UNIVERSITY OF CAPE TOWN

DOCTORAL THESIS

Using Language Similarities in Retrieval for Resource Scarce Languages: A Study of Several Southern Bantu Languages

Author:
Catherine CHAVULA

Supervisors:
Prof. Hussein SULEMAN

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

School of IT
Department of Computer Science

April 18, 2021

Declaration of Authorship

I, Catherine CHAVULA, declare that this thesis titled, "Using Language Similarities in Retrieval for Resource Scarce Languages: A Study of Several Southern Bantu Languages" my own original work. Where collaborations with other researchers are involved, or materials generated by other researchers are included, the parties and/or materials are acknowledged or are explicitly referenced as appropriate.

This work is being submitted for the degree of Doctor of Philosophy in Computer Science at the University of Cape Town, South Africa. This thesis has not been submitted to any other university or institution for any other degree or examination.

Signed: Signed by candidate

Date: 18/04/2021

Publications

Early versions of some of the ideas and figures presented in this dissertation have previously appeared in the following publications:

- **Catherine Chavula** and Hussein Suleman. 2016. Assessing the Impact of Vocabulary Similarity on Multilingual Information Retrieval for Bantu Languages. In Proceedings of the 8th Annual Meeting of the Forum on Information Retrieval Evaluation (FIRE '16). ACM, New York, NY, USA, 16–23. DOI: http://dx.doi.org/10.1145/3015157.3015160
- **Catherine Chavula** and Hussein Suleman. 2017. Morphological Cluster Induction of Bantu Words Using a Weighted Similarity Measure. In Proceedings of SAICSIT '17, Thaba Nchu, South Africa, September 26–28, 2017, 9 pages. DOI: 10.1145/3129416.3129453
- Andreas von Holy, Alon Bresler, Osher Shuman, **Catherine Chavula**, and Hussein Suleman. 2017. BantuWeb: A Digital Library for Resource Scarce South African Languages. In Proceedings of SAICSIT '17, Thaba Nchu, South Africa,September 26–28, 2017, 10 pages. DOI: 10.1145/3129416.3129446
- **Catherine Chavula** and Hussein Suleman. 2020. Intercomprehension in Retrieval: User Perspectives on Six Related Scarce Resources Languages. In Proceedings of 2020 Conference on Human Information Interaction and Retrieval, Vancouver, BC, Canada, March 14–18, 2020 (CHIIR '20), 10 pages. https://doi.org/10.1145/3343413.3377954

Abstract

Most of the Web is published in languages that are not accessible to many potential users who are only able to read and understand their local languages. Many of these local languages are Resources Scarce Languages (RSLs) and lack the necessary resources, such as machine translation tools, to make available content more accessible. State of the art preprocessing tools and retrieval methods are tailored for Web dominant languages and, accordingly, documents written in RSLs are lowly ranked and difficult to access in search results, resulting in a struggling and frustrating search experience for speakers of RSLs. In this thesis, we propose the use of language similarities to match, re-rank and return search results written in closely related languages to improve the quality of search results and user experience. We also explore the use of shared morphological features to build multilingual stemming tools.

Focusing on six Bantu languages spoken in Southeastern Africa, we first explore how users would interact with search results written in related languages. We conduct a user study, examining the usefulness and user preferences for ranking search results with different levels of intelligibility, and the types of emotions users experience when interacting with such results. Our results show that users can complete tasks using related language search results but, as intelligibility decreases, more users struggle to complete search tasks and, consequently, experience negative emotions. Concerning ranking, we find that users prefer that relevant documents be ranked higher, and that intelligibility be used as a secondary criterion. Additionally, we use a User-Centered Design (UCD) approach to investigate enhanced interface features that could assist users to effectively interact with such search results. Usability evaluation of our designed interface scored 86% using the System Usability Scale (SUS). We then investigate whether ranking models that integrate relevance and intelligibility features would improve retrieval effectiveness. We develop these features by drawing from traditional Information Retrieval (IR) models and linguistics studies, and employ Learning To Rank (LTR) and unsupervised methods. Our evaluation shows that models that use both relevance and intelligibility feature(s) have better performance when compared to models that use relevance features only. Finally, we propose and evaluate morphological processing approaches that include multilingual stemming, using rules derived from common morphological features across Bantu family of languages. Our evaluation of the proposed stemming approach shows that its performance is competitive on queries that use general terms.

Overall, the thesis provides evidence that considering and matching search results written in closely related languages, as well as ranking and presenting them appropriately, improves the quality of retrieval and user experience for speakers of RSLs.

Acknowledgements

I would like to thank my supervisor, Professor Hussein Suleman for his support, guidance and mentorship throughout my studies. Thank you for your patience and all the encouragement.

I have also enjoyed the encouragement and support of my lab mates in UCT's ICT4D Centre and DL Research Group. Thank you all for the discussions and your feedback.

I also had the opportunity to work with other students on two projects. I worked with three honours degree students, Andreas von Holy, Alon Bresler and Osher Shuman on Bantu Web project. I also had two honours degree students working with me on this project; Sinead Urisohn and Andre Lopes who worked on interface design and re-ranking of documents based on language similarity of South African Bantu languages. I am grateful for the experience and lessons learnt from these projects.

I am thankful for the text provided to me by Zodiak Broadcasting Corporation (ZBS) and Fuko Newspapers. This research work would not have been possible without their support. Special thanks to Alfred Mtaula and Bright Kumwenda of Fuko Newspaper, and Teresa Chirwa of Zodiak.

Many thanks to all the people who participated in the numerous tasks that were part of my research work. I would like to thank students from Mzuzu University, Malawi Polytechic, Chancellor College and University of Cape Town who participated in user studies and test collection creation tasks.

Finally, I would like to thank my family and friends for their support. I thank my husband Josiah for companionship and support. Thank you Natasha and Nathaniel, for always reminding me that I needed to finish my doctoral studies. Special thanks to Jabulani, my PhD baby for all the surprises you brought. I thank my parents, brother and sisters, and my in-laws for their support. I thank the many friends at Mowbray Presbyterian Church for their support and encouragement throughout my studies.

Contents

List of Figures

List of Tables

List of Abbreviations

CLEF	**C**onferences and **L**abs of the **E**valuation **F**orum
CLIR	**C**ross **L**anguage **I**nformation **R**etrieval
IR	**I**nformation **R**etrieval
LTR	**L**earning **T**o **R**ank
MLIR	**M**u**L**tilingual **I**nformation **R**etrieval
RF	**R**andom **F**orest
TREC	**T**ext **Re**trieval **C**onference
SVM	**S**upport **V**ector **M**achines
UCD	**U**ser **C**entered **D**esign
NDCG	**N**ormalised **D**iscounted **C**umulative **G**ain
SUS	**S**ystem **U**sability **S**cale
RSLs	**R**esource **S**carce **L**anguage**s**
UGC	**U**ser **G**enerated **C**ontent
NCS	**N**oun **C**lass **S**ystem
TF	**T**erm **F**requency
IDF	**I**nverse **D**ocument **F**requency
BM25	**B**est **M**atch **25**
MLE	**M**aximum **L**ikelihood **E**stimate
DFR	**D**ivergence **F**rom **R**andomness
MAP	**M**ean **A**verage **P**recision
ICC	**I**ntraclass **C**orrelation **C**oefficients
ANOVA	**A**nalysis **O**f **VA**riance

Dedicated to my father and mother

Chapter 1

Introduction

The current digital revolution has changed how people seek and use information. The unprecedented large volume of information available on the Web is open for access to everyone at almost no cost. Accordingly, the Web has become the primary source of information in the 21st century. The success of the Web has been driven by many factors, and key among them is its decentralised architecture, whereby anyone can publish and consume information. Anyone can become a Web publisher through User Generated Content (UGC), using various free services and systems, such as social media. Although UGC services have increased the diversity of content on the Web, and in spite of the Web becoming more multilingual, very little content has been published in many languages of the world. English and other widely spoken languages continue to dominate the Web. Consequently, the majority of content currently available on the Web does not represent the cultural and linguistic diversity of the world. Moreover, Information Retrieval (IR) systems and techniques have been optimised for languages with the most content on the Web. Users who are interested to read content written in Resource Scarce Languages (RSLs) tend to have difficulties in finding relevant information. Struggling to find information may lead to frustration and unsatisfactory search experience. Therefore, an important challenge is to make such small amounts of content more accessible to users.

1.1 Context and Motivation

Bantu languages are spoken by a majority of the people in Sub-Saharan Africa (Nurse and Philippson, 2006). The prevalent use of technology requires that information access systems support multiple languages, including less dominant languages such as Bantu languages, to reflect the linguistic diversity of the users. Multilingual Information Retrieval (MLIR) systems support users to search for content written in multiple languages using a single query. MLIR uses language resources and tools such as bilingual dictionaries and machine translation systems to bridge the gap between the language of the query and the languages of the documents. However, most Bantu languages lack such resources and tools. To illustrate the extent of difference in Web content, Figure 1.1 shows the number of Wikipedia articles

for some of the Bantu languages. In contrast, there are over six million Wikipedia articles written in English[1]. As a result, speakers of these Bantu languages struggle to access digital information written in languages familiar to them.

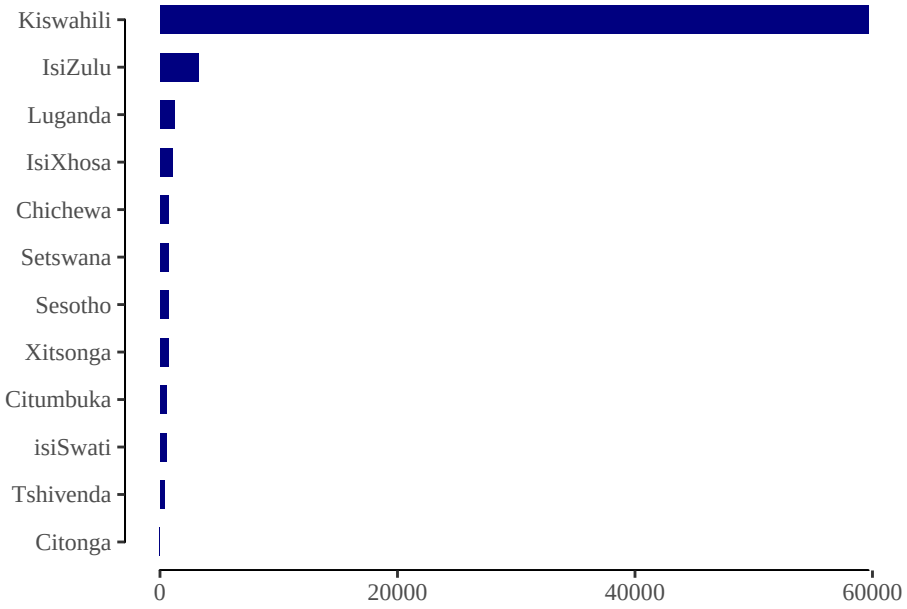


FIGURE 1.1: Number of Wikipedia articles by language

Bantu languages have agglutinative morphological structure: a single word may contain several syntactical elements that could form a sentence in other languages, e.g. *'We will see them.'* in Chichewa as a single word is *'ti-dza-wa-wona'*, with literal translation as *'we-will-them-see'*. With regards to retrieval, such word properties can negatively affect recall if there is no morphological analysis – the many morphological variants that may be present in a corpus may not be matched to a single token in retrieval. However, developing tools for every single Resource Scarce Language from scratch is time consuming and expensive. Fortunately, Bantu languages share similarities with respect to sound, grammar, meaning, and the lexicon. These similarities provide opportunities to improve MLIR of Bantu languages through the use of language similarities in morphology, syntax, and vocabulary to develop resources, tools, and IR models. For example, language relatedness features, such as lexical similarity, provide cross-lingual links that can be exploited in MLIR in cases where under-resourced languages are used.

