006). Mining search engine query logs for query recommendation. In *Proceedings of the 15th International Conference on World Wide Web*, WWW '06, pages 1039–1040, New York, NY, USA. ACM.

Zhao, Y., Scholer, F., and Tsegay, Y. (2008). Effective Pre-retrieval Query Performance Prediction Using Similarity and Variability Evidence. In *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval*, pages 52–64, Berlin, Heidelberg. Springer-Verlag.