1.2 Problem Specification

Searching the Web using RSLs such as most of the Bantu languages is frustrating due to several reasons that include: i) RSLs have limited number of available relevant documents

[1]Information accessed on 23^{rd} July, 2020 at https://en.wikipedia.org/wiki/List_of_Wikipedias#Detailed_list

and, therefore, either no relevant content is returned for many queries or no search results are returned at all; ii) search engines often rank relevant documents written in RSLs lowly because either queries in RSLs are rare and lack previous examples to learn from (Golebiewski and Boyd, 2018) or current retrieval techniques are tailored to favour dominant languages (Mustafa and Suleman, 2011); iii) and pre-processing tools and retrieval techniques are usually tailored for Web dominant languages and when these techniques are applied on RSLs queries and documents, the quality of the search results is negatively affected (Golebiewski and Boyd, 2018); and, search engines may return results in other languages based on matches of some of the query terms, which the user may not be able to understand (Chavula and Suleman, 2016).

1.3 Research Approach

The principal focus of this thesis is in dealing with limited language resources and tools for RSLs. Our goal is to develop techniques that can be used in systems that provide access to information for RSLs speakers in regions where there is high similarity of languages. In particular, we investigate whether taking into account language similarities and relevance can improve quality of results and user experience. We aim to give users who are interested in searching in local languages more relevant content by incorporating relevant results written in closely related languages.

Using this approach, users will have more relevant documents that have different levels of comprehensibility, i.e., a document is deemed to be useful only if the user is able to understand the presented content and if the presented content meets his or her information need. Our assumptions are that users will be able to understand the search results either due to mutual intelligibility or multilingualism. The idea pursued in this thesis is the use of language similarities to read and understand search results written in related languages. As an example, we imagine a user with the following information needs and context as below (see Figure 1.2):

> Kondwani is a small restaurant owner at a rural trading centre. He can read and understand content written in Citumbuka, which has no translation tools and has limited digital resources. He would like to find information about new recipes for cooking chicken and rice for his customers. He goes through four steps to find a document that he could use (steps also illustrated in Figure 1.2). (1) He formulates a query in Citumbuka *'kuphika nkhuku na mpunga'* or *'cooking chicken and rice'* in English. He opens a Web browser on his phone and enters his query into the search engine to find information he needs. The search engine returns irrelevant search results, and suggests an alternative query in Kiswahili. (2) He then reformulates the query and removes the rice component

of the query. The search engine provides him with relevant search results in Kiswahili. (3) He decides to reformulate the query into *'kuphika nyama'* or *'cooking meat'* in English, but still gets irrelevant results, including a page on Ebola as the first result. (4) He then reformulates the query yet again, and gets a relevant result in a related language.

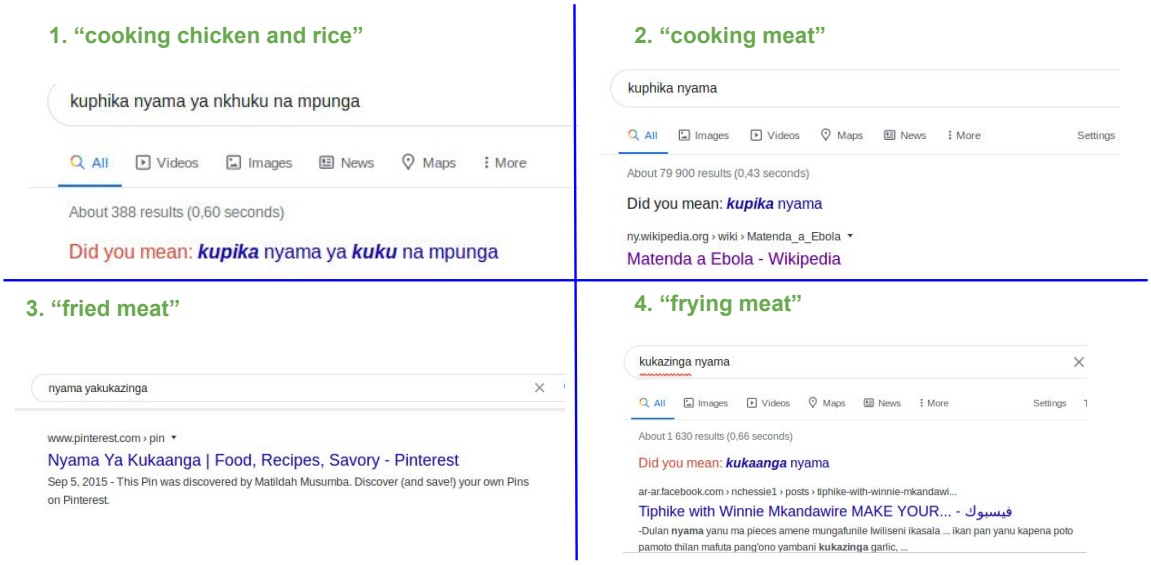


FIGURE 1.2: Steps for the example information need

The user search scenario depicted in Figure 1.2 provides an example of user information and search context of speakers of languages with limited resources. Most of their search tasks involve struggling and usually end with no success. In the above scenario, the user is provided with results in Chichewa and Kiswahili for his search query, and the assumption is that these results are relevant to his information need. The basic idea is that the user will be able to understand the documents based on his assumed prior language knowledge. The more similar his language is to the search result language, the more comprehensible the presented content will be. This form of communication is called receptive multilingualism, i.e., using one's language to communicate with others who speak another language (Peter, 2005).

Receptive multilingualism is a form of intercultural communication in which people use different languages to interact and communicate without resorting to a common language (Peter, 2005). Receptive multilingualism is either inherent or acquired. Inherent receptive multilingualism is based on shared language features and occurs only between speakers of closely related languages. Acquired receptive multilingualism is subjective and is dependent on the language skills and personality traits of an individual. The term intercomprehension is commonly used interchangeably with receptive multilingualism but is usually preferred in the context of text readability or intelligibility in receptive multilingualism

(Stenger et al., 2017). Harnessing intercomprehension in retrieval can offer users of under-resourced languages access to more content than readily available in the user's native language. Intercomprehension is applicable to closely related Bantu languages because most of these languages share similar features such as morphological, phonetic, syntactical, lexical and orthographic attributes as they are mostly written using the Latin alphabet (Nurse and Philippson, 2006).

In this thesis, intercomprehension is proposed to be used in IR for closely related Bantu languages queries and documents, with the assumption that users submitting queries in a particular language can understand the content in other closely related languages with some difficulty or effort, referred to here as *intercomprehension cost*, i.e, the effort required to understand text written in an unfamiliar language. This cost is dependent on how intelligible the language of the query is with respect to languages of the retrieved search results. Given a retrieval system that expands search to related languages, search terms would be matched with documents in closely related languages as well as documents in the language of the query, and the search result lists would consist of documents in several languages with varying levels of relevance and language similarity or intelligibility. To this end, one of the questions asked in this thesis is how to present such search results with varying levels of relevance and intelligibility to the user. Accordingly, the problem is a search result re-ranking problem or result aggregation problem, i.e., given a set of documents retrieved based on relevance and language similarity, how should the results be rearranged to maximise the gains in terms of relevance while not degrading comprehensibility or increasing intercomprehension cost.

Our thesis is that by considering and matching search results written in closely related languages, ranking the search results using relevance and intelligibility features, and presenting them appropriately, we can improve retrieval quality and user experience for RSLs speakers. Thus, our work investigates both system and user aspects of retrieval, focusing on ranking, search results presentation, user interaction and text pre-proccessing. Figure 1.3 shows the different aspects of search our work focused on. In our system oriented component we develop a test collection and perform off-line evaluation on learning to rank models using relevance and intelligibility features, and text pre-proccessing techniques. User oriented aspects of our work investigated search results presentation from the perspectives of interface design, user document preference, and affective response to search results requiring intercomprehension.

1.4 Research Questions

This section starts with research questions on understanding user interaction with search results written in related languages. We then provide research questions for the ranking

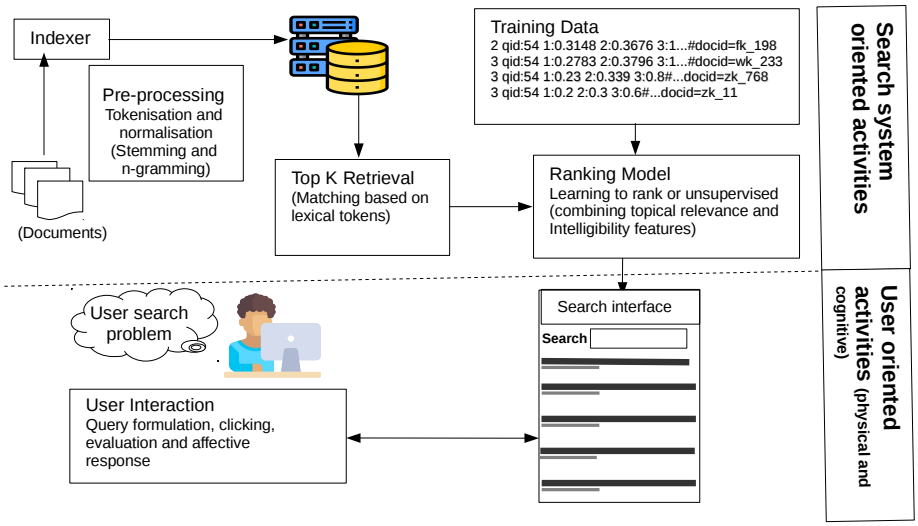


FIGURE 1.3: The search process in the context of the work thesis the thesis focused on.

problem. Finally, we provide a description of the problem in the context of morphology, and proceed with a specification of the morphology research question pursued in this thesis.

1.4.1 User Search Experience

User interaction with a retrieval system that uses intelligibility in its ranking model may produce new search behaviours, hence the need to understand the interactions and design a user search experience that takes into account the different behaviours that may manifest. Moreover, users may require new interface features that can help them use such an IR system effectively. To this end, the following sub-questions are investigated:

RQ1 Are search results written in closely related languages useful to the user?

RQ2 What are the ranking preferences of users for search results written in related languages with varying intelligibility? Does intelligibility matter in the rank preference of such results?

RQ3 What types of emotions do users experience when interacting with search results that require intercomprehension?

RQ4 What is the appropriate search results presentation style for related languages' search results.

We investigate user behaviour and preferences in retrieval involving related languages in Chapter 5.

1.4.2 Re-Ranking of Search Results

Rank aggregation based on relevance and intelligibility features requires combining search results from multiple ranked lists originating from related languages into a single ranked list. In this approach, a query submitted by the user is matched with documents in several related language collection indexes; each system returns a ranked list of results, which are then combined to form a single ranked list. Aggregation is performed by considering the probability of relevancy of the document with respect to the query, as well as intelligibility in relation to languages of the documents and original query. Given that the ranking for each list is corpus dependent, the rankings and the similarity scores from each list are not comparable. Hence, there is a need to normalise similarity scores from different engines or languages to make them homogeneous. Nevertheless, the primary problem is how to effectively rank documents in different languages in such a way that intercomprehension costs are minimised while improving topical relevance gains.

Our approach, described in Section 6.2, is to re-rank documents by integrating topical relevance and intelligibility features to find the best scoring function for ranking search results written in closely related languages. Learning To Rank (LTR) is proposed, whereby a ranking function is trained to rank documents as preferred by a monolingual user looking at documents in related languages. Naturally, LTR is the best approach to solve this problem since supervised learning is known to automatically tune parameters and combine multiple evidences (Li, 2011). Hence, the ranking function should be able to learn from user ranking preferences of multilingual search results of closely related languages to rank documents effectively. Ultimately, our research answers the following research question:

RQ5 Does re-ranking of search results based on relevance and intelligibility improve retrieval effectiveness?

We investigate and evaluate models that integrate relevance and intelligibility features in Chapter 6.

1.4.3 Morphological Analysis

Bantu languages are morphologically agglutinating and have a rich morphology. The Noun Class System (NCS) and the concord agreement system contribute to the complexity of Bantu words - every noun belongs to a class that prescribes a prefix that the noun can get and the head noun demands agreeing elements on each word in a construction (e.g, *chi-patso cho-kupsa chi-makoma* – a ripe fruit is delicious. The *chi-* in *chi*-patso is a prefix provided by a noun class, and all the other elements in the sentence have a prefix *ch-* to agree with the head noun prefix.). In addition, compounding and reduplication make Bantu word structure unique, e.g., to say that something happens frequently, the root of the word may be repeated (cooking frequently – *-phikaphika*).

Further, verbal inflection includes elements that are expressed lexically or syntactically in other languages, e.g., a Bantu word can stand on its own as a complete sentence (Nurse, 2008). A single Bantu word can thus express something that can only be verbalised using multiple words in other languages such as English.
For example, consider the following examples given by Kosch (2006):

1. `Citumbuka: nikubabona`
2. `isiZulu: ngiyababona`
3. `Chichewa: ndikuwawona`
4. `English: I see them`

The Bantu constructions (1, 2 and 3) contain only one word for what is expressed in English using three words. The components that have been expressed lexically in English, i.e., *I* and *them* have been conjugated to the verbal root *bon-*. The example above shows how the three languages – Citumbuka, isiZulu and Chichewa – share some structural and lexical features, i.e., *bona* is common in two languages (but pronounced as *wona* and is written using a (*w*)ona in Chichewa), while the orthographic conventions may be different. For example, *bona* in Citumbuka and *wona* have the same pronunciation. Also, the Bantu languages in the illustration use prefixes to indicate some grammatical contexts, i.e., adding all the affixes to the root. This thesis investigates the impact of morphological analysis on the quality of retrieval in MLIR contexts for languages in the family of group N Bantu languages, particularly focusing on methods that can use morphological similarities, and aims to answer the following question:

RQ6 How do multilingual stemmers that use structural similarities of Chichewa and Citumbuka compare in terms of retrieval effectiveness with no stemming, language independent stemming using character tri-grams, and language specific stemming?

We investigate the issue of morphology and the evaluation of our proposed stemming approach in Chapter 7.

1.5 Research Significance

The findings of this research work contribute towards improving access to information for under-resourced languages, and will provide insights to MLIR for related languages in general. Our thesis makes the following contributions:

- We provide insights to user interaction behaviour towards results written in related languages.

- We show how a language-similarity based ranking model improves retrieval effectiveness for related languages.
- We design an appropriate search system user interface for presenting results of related languages.
- We design morphological analysis techniques and tools that use similarities of Bantu languages to improve retrieval for related languages.

1.6 Structure of Thesis

The rest of the thesis is organised as follows:

- Chapter 2 Background: The chapter introduces the major concepts used in the thesis, with focus given to IR and linguistics – more specifically intelligibility and morphology. We first introduce IR concepts, including retrieval models and retrieval evaluation. Further, we introduce the concept of intelligibility in the manner explored in our research. Finally, we introduce Bantu languages and their general morphological features. Particular emphasis is given to languages spoken in Zone N of the classification of Bantu Languages, i.e., Citumbuka and Chichewa.
- Chapter 3 Literature Review: The chapter presents an analysis of relevant literature pertaining to the thesis. Firstly, we discuss work on retrieval involving related languages. Secondly, we discuss approaches for merging of results involving distributed architectures in different scenarios, such as multilingual retrieval, federated search, meta-search and data fusion. Thirdly, we discuss LTR methods used to learn document ranking functions, including for relevance, diversity and recency. We then discuss user perspectives in retrieval focusing on multilingual retrieval. Finally, we discuss related work on morphological analysis in retrieval and, in particular, we discuss methods for stemming.
- Chapter 4 Test Collection Development: In this chapter, we discuss the procedure used to develop a test collection used in system-centered evaluation involving stemming techniques and ranking of search results using LTR. We also discuss the features, procedures and steps undertaken to obtain relevance and intelligibility features used in our experimentation.
- Chapter 5 User Perspectives: The chapter provides the research design, methods and procedures used for our user-centered approach to investigate retrieval involving related languages, including the experimental results and their discussions. We first provide our methodology for a user study to understand the interaction behaviour

and perspectives for users interacting with search results written in related languages. We then report on the design of search interfaces and features to assist users to navigate search results written in related languages. Finally, we discuss our results and provide our insights on users interacting with such search results.

- Chapter 6 Ranking for Relevance and Similarity: In this chapter, we investigate re-ranking of search results written in related languages using relevance and intelligibility features. We start by providing an analysis of intelligibility features through feature selection and prediction. We then present the approach used in re-ranking. This is followed by a discussion of our results on intelligibility prediction and feature selection, and re-ranking of search results.

- Chapter 7 Multilingual Stemming: The chapter presents an approach for multilingual stemming of text written in Citumbuka and Chichewa. We start with a presentation of the proposed approach for stemming. This is followed by the experimental set up. We then provide results of the evaluation of stemming. We end with a discussion of results for the stemming approach.

- Chapter 8 Conclusion: The chapter provides a summary of our findings and conclusions. We also highlight our key contributions, and discuss how the work presented in this thesis can be extended in the future.

Chapter 2

Background

This chapter provides the relevant background for the different topics in IR and linguistics related to the research work presented in this thesis. Firstly, we introduce the basic concepts and components of IR, including retrieval models used in our experiments, and the nature of experimentation in IR and its evaluation. Secondly, we present a summary description of Bantu languages, including the major topics relevant to this thesis; we provide a brief introduction on intelligibility for the studied languages, and a description of morphology for Bantu words, focusing on morphological processes for Southeastern Bantu languages.

2.1 Information Retrieval Fundamentals

The field of IR is devoted to the representation, storage, organization of, and access to information objects (Manning, Raghavan, and Schütze, 2008). IR systems such as Web search engines allow users to find digital content that meets their information needs, from a collection of documents. The document containing information that meets their information need is known as a relevant document. This information may be in different forms such as text, video, graphics or audio documents. Our research focuses on retrieval of textual documents.

In this section, we present basic concepts of IR that are relevant to our thesis. We first provide a brief description of the basic components of a retrieval system including, preprocessing step, and indexing and retrieval in Section 2.1.1. This is followed by a discussion of the different views of relevance in Section 2.1.2. We then discuss traditional retrieval models used in search systems in Section 2.1.3. The section ends with a discussion on the evaluation of IR systems in Section 2.1.4.

2.1.1 Retrieval Process

In a typical text retrieval system, users search for textual documents by submitting a query based on their information need (Manning, Raghavan, and Schütze, 2008). Thereafter, the system searches its collection for documents that match the query and, if there are matches, the users are presented with a ranked list of documents. The process of transforming a

user's information need into a representation that an information retrieval system can use, i.e., query, is known as query formulation. Although the user may have an idea of a prototype document that meets their information need, usually the formulated query may be underspecified due to the user being unaware of what vocabulary is used in a relevant document. In this context, the retrieval system matches the query with documents that have some degree of likelihood for relevance. This inherent unpredictability makes retrieval a challenging task, i.e., retrieval problem. This has motivated the formulation of ideas such as Probability Ranking Principle (PRP) (Robertson, 1977). PRP asserts that documents should be ranked in an order of decreasing probability of relevance. The retrieval problem is exacerbated by retrieval across languages (Peters, Braschler, and Clough, 2012).

Retrieval systems consist of a pipeline of activities involving the retrieval system itself, as well as tasks involving users interacting with the retrieval system. A retrieval system needs to have documents ready for access: documents to be searched are first gathered – Web crawled or from some digital collection; and the text from documents is pre-processed and indexed. When a user has an information need, they formulate queries that specify their information need and submit them to the retrieval system. The retrieval system pre-processes each query before query terms are matched with document terms. Matching documents are ranked based on some scoring function and are presented to the user for relevance feedback. Generally, IR systems are evaluated using labelled relevance judgement data or user implicit search behaviour such as query logs. We used user explicit relevance judgement labels and user ranking preferences in our evaluation. Figure 2.1 shows the pipeline of activities in the retrieval process in IR systems (Manning, Raghavan, and Schütze, 2008).

Preprocessing

Documents are transformed to a data structure that makes document and query matching easier and faster through indexing. Before a document is indexed, its text is preprocessed to find the forms of terms to be indexed to optimise retrieval performance. Terms are words from the document text or submitted queries. Preprocessing includes such tasks as tokenisation, stopping and normalisation. We provide a brief description of each of these tasks in the remainder of the section.

Tokenisation: Tokenisation is the process of splitting text into sequences of characters (splitting text into what is typically words). For languages that use the Latin Alphabet, such as Bantu languages, tokenisation is simple – the general approach is to split at non-letter characters, such as white spaces. Tokens are processed further in subsequent preprocessing steps before they are indexed.

Stopping: Stopping is the process of removing stopwords – the most common words in a collection (Manning, Raghavan, and Schütze, 2008). For example, functional words such as articles */a,the/* and prepositions */at, with/* are frequently used in human utterances and

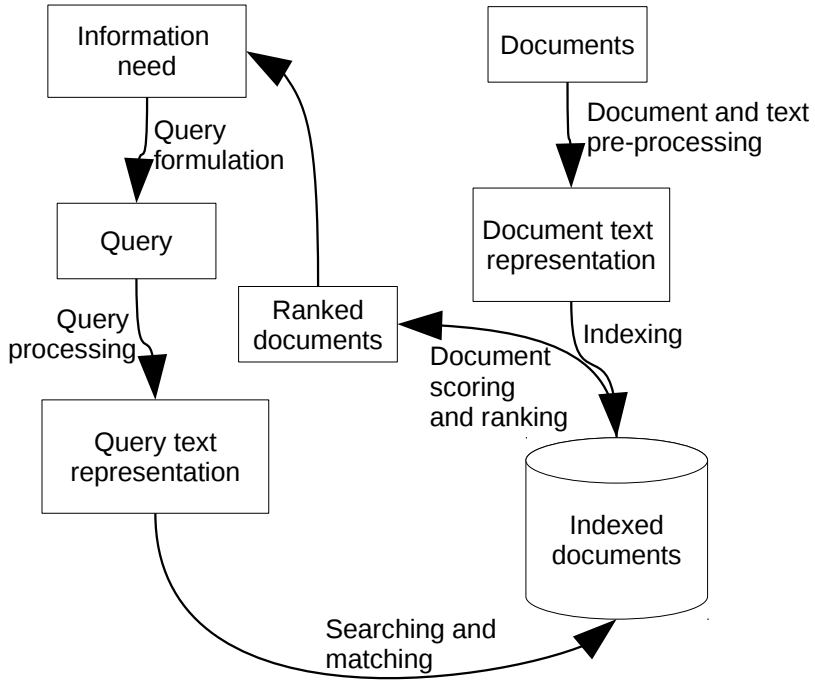


FIGURE 2.1: The Retrieval Process: Major tasks in IR systems (Manning, Raghavan, and Schütze, 2008)

do not usually carry significant content necessary in retrieval. The common approach is to remove words, from queries or text, that appear in a list of frequent words that are not content words of a language. Some words or sequences of words may be reserved to be indexed because of having a special usage in a language.

Normalisation: Normalisation is performed to map similar words of different surface forms to one word (Manning, Raghavan, and Schütze, 2008). The most common tasks in normalisation are case folding and stemming. Case folding changes the case of terms to a common case, i.e., changing all uppercase letters to lowercase. This may be a challenge for special terms such as abbreviations. A typical approach is to have a reserved set of words that may not be normalised for both the query terms and document terms.

Stemming: The process of stemming reduces morphological variants into a common stem, i.e., removing affixes in words – [play, played, plays, playing] $\rightarrow$ play (Manning, Raghavan, and Schütze, 2008). Stemming is largely language dependent because the stripping of affixes depends on the morphological rules of the language. Several approaches have been proposed for stemming, namely: using algorithms in rule based stemmers (Porter, 1980); using large dictionaries that have entries for morphological forms and their stems (Krovetz and Croft, 1992); and corpus with statistical methods (Majumder et al., 2007; Mayfield and McNamee, 2003). Stemming improves retrieval effectiveness as it enables more documents related to the term in the query to be retrieved. The terms generated from documents are added to the index. The matching of terms from the documents and the queries

are done based on terms produced from the normalisation process. We investigate the use of statistical and language family rules for Chichewa and Citumbuka in Chapter 7.

Indexing and Retrieval

Indexing is the process that adds transformed document terms to a data structure, such as an inverted index, for efficient retrieval (Manning, Raghavan, and Schütze, 2008). An inverted index contains each distinct term in the corpus of the document collection or vocabulary of the corpus, and the list of documents where the terms appears. The index may contain additional information, such as positional information of a term in each document and the frequency of each word in each document. In retrieval, query terms and index terms are matched to find relevant documents. Matching documents are ranked based on a scoring function.

2.1.2 Notions of Relevance

Relevance is the central concept in IR – search systems need to predict relevance of documents to meet diverse user information needs. However, it has been argued that relevance is multidimensional, with perspectives such as topicality, utility, scope, reliability, novelty and understandability (Saracevic, 1975; Mizzaro, 1997; Cosijn and Ingwersen, 2000; Xu and Chen, 2006; Mao et al., 2016). Accordingly, these different forms of relevance have been classified into two categories (Saracevic, 2007), namely: user oriented and system oriented relevance. System oriented relevance is concerned with the topicality of documents – how well the topic of the query matches the topic of the document. In this regard, the relevance of the document is objective and remains static – relevance does not change. User oriented relevance is based on a user's interpretation of relevance, which is subjective and depends on such factors as the user's state of knowledge, experience, perceptions, preferences, and search context. For example, users speaking different varieties of a language may have different language preferences, which may affect their relevance judgement of the same documents. Consequentially, different retrieval strategies or models have been proposed to meet these complex user information needs. These retrieval models assume different conceptual views of relevance.

2.1.3 Retrieval Models

An information retrieval model describes a computational model for representing documents and queries as well as how the two representations are matched (Manning, Raghavan, and Schütze, 2008). The representations are used to compute a measure of similarity between a query and a document. Similarity measures are usually based on how frequently

the terms in a query have been used in a document, i.e., assuming topicality sense of relevance. Information retrieval systems use retrieval models to match and rank documents. The section proceeds with an overview of retrieval models as follows. We first provide a description of the Boolean and vector space retrieval models. This is followed by summary descriptions of models used in our dataset development and stemming experiments, namely: BM25, Divergence for Randomness and Language models. Finally, we provide a description of the Learning to Rank approach, which is a major approach used in our ranking experiments in Chapter 6.

Boolean Retrieval: The Boolean retrieval model is an exact-match model where queries are Boolean expressions (Manning, Raghavan, and Schütze, 2008) – a combination of keywords and Boolean operators or connectives: *not, and, or*. Documents are matched with queries based on a Boolean operation between the query and the document. A Boolean model predicts that a document is relevant or not based on the outcome of the evaluation of the Boolean expression and document terms in the index. Traditional Boolean IR systems are referred to as exact matching models because search results are made up of documents that perfectly matched the terms in the query. The major limitation of this Boolean model is that no measure is used to determine the degree of relevance for the documents that matched with the query. Moreover, users are not specifically trained to write Boolean expression queries. Boolean models are widely used in fields with skilled users with precise information needs such as librarians and legal experts.

Vector Space Model: Ranked retrieval models estimate relevance of documents relative to a submitted query, through a relevance similarity score, i.e., the degree of similarity between the query and document based on a ranking algorithm (Salton and McGill, 1986). In Vector Space Model (VSM), documents and queries are represented as vectors; terms in a query are used to construct a query terms vector and, similarly, terms in documents are used to construct document vectors. A similarity measure such as Cosine Similarity is computed to estimate the similarity between the document vector and query vector (Salton and McGill, 1986). In this vector space, each term from a document and query is a dimension. VSM does not specify what weighting mechanism to use, e.g., binary numbers may be used to represent the presence or absence of a term in a document but in practice this does not help in returning a ranked list. In practice, Term Frequency–Inverse Document Frequency (TF-IDF) and its variants are used as weighting. TF–IDF is a weighting factor that indicates the significance of a term in a document in relation to a document collection. Term Frequency (TF) is the number of times a term appears in a particular document. Document Frequency (DF) is the number of documents in which a particular term occurs. Simple ranking models may use the sum of TF-IDF values of the query terms to compute the similarity between the query and the documents to rank documents. In generic vector space models, a document

and query are represented as t-dimensional vectors using TF-IDF weights and a cosine angle is used to determine the degree of similarity between the document and query. Cosine similarity is calculated by finding the dot product between the vector of the document and query. If d and q be two vectors of the query and document, the cosine similarity of the two vectors can be calculated as:

$$sim(q,d) = \frac{\sum_{i=1}^{n} q_i d_i}{\sum_{i=1}^{n} q_i^2 \sum_{i=1}^{n} d_i^2} \tag{2.1}$$

VSM does not predict whether documents are relevant or not but presents the user with search results that are ranked in descending order according to their degree of similarity to the query. We used Cosine similarity as an intelligibility feature calculated between equivalent documents of language pairs as well as a training feature for document and query pairs in Chapter 6.

BM25: BM25 is a probabilistic model based on the assumption that systems do not have explicit information about relevance status and, hence, PRP should be used (Robertson and Walker, 1994). Probabilistic models use probabilities based on the likelihood that a document is relevant as a scoring function (Manning, Raghavan, and Schütze, 2008). The BM25 weighting scheme, often called Okapi weighting, after the system in which it was first implemented, was developed as a way of building a probabilistic model sensitive to term frequency and document length while not introducing too many additional parameters into the model. BM25 is used as a feature for representing the matching degree of relevance between query terms and document terms (Robertson and Walker, 1994). BM25 of query q and document d is calculated as follows (Robertson and Zaragoza, 2009):

$$BM25(q,d) = \sum_{w \in q \cap d} idf(w) \frac{(k+1)tf(w)}{tf(w) + k((1-b) + b\frac{dl}{avgdl})}, \tag{2.2}$$

where w denotes a word in d and q , $tf(w)$ denotes the frequency of w in d , $idf(w)$ denotes the inverse document frequency of w and dl denotes the length of d and $avgdl$ denotes the average document length. k and b are parameters that can be optimised on specific collections. We used BM25 in all our system oriented tasks including ranking documents for relevance judgements in Chapter 4 and using BM25 scores as training features in Chapter 6.

Language Models for Retrieval: Language models learn the probability of word occurrence and are used to estimate the probability distribution of words. A language model may be based on single words or unigrams, and the model estimates the probability distribution over individual words. N-gram language models estimate probabilities of sequences of n words. The most commonly used approach is to rank documents based on the likelihood that the model which produced the document could also generate the query, i.e., query likelihood model (Ponte and Croft, 1998). The underlying principle in this approach is that

users have an idea of what words should appear in a relevant document and that queries are formulated to distinguish what documents are relevant and not relevant (Ponte and Croft, 1998). The more likely a query would be randomly generated from a document, the higher the likelihood the document is relevant. Maximum Likelihood Estimate (MLE) is used to estimate $p(q|m_d)$, the probability of query q to be generated by language model of document d as follows:

$$p(q|m_d) = \prod_{t \in q} \frac{t_f(t,d)}{|d|} \tag{2.3}$$

where m_d is the language model of the document d, t_f is the frequency of term t in d, and $|d|$ is the total number of tokens in document d. Terms missing in a document would have an estimated probability as zero, i.e, $p(t|m_d) = 0$. For conjuctive semantics, it means that $p(q|m_d) = 0$, for any missing terms. Smoothing is used to avoid assigning zero probability to a document with no term matching in the document by assigning non-zero values to such terms . The most simple approach to smoothing is Raplace smoothing, adding 1 to every count and normalising the probabilities by dividing by the sum of the number of tokens in the document and number of terms in a query. The probability of having a query term given a language model of a document, p $(t|m_d)$, is estimated as follows:

$$p(t|m_d) = \frac{t_f(t,d) + 1}{|d| + k} \tag{2.4}$$

where t_f is the frequency of term t in d, |d| is the total number of tokens in document d, k is the number of terms in query $Q = t_1, t_2, ..., t_k$. Another common approach is to assign probabilities for non-occurring terms based on a reference probabilistic distribution, the language model of the collection, i.e., if t_f = 0, then :

$$p(t|m_d) \leqslant \frac{Ct_f}{T} \tag{2.5}$$

where Ct_f is the frequency of term t in the whole collection, T is the total number of tokens in the collection. One way of using this is to use a language model built from the whole collection as Bayesian Smoothing using Dirichlet priors : $(p(t_1|m_d), p(t_2|m_d), ..., p(t_n|m_d))$, model is given by:

$$p(t|m_d) = \frac{t_f(t,d) + \mu p(t|M_c)}{|d| + \mu} \tag{2.6}$$

where M_c is a language model built from the whole document collection. |d| is the total count of terms in document d. μ is a constant, a smoothing parameter that can be adjusted. Therefore, the document ranking score is given by:

$$logp(q|d) = \sum_{t_i \in q} log \frac{t_f(t_i d) + \mu p(t|M_c)}{|d| + \mu} \tag{2.7}$$

The amount of smoothing is controlled by μ, which can be tuned to optimise performance. Smoothing in LMs for IR has generated good retrieval effectiveness and is known to have the same effect of document length normalisation of IDF in classic retrieval models such as VSM (Zhai and Lafferty, 2001). We used LM with Jeliniker-Mercer smoothing for calculating our relevance features for learning to rank and retrieving documents for relevance assessments tasks (see Chapter 4).

Divergence from Randomness: Divergence from Randomness (DFR) is a framework that consists of probabilistic methods for weighting terms using the concept of eliteness (Amati and Van Rijsbergen, 2002). Documents are ranked based on term weights computed by measuring the divergence between a term distribution produced by a random process (within the collection) and the actual term distribution (within the document). The framework consists of three components, namely: the information content component – models the information content of a term with respect to the whole collection, the information gain normalization factor – based on a small set of documents called the elite set deemed to be significant based on the distribution of the term in these documents, and the term frequency normalization function – component responsible for document length normalization and other collection specific statistics. Each component uses its own Probability Density Function (PDF) to calculate a value (Amati and Van Rijsbergen, 2002). Generally, DFR ranks documents by computing a gain in retrieving a document containing a term of the query. The significance of a term is evaluated using a PDF such as Poisson distribution. Words are divided into two classes: content (speciality) words and non-speciality words. Speciality words are rare words and appear in elite documents. Non-speciality words are common words and their distribution can be modelled using a function such as Poisson; words are randomly distributed across the whole collection. The distribution of speciality words can only follow Poisson in elite documents. The information gain or content of a term is detected by measuring the extent of its deviation from the distribution. We used DFR for retrieving documents for relevance assessments tasks (see Chapter 4).

Learning to Rank: LTR trains a machine learning algorithm to predict the relevance score for a document based on some fixed set of document features (Li, 2011).. Training examples consist of document-query pairs: feature values and relevance labels obtained from human annotators or derived from implicit relevance judgements such as clickthrough data. Datasets used in LTR are divided into a training set, validation set and test set. An algorithm is trained on the training dataset to construct a ranking model. Training involves finding optimal values of the model from the features, which will be used to rank unseen future requests. The constructed ranking model is used on the test set to rank documents and the performance of the model is based on this dataset using a quality evaluation metric such as Mean Average Precision (MAP) or Normalised Discounted Cumulative Gain (NDCG). The process of learning attempts to minimise the difference between the predicted ranking and

the gold standard, i.e. judgements given a dataset. This is done through optimisation; i) by minimising a loss function, i.e., a function that is used to measure how well a model is performing, and ii) maximising a quality evaluation metric.

Ranking using supervised learning is commonly divided into two categories: (i) based on how learning is done in terms of how a loss function is employed and (ii) based on the machine learning techniques being used. A loss function measures how well a model has performed given different values of the model parameter and is used to find the optimal parameters for the model, using an optimisation function (Li, 2011). The former categorises learning to rank algorithms as pointwise, pairwise and listwise. Pointwise and pairwise approaches reduce the problem of ranking into either classification, regression or ordinal classification problems. Generic machine learning algorithms are adapted for ranking. The loss function for pointwise approach is defined on a single learning example and usually charges for regression error. The pairwise approach is defined on a pair of two documents for the same query and the loss function charges when there is a wrong ordering of documents. Listwise approach uses ordered lists of documents as training data and learns a model to rank documents. Loss function is based on the ideal ordering of documents and charges for wrong ordering of documents. The latter groups learning to rank algorithms in terms of techniques used: (i) boosting – combining weak learners to create a more accurate model, (ii) large margin approach – using Support Vector Machine (SVM) techniques – model learns a separating hyperplane to maximise a margin and (iii) using neural networks model to train a ranking model (Li, 2011).

The pointwise approach uses training features of single documents, i.e., the input space consists of query-document similarity scores such as BM25 and output space as class labels, real values or grade values. The ranking model predicts the ordered class of a document or grade value. Examples of LTR algorithms using the pointwise approach are PRANK (Crammer and Singer, 2001) and MCRANK (Li, Burges, and Wu, 2007). The pairwise approach uses input features from document pairs and outputs partial order preference. In this set up, ranking is a classification problem, i.e, the model considers the question of what document should be ranked higher between a pair of documents. The advantage of this approach is that it is easier to obtain relative preferences than a complete ranked list for the training set. On the down side, results may be biased towards queries with more training examples. Examples of LTR algorithms using pairwise approach are Ranking SVM (Cao et al., 2006), RankBoost (Freund et al., 2003), LDM (Gao et al., 2005), FRANK (Tsai et al., 2007a), RANKNet (Burges et al., 2005), GBRank (Zheng et al., 2007b), QBRank (Zheng et al., 2007a) and MPRank (Li, 2011).

The listwise approach uses a list of ordered or ranked lists and their feature sets to learn a ranking model. The listwise approach differs from the other two approaches in the sense

that listwise focuses on accurately ordering objects while the others focus on correct categorisation of objects – predicting the relevance class of an object given a query. Learning a ranking model based on ranked lists is a specialised case of learning and, therefore, listwise approaches use new machine learning techniques to solve the problem of ranking. Listwise algorithms learn in two ways: (i) directly optimising IR evaluation measures such as MAP and NDCG; and (ii) using a loss function that indirectly optimise an IR quality evaluation measure. Examples of algorithms using listwise approach includes LambdaRank (Burges, Ragno, and Le, 2006), Adarank (Xu and Li, 2007), SVM-MAP (Yue et al., 2007), SoftRank (Taylor et al., 2008), RankCosine (Qin et al., 2008), Listnet (Cao et al., 2007), (ListMLEXia et al., 2008), LambdaMART, LambadaneuralMART (Papini and Diligenti, 2012) and DeepRANK (Li, 2011).

Traditional IR models tend to focus on retrieving relevant documents in terms of topicality. Machine learning algorithms have emerged as a way of combining evidence of relevance from several features that represent different desirable attributes of relevance. Several learning approaches and algorithms have been proposed in literature for LTR for different retrieval scenarios – monolingual retrieval, multilingual retrieval and ranking for relevance together with other desirable attributes such as freshness, diversity and understandability. We provide a literature review on these techniques in Section 3.2.3. We provide our experimental settings and re-ranking results integrating topical relevance and intelligibility using LambdaMART in Chapter 6.

2.1.4 Retrieval Evaluation Paradigms

The common experimentation paradigm in retrieval systems has been the empirical approach where systems are evaluated for retrieval effectiveness: how good retrieval systems are able to separate relevant and non-relevant documents. Since the 1950s, this system oriented approach has been widely adopted by researchers. However, the trend has changed as more user oriented needs are being solved using retrieval systems. A user centered or Interactive Information Retrieval (IIR) approach is becoming popular with researchers. Our thesis investigates both system and user oriented aspects of retrieval and both approaches have been employed in different parts of the thesis.

System-Centered Evaluation

The traditional IR evaluation approach is widely used to evaluate retrieved documents based on topicality (Sanderson, 2010). The approach, also known as the Cranfield paradigm, uses experimental collections. Collections consists of a set of documents, queries and relevance judgements also called qrels or relevance assessments. Using this approach separates the process of creating a collection from the retrieval techniques being evaluated. Standard

benchmark collections are created by different forums or groups, which are then used by researchers to evaluate their new retrieval techniques. In collection development, documents are collected. Assessors then browse the collection to come up with queries of interest. Usually, not all documents are assessed for relevance against the queries since it is expensive and time consuming. A process called pooling is used instead where only a subset of the documents are assessed for each query. The queries and corpus are distributed to participants of an evaluation to run their experiments. After submitting their results, the combined retrieved documents are assessed for relevance. Relevance assessments that are done may be binary or graded. Binary assessments have 'relevant' or 'not relevant' outcome, i.e., 0 or 1. Graded scores can take values that relatively quantify the relevance of a document, e.g., 0 for not relevant, 1 for marginally relevant, 2 fairly relevant and 3 perfect match. Using retrieval benchmarks means that it is possible to compare results from other studies as well as to reproduce results from other studies.

There are large scale evaluations that take place around the globe, namely: TREC (Text REtrieval Conference), USA, since 1992; [1]; NTCIR (NII Testbeds and Community for Information access Research), Japan, since 1999,[2]; CLEF (Conference and Labs of the Evaluation Forum), Europe, since 2000 [3]; and FIRE (Forum for Information Retrieval Evaluation), India, since 2008 [4].

User-Centered Evaluation

User centered evaluation presents a competing evaluation paradigm for search systems focusing on user aspects of retrieval (Kelly, 2009). The major challenge with the system oriented approach is that users are abstracted from the evaluation process, and search results assessment on the user side is oversimplified. User centered evaluation considers user's cognitive and affective retrieval perspectives as well as the physical aspects of the system. For example, users evaluate search interfaces, search results or other aspects of the search system. Users' search behaviour and interaction may be studied and evaluated through methods such as observation and from user implicit relevance data such as log analysis. The evaluation using these methods is diverse. Other methods from Human Computer Interaction (HCI) are used to evaluate interfaces and other aspects of search engines (Kelly, 2009). Discounted gain based evaluation metrics have been proposed for modelling user interaction and assessment of search results, such as time-biased gain measure (Smucker and Clarke, 2012) and understandability biased evaluation measures, focusing on readability of documents using rank-biased precision (Zuccon, 2016).

[1] https://trec.nist.gov/
[2] http://research.nii.ac.jp/ntcir/index-en.html
[3] http://www.clef-initiative.eu/
[4] http://fire.irsi.res.in/

Evaluation Metrics

Evaluation metrics are used to measure how well the retrieved results meet the user information needs – topical relevance of the returned results – although some of the metrics are used or adapted for user centered evaluation. Common evaluation measures are divided into two major groups: set based metrics and rank based metrics. Set based metrics are usually used with binary assessments, and commonly used metrics are precision and recall.
Precision: Precision is the the proportion of retrieved documents that are relevant.

$$P = \frac{\text{Total number of relevant results retrieved}}{\text{n}}, \tag{2.8}$$

Where n is the total number of retrieved documents.

Precision@K: Precision can also be calculated based on a threshold (Manning, Raghavan, and Schütze, 2008). For example, P@10 is Precision calculated for the top 10 documents.

$$P@k = \frac{\text{number of relevant documents retrieved}}{\text{k}} \tag{2.9}$$

Average Precision: The Average Precision (AP) of the results for a query is calculated by getting the average Precision for all values of n. (Manning, Raghavan, and Schütze, 2008).

$$AP = \frac{\sum_{n=1}^{N}(P@n)}{n} \tag{2.10}$$

Mean Average Precision: Mean Average Precision (MAP) is therefore calculated by taking the arithmetic mean of the AP values for each query.

Recall: Recall is the proportion of relevant documents that have been retrieved.

$$Recall = \frac{\text{number relevant results retrieved}}{\text{m}}, \tag{2.11}$$

Where m is the number of relevant documents for this query in the collection. Precision and Recall measure retrieval effectiveness – separating relevant documents and non-relevant documents – the assumption is that relevance is binary and all relevant documents are equally important.

Normalized Discounted Cumulative Gain: The Normalized Discounted Cumulative Gain (NDCG) allows for graded relevance judgements as opposed to the binary relevance judgements used in MAP (Järvelin and Kekäläinen, 2000). Based on the fact that users are less likely to look at the results further down the list, a rank-based discount factor is removed from the score of the Web pages or documents found further down the result list (Järvelin

and Kekäläinen, 2000). DCG for queries is calculated as:

$$DCG = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log(i+1)} \tag{2.12}$$

rel_i is the graded relevance of a document at position i in the search result list. The cumulated gain is discounted for any given position p. For each query, documents are arranged in descending order of their relevance to calculate their possible maximum DCG, known as Ideal Discounted Cumulative Gain (IDCG). nDCG is calculated by dividing DCG with IDCG, the DCG for ideal ordering:

$$nDCG = \frac{DCG}{IDCG} \tag{2.13}$$

Average nDCG is calculated by taking the arithmetic mean of the nDCG values for all queries (Järvelin and Kekäläinen, 2000). Relevance is assumed to be graded, throughout the thesis, and we have used nDCG as a primary metric for our evaluation.

Kendall's Tau: Kendall's Tau estimates the similarity of two ranking lists, i.e., the ground truth ranking and the ranked list produced by the model. Kendall's Tau is calculated as follows:

$$T_i = \frac{2c_i}{\frac{1}{2}n_i(n_i - 1)} - 1, \tag{2.14}$$

where c_i denotes the number of concordant pairs between the two lists, and n_i denotes the length of the two lists.

2.2 Aspects of Language

2.2.1 Bantu Languages

Bantu languages are spoken by over 240 million people in about twenty eight countries in Sub-Saharan Africa (Nurse and Philippson, 2006). Genealogically, Bantu languages belong to the Niger-Congo phylum (Greenberg, 1957; Meeussen, 1967) and is the largest branch in terms of number of speakers and geographical coverage. Bantu languages are characterised by their unique grammar: Subject:Verb:Object (SVO) word order, noun classes participating in a grammatical agreement system and a fixed order of affixes. Structurally, Bantu languages are agglutinating. Bantu syllable structure is open, taking the form: CV where V represents a vowel and C a consonant – words do not end with a consonant. Bantu languages are written using the Latin alphabet.

Bantu languages are uniquely identified by a character code system of three to four letters proposed by Guthrie (1967 – 71) (Guthrie, 1967). The first character in the code, an

uppercase letter, indicates the regional zone (A to S) and is followed by two digits in which tens indicate the language group and the units indicate the individual language. The code sometimes ends with a lower case letter, which corresponds to a dialect. For example, the Chewa-Nyanja group belongs to zone N, group 30, i.e., N30. N31 corresponds to Chewa-Nyanja or Nyanja-Chewa language while adding letters refers to specific dialects, i.e, N31a is Chichewa, N31b is Nyanja, N31c is Manganja and N31D is Nyasa (Mozambique) or Nyasa-Chewa. Figure 2.2 shows Guthrie's Bantu languages Zone (Maho, 2006). Mostly, geographical but some of the zones have been argued to be genealogical (Maho, 2006).

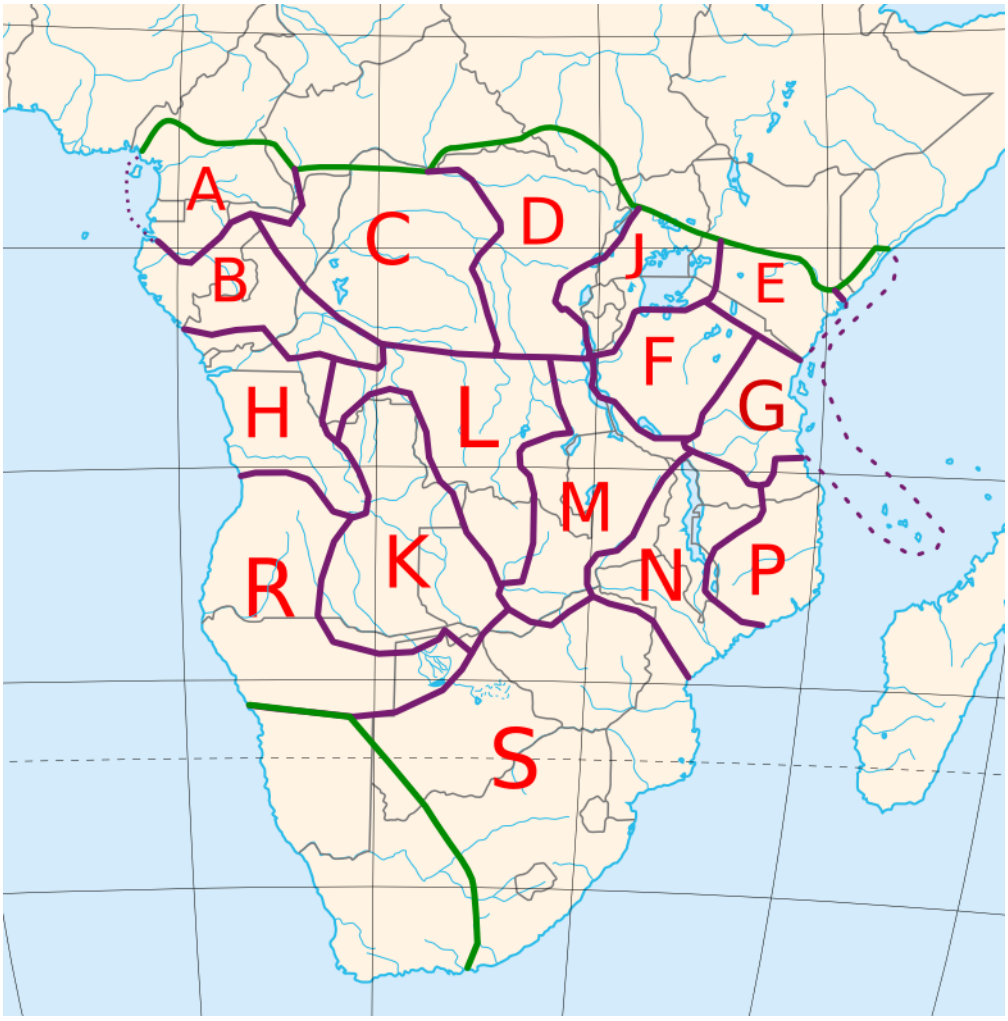


FIGURE 2.2: Map showing Guthrie's Bantu Zone.

2.2.2 Intelligibility

Intelligibility is the degree to which a speaker of a language understands the speaker of another closely related language (Gooskens, 2018). Closely related languages or genetically related are known to have one ancestral language – proto language – from which all the

descendant languages inherit linguistic features such as vocabulary and syntax. Orthographically similar words are words that are spelled the same while phonetically similar words are word pairs that have similar pronunciations. Related languages share content words, which belong to the core lexicon of the language (Haspelmath and Tadmor, 2009). Languages that are not related may share some linguistic features such as vocabulary as a consequence of language contact through lexical borrowing, whereby speakers adopt words from another language and transform them to be similar to the local language – nativization (Haspelmath and Tadmor, 2009). Words may also come into the vocabulary of another related language without major transformation, e.g., proper names – are transliterated from the source language to the orthography of the recipient language (Haspelmath and Tadmor, 2009). There are also words that are phonetically and orthographically similar but are not translation equivalents and these are called false friends but are few across language. Only orthographic similarity is explored in this thesis.

Intelligibility is affected by linguistic factors including vocabulary, phonetics, morpho-syntax and extra-linguistic factors such as previous language knowledge or exposure and attitude (Gooskens and Swarte, 2017). In linguistics, intelligibility is broadly measured by two methods: (i) opinion testing in which L1 speakers of a particular language rate themselves using a scale on how they understand another unfamiliar language under study; and (ii) function testing where participants complete tasks such as translation of a word list, or answer multiple choice questions from a text given in the task (Gooskens, 2013).

Intelligibility is usually expressed as linguistic distance: the smaller the distance the more related the languages are (Gooskens and Swarte, 2017). Linguistic distance is estimated using language features such as vocabulary, syntax and morphology. For example, the percentage of the number of non-cognates (cognates i.e., words with approximate similarity with respect to sound or (orthographic) form and equivalent meaning) across a vocabulary list of two languages is expressed as a lexical distance (Heeringa et al., 2013). Computational approaches based on information theory and statistics use metrics such as entropy (Jens et al., 2007), surprisal (Stenger et al., 2017; Hale, 2001; Goodkind and Bicknell, 2018) and perplexity distance to estimate intelligibility (Fischer, Vreeken, and Klakow, 2017; Gamallo, Pichel, and Alegria, 2017). These methods use language features derived from written text such as wordlists or corpora.

Group N Languages

Languages in group N are spoken in Southern and South–East Africa in countries including Malawi, Zambia, Mozambique and Zambia. Specifically, our languages of focus are identified as Citumbuka (N20), Chichewa (N31a) and Zambian dialect Cinyanja (N31b), Cisena (N40) and Citonga (N15). Figure 2.3 shows a map of regions speaking group N languages studied in this thesis. Group N languages are known to have major similarities based on

syntax, vocabulary and morphology, and have been argued to be truly genetically related (Nurse and Philippson, 2006). Kiso reported that Cisena, Citumbuka and Chichewa are not mutually intelligible, based on information obtained from informants (Kiso, 2012).

Chichewa is widely spoken in Malawi and several varieties and dialects are spoken in Zam-

FIGURE 2.3: Map of Malawi and areas of neighbouring countries speaking group N languages.
Source:Britannica [5]

bia, Mozambique and Zimbabwe. Chichewa is spoken as a native language by the Chewa people found in central Malawi, eastern Zambia and Western Mozambique in Tete Pronvince. The language is taught in public schools in Malawi as a subject and as a language of instruction in early classes. Chichewa is also used widely in urban areas as a lingua franca and the majority of L_1 speakers of Cisena, Citumbuka and Citonga speakers are familiar with the language. However, many Chichewa speakers are not familiar with Citumbuka, Cisena or Citonga as these languages are only spoken in specific areas and contact with the language is usually through travelling and media. Zambian Cinyanja and Chichewa are dialects of Cinyanja spoken in Zambia and Malawi respectively. Zambian Cinyanja has borrowed many words from other indigenous local languages such as Cibemba and Cilyambia as well as English through language contact. The Nyanja widely spoken in urban areas is

a different variety known as Town Nyanja. Malawi Citonga is spoken only in Malawi by the Tonga people in the northern part of Malawi, especially in the lake region. Citumbuka and Chichewa are related languages that share content words such as function words and affixes. The two languages also share some structural similarities although morphological rearrangements exist for some syntactical elements such as negation. Regular sound shifts also exist for both the core vocabulary and loan words. Both languages have borrowed words due to language contact with each other and other languages (Matiki, 2016). English is a major donor to both languages due to colonial history (Chavula, 2016; Matiki, 2016). Cisena is spoken in Malawi and Mozambique by the Sena people found on the southern tip of Malawi along the Shire River and in Mozambique in the Zambezia. The Mozambican Sena has largely borrowed from other local languages as well as Portuguese. These factors have made contributed to the differences of the Sena varieties spoken across the border. A comparative study of the two languages found that the languages are similar at lexical and grammatical level (Funnell, 2004). Our Cisena corpus described in Chapter 4 has text from both varieties.

In many cases, the borrowed forms of words are similar with some minor phonological alterations across the languages. Because of these factors, the languages share words in the core vocabulary through cognates, established loan words and foreign vocabulary such as technical words and named entities. However, no analysis has been done to understand the intelligibility of these languages, and Chapter 5 presents some data on the intelligibility of the languages. In some of the experiments, we added documents written in Luganda (JE15) – a Bantu language widely spoken in Uganda by the Buganda people – to include a non-neighboring language. Table 2.1 shows the classification information of the languages and the number of people who speak them as a first language.

TABLE 2.1: Number of speakers and (classification) codes for the languages investigated in group N and Luganda

ISO Code	**Classification Code**	**Language Name**	**First Language Speakers**
swk	N44	Cisena	900,000
ny	N31a	Chichewa	7,000,000
nya	N31b	Nyanja	1,000,000
tum	N21	Citumbuka	1,500,000
tog	N15	Citonga	170,000
lug	JE15	Luganga	8,000,000

Group S Languages

The problem of search interface design was investigated using Bantu languages spoken in South Africa belonging to group S. The languages investigated were isiZulu, Sesotho se

Leboa, Setswana, Tshivenda, isiXhosa, isiNdebele, siSwati, Sesotho and Xitsonga. Some of these languages are more similar and are further divided into sub-groups such as Nguni (isiZulu, isiXhosa, isiNdebele and isiSwati) and Sotho (Sesotho, Sesotho sa Leboa and Setswana). Nguni languages are spoken by the Nguni people known to have the same ancestral heritage – the languages are also known to be mutually intelligible. For example, speakers of isiZulu may communicate with IsiXhosa speakers without switching to a common language. Although isiNdebele belongs to the Nguni group, it has also borrowed from the Sotho family through contact. These languages are spoken as home languages and some are taught in schools in the specific regions (see Table 2.2). Additionally, many Bantu speakers are multilingual – they are able to speak at least one foreign language such as English or Afrikaans and other local Bantu languages (Nurse, 2008). Many people whose first language is a Bantu language in South Africa speak IsiZulu as a second language. Figure 2.4 shows the map of the dominant languages in South Africa including the eleven languages used in our interface design study in Chapter 5.

TABLE 2.2: Distribution of First Language Speakers (2011 Census in South Africa. Although some of the languages or dialects are spoken in neighbouring countries, here our focus was on South African speakers.

ISO Code	**Classification Code**	**Language Name**	**First Language Speakers**
zul	S.42	isiZulu	11,587,374
xho	S.41	isiXhosa	8,154,258
nso	S.32	Sesotho sa Leboa	4,618,576
tsn	S.31	Setswana	4,067,248
sot	S.32	Sesotho	3,849,563
tso	S.53	Xitsonga	2,277,148
ssw	S.43	siSwati	1,297,046
ven	S.21	Tshivenda	1,209,388
nbl	S.44	isiNdebele	1,090,223

2.2.3 Bantu Morphology

Morphology is the study of the internal structure of words. This includes the way words are broken down into their smallest constituents or morphemes, and the processes of producing new words. Generally, morphological processes happen when some character sequence, like an affix, is added to the base form of a word or root, i.e., affixing. When an affix is added at the beginning, it is called prefixation while at the end is called suffixation. Morphemes of a language are its smallest meaning bearing units, which may be affixes or roots. A stem consists of the root and derivational affixes. Free morphemes are those that can stand on their own as words while bound morphemes are attached to other morphemes to form words.

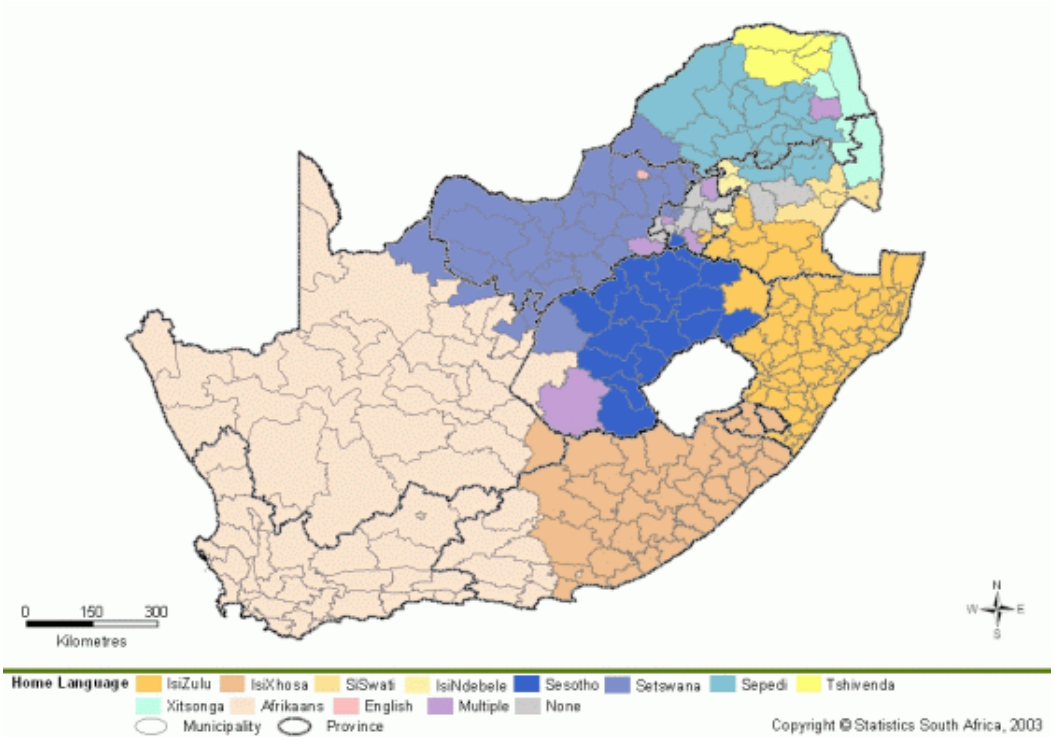


FIGURE 2.4: Map showing dominant languages in South Africa. Source: Statistics South Africa (census 2003)

Inflection and derivation are morphological processes that are common in many languages. Inflectional morphology is concerned with various forms of words (same meaning), used in different syntactic contexts. For example, the common inflectional categories for nouns are singular and plural (e.g., *galu/agalu* in Chichewa; *dog/dogs* in English). Derivational morphology encompasses processes involved in the formation of new words from existing ones, usually by affixing (e.g., *-lim-/m-lim-i* in Chichewa; *farm/farmer* in English). Compounding is the process of forming new words from multiple existing words or free morphemes (e.g., *mwana-dala* in Chichewa; someone who looks younger than his or her age in English). Reduplication is when an existing word or part of it is repeated to form another new word, typically with a slight change in word meaning. For example, in Chichewa the whole word can be reduplicated (e.g., *munthu /person* can be reduplicated to *munthumunthu*; *real person* in English).

Bantu languages are agglutinating, i.e., there are clear boundaries between stems and affixes. For example *mi*-tengo 'tree-*s* in English', has '*mi*' to indicate plural while *tengo* is a root. For most Bantu languages, derivational processes are mostly suffixal and inflectional processes are prefixal. Compounding and reduplication are common in many Bantu Languages. The structure of words largely is templatic, i.e., morphemes take specific position, also called slots. These features make Bantu morphology complex and unique.

Nominal Morphology

The Noun Class System is a common and central feature for Bantu nouns. Every noun belongs to a class and the class prescribes the prefix the noun can take. Noun classes come in pairs of singular and plural. Twenty four (24) classes have been identified for Proto Bantu (Meeussen, 1967; Welmers, 1973). However, only a subset with varying number of these classes are available in all languages. Bantu noun classes are referenced using a numbering system that was proposed by Bleek (1862) and Meinhof (1899, 1932) (Katamba, 2003). Noun classes across languages vary from language to language. However, closely related languages usually have a similar set of prefixes and classes. The head noun of a sentence determines the affixes that other elements of the construction take. Thus, each word in a sentence or phrase have a prefix, which agrees with the head noun. The noun class prefix and agreement marker are the same for some classes. For example, classes 7 (*ch-*) and 8 (*zi*) for Chichewa have prefixes *chi-* and *zi-* (e.g.., *Ch-i-*patso *ch-o-*koma *ch-a-*bedwa in Chichewa; the tasty fruit has been stolen in English). The prefixes *ch-* on the last two words of the sentence are concordial morphemes that agree with the head noun. The head noun also has the prefix *ch-*, which also indicates the noun class of the head noun. In other cases, the prefix on the noun and the concord affixes are different in form. In addition, some classes take zero prefixes but demand a specific concordial affix (e.g., *mu-nthu w-a-menyedwa* in Chichewa; a person has been beaten in English – the concord prefix *'w-a-'* goes with noun class 1). The following show examples of concords or agreement markers demanded by some of the classes:

(a)	Noun + Verb	tsamba la-gwa	'the leaf has fallen'
(b)	Noun + Verb	masamba a-gwa	'the leaves have fallen'
(c)	Noun + Possessive	tsamba la-thu	'our leaves'
(d)	Noun + Possessive	masamba a-thu	'our leaves'
(e)	Noun + Adjective Stem	tsamba la-li-kulu	'big leaf'
(f)	Noun + Adjective Stem	masamba a-a-kulu	'big leaves'

Nouns are marked morphologically by a noun class prefix. The NCS specifies the prefix that expresses number (singular, plural and collective). Paired classes, also called genders, express number. The prevalent nominal morphological structure is therefore fixed and linear, consisting of the prefix and stem slot. The prefix slot can hold null, one or multiple morphemes.

noun = [noun class marker] + [stem]

However, some nouns may have suffixes, i.e., clitics attached to them (e.g., a noun and demonstrative *galu uyu* 'this dog', is usually shortened to *galuyi'*). This suggests that a noun

may be combined with suffixal material to produce a word with three slots, i.e., prefix, stem and suffix. This feature is available in a few Bantu languages such as Chichewa (Mchombo, 2004).

Verbal morphology

Bantu languages are characterised as verby, i.e., a verbal form contains morphemes of categories that are usually free morphemes in other languages (Nurse, 2001). For example, *a-na-mu-meny-ets-a* in Chichewa 'She got her beaten' has morphemes of categories Subject Marker (SM) (also known as Concord affix), Tense Marker (TM), Object Marker, Root, Causative, Final Vowel (FV). The agglutinative nature of Bantu languages is very much vivid in verbs as several morphemes are attached to the verbal root one after another in a linear order. The structure of verbs together with the Noun Class and concordial system contribute to the complexity of the Bantu morphological system. The Bantu verbal morphological structure is consistent across languages, with similar slots and morpheme order.

Bantu verbs contain several elements with features common in many languages. The common order of affixes in Bantu is linear or templatic: affixes tend to follow a fixed order. Consequentially, a system of slots or templates have been proposed for Proto-Bantu (Meeussen, 1967; Nurse, 2001) as well as pan African (Maho, 2007). Generally, the Bantu verb consists of categories that have been given the following terminology:

- Verb root, e.g., *-tum-* in *ku-tum-a* 'to send ' in English
- I-stem, e.g., *-yendets-* in *ndi-na-yendets-a* 'I drove' in English.
- Macro-Stem = verb stem + object marker as *wayend-* in n*di-da-wa-yende-ts-a* 'I made them walk' in English,
- Base or verb radial = root + extensions(suffixes) *tum-idw-* in *ku-tum-idw-a* 'to be sent' in Chichewa
- Stem = root + extensions + final vowel as *tum-idw-a* in *ku-tum-idw-a*
- Extensions = *-idw-* in *ku-tum-idw-a*. A verb may contain multiple or repeated extension morphemes.
- Object marker, a category referring to an object in a construction, e.g., *-ka-* in *ndi-na-ka-yendetsa* 'I drove it' in English.
- INFL markers or inflectional markers consists of Subject Marker (SM) and Tense Marker (TM).

Prefixation in Bantu verbs is common for inflectional morphology. The common categories expressed in most languages for prefixal inflection include NEGative (NEG), Subject Marker (SM), Tense and Aspect (TA) and Object Marker (OM) and usually appear in the order NEG, SM, TA and OM. However, for some languages, the SM comes before the NEG category. Therefore, the fixed ordering of prefixes allows for a slot system or templatic affix ordering. In this type of template, a slot may contain multiple or zero inflectional verb prefixes. The morphemes for prefixation are normally of CV syllable structure. In other languages, such as Citumbuka, the NEG and/or TA categories are expressed as suffixes. Usually, the final position (suffixal) for templates of such languages cater for inflectional affixes.

Suffixation is common for derivational morphology in Bantu words, although a few suffixes appear for inflection. The order of slots or positions of categories expressed by suffixes is fixed: extension(s), final vowel and post final vowel or final suffix. The majority of the suffixes for the verb occupy the verbal extension slot. Verbal extensions express categories for changing verbal valency and concerns verb to verb derivation. The common categories for Bantu verbal extension are causative, applicative, reciprocal and passive, and take the form: -VC-. A single verbal root may take multiple extensions or the same extension may be repeated. There are two major differences on the order of extensions when multiple extensions appear in one word. In some languages such as Chichewa, the order of the verbal suffixes is fixed and the default ordering follows the morphological template: Causative (CAUS) – Applicative (APP) – Reciprocal (REC) – Passive (PASS) or CARP (Hyman, 2003). The remaining slots contain morphemes that express other categories, i.e., the Final Vowel (FV) and the Final Suffix. The final vowel is present in all Bantu verbs and the vowel *-a-* is common in almost all languages (Nurse, 2001). In Chichewa, all verbs end with *-a* except the imperative: the plural imperative ends with *-i*. The final position take a final suffix that expresses categories such as mood and clitics. Additionally, in languages like KiSwahili and Chichewa, the plural for imperatives and subjunctives are expressed by a morpheme that takes the final position.

Other Word Categories

Adverbs tend to modify words of other categories such as verbs, adjectives and other adverbs. Similar to adjectives, the number of true adverbs is limited for many Bantu languages. Usually, a prefix is required to create an adverb from stems of other word categories (e.g., *pa-* is added to *-ng'ono* 'small' in Chichewa to form an adverb *pang'ono* 'slowly in Chichewa'). Reduplication similar to that of verbs is also common in Bantu adverbs. The templatic form of the Bantu adverb consists of two slots ,i.e., prefix and stem.

adverb = [possessive marker] + [noun class marker] + [stem]

An adjective is a syntactic category that specifies the attributes of a noun. Similar to nouns, Bantu adjectives come in two forms: derived and underived or 'true adjectives' (Mchombo, 2004; Matiki, 2000). True adjectives are created using adjectival roots, while derived adjectives are created from roots of other word categories. Due to the Bantu noun class and concordial system, adjectives contain noun class markers and take the noun class prefix of the noun they qualify. The markers on the adjectives serve as agreement markers for both number and noun class. For example, in *ka-bango ko-koma delicious small mango*. *ko-* on *-koma* agrees with the noun class of the noun. Therefore, adjectives or adjectival phrases tend to take any marker that corresponds to the noun they modify, i.e, they do not belong to a specific class as nouns do. Adjectives can take negative markers to create antonyms.

Most Bantu languages have a small set of true adjectives. True adjectives are formed by adding a noun class prefix to an adjective root. The prefix is the same as the prefix of the noun they qualify. However, some languages like Chichewa take double prefixation (Matiki, 2000; Mchombo, 2004). An adjective is made up of a possessive agreement marker (POSS), class prefix (Pr) and an adjectival stem (Matiki, 2000) (e.g, *chi-thunzi cha-chi-kulu* 'big picture' , *cha-* is a possesive marker and *chi-* is a noun class marker for class 7). Derived forms of adjectives come from verbs and nouns. The derived forms usually do not take any class prefixes: the nominal forms already contain the class prefix and the verbal form is usually in infinitive form (also noun class 15). The possessive agreement marker prefix or relative marker is added to a verbal or nominal stem to form an adjective. Thus, the Bantu adjective has the following morphological shape:

adjective = [possessive marker]+ [noun class marker]+ [stem]

2.3 Summary

The chapter has provided background of the topics discussed in this thesis. We have discussed the fundamental issues in retrieval, including retrieval models used in experiments in our work and evaluation approaches used in IR. We have also introduced the languages that have been used in the thesis, including their morphology and intelligibility issues. Bantu words have unique structure: agglutinating and templative. This structure is common across many languages, and provides opportunities for shared resources and tools. We also have seen that many languages Bantu languages are related and closer languages are clustered into smaller groups that are made up of closely related languages.

Chapter 3

Literature Review

The widespread use of search systems has driven the demand for such systems to meet new diverse information needs of users. Accordingly, the field of IR has proposed several theories and algorithms – from those that use document and query features to estimate document relevance to complex document ranking function learners that use user(s) search history and features beyond relevance to meet subjective information preferences of different users.

The goal of this chapter is to provide an analysis of previous research pertinent to the work presented in this thesis. We investigate different facets of retrieval for related languages and neighbouring fields. We first provide an analysis of literature for retrieval involving related languages. Retrieving documents that may be written in several different languages requires techniques for merging and ranking such results. Secondly, we discuss approaches for merging and ranking search results including, methods for learning to rank search results using multiple criteria such as recency and diversity. Users interacting with multilingual search results may have unique search behaviour and preferences. Thirdly, we provide an analysis of studies reporting on user perspectives and preferences in MLIR. Finally, we discuss morphological pre-processing, focusing on stemming and methods that use string similarity to cluster morphologically similar words.

3.1 Using Similarities of Languages in Retrieval

Languages are diverse and evolve with time. Nevertheless, certain languages are similar and share more features than others. Similarities among languages, such as shared words and similar sounds, are due to genetic relationships among them, i.e., related languages are known to have a common ancestor language. Language contact contributes to shared features among languages that are not closely related. A typical example of language similarities brought about through contact is lexical borrowing – using a name of a concept from another language (Taylor and Grant, 2014). So far language similarities, such as in vocabulary similarity, have been explored to retrieve documents written in related languages without the translation step typically used in CLIR and MLIR systems (Buckley et al., 2000; Gey, 2007; Chew and Abdelali, 2008). Such a retrieval setting, with no query or document

translation for CLIR, has the following form of interaction:

Input: Queries specified in a single language with some shared features with the corpus language(s).
With: Limited translation help such as spelling transformation rules for changing words from one language to another and fuzzy character matching of strings.
Output: A ranked document list written in languages similar to the language of the query. Ranking is based on pure similarity scores — not considering score normalisation or document language.

Generally, untranslated queries, together with fuzzy string similarity matching methods, have been used to match index and query terms for closely related languages in CLIR. This is done on the premise that matching is possible due to similarities in words across languages. For instance, Buckley et al. (2000) used English-French cognates with spelling rules to perform CLIR between English and French. Järvelin et al. (2006) used fuzzy string matching techniques, including Transformation Rule based Translation (TRT), n-grams and skip-grams, to perform dictionary independent query translation on Swedish and Norwegian CLIR and found no significant difference between their best performing matching approach and using a dictionary for translation. Similarly, CLIR with no query translation based on script similarities involving non-alphabetic languages was investigated by Gey (2007). Chinese queries were used on Japanese text and vice versa based on the assumption that the Japanese Kanji alphabet was derived from Chinese characters. Overall, these studies reported lower performance than retrieval using query translation, although in some cases the results were comparable.

Likewise, Chew and Abdelali (2008) investigated script similarity and genetic relatedness for Indo-European and Semitic languages. The goal of their work was to study the effects of language relatedness on retrieval effectiveness. A Latent Semantic Indexing (LSI) model was used with parallel corpora of related and unrelated languages. The study concluded that retrieval improves as the number of languages for parallel text in training increases and that text from genetically related languages significantly boosts retrieval effectiveness, while script similarity did not improve retrieval effectiveness. The work of Gey (2007) and Chew and Abdelali (2008) presents an interesting research perspective – investigating script similarity for languages, which are not similar. It would be interesting to study matching, ranking, and user perspectives for closely related languages that use different scripts but are historically similar. Languages like Hindi and Urdu provide that scenario with some complex social-linguistic issues – Hindi uses the Devanagari script and Urdu uses Perso-Arabic script but the two languages are known to be dialects of Hindustani (Hans, 2006).

TABLE 3.1: Studies involving related languages. Only matching of query and document terms has been studied.

Author and Year	**Test Collection**	**Transformation Techniques**	**Evaluation**
Buckley et al., 2000	French and English using SMART and TREC 6	Spelling rules and concepts	Comparable precision and recall
Gey, 2007	Chinese and Japanese in NTCIR-6	No transformation	Comparable precision and Recall
Järvelin et al., 2006	Norwegian and Swedish, CLEF 2003	RTR and Skip-grams	No significant difference for precision and recall
Chew and Abdelali, 2008	Indo-European and Semitic using Quran and Bible parallel corpora	Latent Semantic Space with SVD	Improved precision

Retrieval for related languages has been applied mainly where translation is excluded in MLIR/CLIR to take advantage of linguistic and orthographic similarities of languages involved and to avoid costs associated with translation systems. Results obtained using this approach seem to be comparable or worse relative to classic approaches such as using a dictionary for query translation. Table 3.1 provides a list of studies focusing on related languages in retrieval and how their performances compared with other approaches. While these studies only explored the matching of queries and documents for similar languages, they are an appropriate starting point to investigate any opportunities when similar languages are involved – the main question was whether it is possible to retrieve relevant documents based on language similarities. However, there are several aspects of retrieval in the context of related languages that can be explored further. Firstly, the assumptions made about users of such systems is unclear – whether the matching documents would be translated or users would be able to understand the material through multilingualism or intercomprehension. More importantly, it is unclear if the retrieved documents would be useful to users. Moreover, user interactions with search results retrieved in this context have been ignored – user preferences in terms of ranking and presentation of search results, their interaction behaviour in the context of cognitive and affective responses. Therefore, a new direction of research focusing on supporting user contextual needs and interactions is required in this area. Previous work on MLIR and CLIR can provide insights and a starting point for this retrieval context. We provide a discussion of user interaction studies in the context of CLIR and MLIR in Section 3.3. Again, to meet user information needs and preferences, search results need to be merged and ranked to improve user satisfaction. We

provide a discussion of studies of search results merging in Section 3.2.

3.2 Search Results Ranking

Retrieval systems return ranked lists of documents, which users consult to locate relevant information. Better ranking algorithms rank relevant search results highly to minimise the effort applied to find information that is relevant to meet users' information needs given their search contexts. Multilingual information retrieval systems return search results written in multiple languages in response to a query submitted by a user in a particular language (Peters, Braschler, and Clough, 2012). A common approach to handle MLIR involves a query translation step in which user submitted queries are translated to the languages of information sources using resources and tools such as multilingual dictionaries and machine translation systems (Peters, Braschler, and Clough, 2012). Additionally, matching documents in MLIR may need to be re-ranked, i.e., if merged results are required, documents written in different languages should be ranked in such a way that document utility or relevance should go down as the user moves down the list. However, documents may come from different sources with their own ranks and/or similarity scores and may require multiple lists to be merged to present the user with a single ranked search result document list. Also, user language competencies may differ and their level of understanding of the returned documents may be different. Therefore, more factors come into play when retrieval systems deal with documents written in several languages. We first start our discussion with research on aggregating search result lists from federated search, data fusion, and meta-search, and then we discuss merging in MLIR and end with LTR approaches that combine several features for ranking and/or merging search results.

3.2.1 Merging Search Results

Search result merging refers to combining top ranked retrieved documents from either several collections or search systems or models, and the new ranked results are presented to the user in what is called federated search, meta-search, and data fusion (Shokouhi and Si, 2011). Different merging algorithms have been proposed for different types of results aggregation or availability of data – for example, similarity scores between documents and queries used for ranking in initial search systems may not be available for meta-search but ranks only.

Several algorithms have been proposed for merging search results using raw (original), normalised similarity scores, and supervised learning (Shokouhi and Si, 2011). Merging search results using raw similarity scores is not effective as similarity scores may come from different ranking models such that corpora from which the scores are based may have different statistics, making the scores not comparable (Callan, 2000). Callan, Lu, and Croft

(1995) proposed the CORI algorithm to linearly combine scores. CORI uses a Bayesian Inference Network Model and has two phases, namely: i) resource selection in which the top k databases based on collection or database scores are selected, and ii) result merging in which collection selection scores and collection specific document scores are combined to rank documents to a single list. CORI weights results from a particular list based on the predicted relevance of the collection (collection score). Steidinger (2000) used an interleaving model to combine results based on ranks in different variants of Round Robin (RR) (simple RR, RR Block, RR Random) based on a RR scheduling strategy. Results of simple RR model were worse than the two other variants.

Shaw and Fox (1994) proposed a series of methods for selecting relevance scores to be used in result merging of several search result lists, namely: CombMax (maximum score), CombMin (minimum score), CombSum (sum of scores), and CombMNZ (sum of scores divided by the number of collection). CombMNZ was shown to be better than members of its family using a TREC-3 collection and START retrieval system (Shaw and Fox, 1994). COMBMNZ has been widely used as a baseline for result merging in literature (Markov, Arampatzis, and Crestani, 2013; Aslam and Montague, 2001). Lee (1997) showed that normalising similarity scores before combining them using any of these combining methods yielded better results using a TREC-3 collection. Most recent analyses have shown better performance for CombMax than its counterparts (Markov, Arampatzis, and Crestani, 2012).

Bartell (1994) was the first to report on supervised results merging from different collections or search systems, by proposing an adaptive algorithm to train a neural network to merge search results lists from multiple sources. Later, Vogt (1999) used linear regression to come up with weights for combining similarity scores to improve average precision for merging results from several resources. The results from these studies were promising but the availability of data at the time hindered further exploration of these data hungry methods.

Aslam and Montague (2001) proposed supervised and unsupervised metasearch aggregation algorithms based on ranks only, including Bordafuse, weighted Bordafuse, and Bayesfuse. Bayesfuse uses a Bayes probabilistic model and calculates the probability that a document of a given rank is relevant or not. Bordafuse merges lists based on a voting model that awards points to documents based on how many other documents are ranked lower than itself and sums the score from each system or ranked lists. Weighted Bordafuse learns the weights to associate with each collection. Weighted Bordafuse performed better than the unweighted Bordafuse but had comparable results with CombMNZ. BayesFuse reportedly had better performance than CombMNZ.

Si and Callan (2005) proposed a merging model that uses returned utility as a measure of how many relevant documents a particular retrieval system or resource can return, and

trains a model to learn the probabilities of the relevance of all documents in all collections, i.e., merging is based on the retrieval effectiveness of each search system or resource. The combination step maximises the returned utility (return utility estimates how much relevant content a search engine can return rather than how much relevant content it has). The proposed method was compared with the CORI merging algorithm using TREC Web collections and TREC news/government test collections and obtained better results. The approach is similar to the supervised collection fusion strategy proposed by Voorhees, Gupta, and Johnson-laird (1995). The approach starts with the training of queries to learn the distribution of relevant documents. The document distribution is used to estimate the number of relevant documents in each resource to find the cutoff point for retrieval for each resource or collection (Voorhees, Gupta, and Johnson-laird, 1995; Voorhees, Gupta, and Johnson-Laird, 1995). The approach was evaluated on TREC collections and was compared with retrieval from the use of a single resource. The results obtained show degradation of precision by 10% when multiple resources are used.

Lillis et al. (2006) proposed Probfuse, a supervised probabilistic approach that aggregates results based on the probability of a document being relevant using segments (instead of using ranks, segments are parts of the ranked lists generated by dividing each returned ranked list into segments, for example, dividing the returned lists into parts of ten results). Training data is used to learn the distribution of relevant documents and these are used to estimate the relevance of documents from different search systems to produce a single ranked search result list. The approach was compared with CombMNZ on TREC3 and TREC5 collections and Probfuse produced better results.

Several approaches have been proposed for result merging for data fusion, meta-search, and federated search. Table 3.3 tabulates studies on merging search results. Surprisingly, early work on result merging includes supervised learning and probabilistic models. These approaches did not produce good results at the time, as the amount of data available was limited. Later the trend changed, and much of the work focused on using similarity scores – based on the statistics of the collection and query and document pairs. Using similarity scores did not produce consistent results – the same approaches would result in better or worse results in different studies and evaluation for data fusion (merging lists generated by different retrieval models run on the same collection) based on improved retrieval effectiveness compared with single best performing system, model or collection, i.e., combination of evidence (Ng, 1998). Many studies were unable to beat the performance and any recorded improvement diminished as the number of systems increased, e.g., four (4) was found to be the best number (Vogt, 1999). Result merging was a hot topic in the '90s but research efforts did not produce excellent results – it is very hard to improve retrieval for queries that have very few relevant documents using multiple searches, especially if there is a system that retrieves relevant documents. The availability of data has seen the rebirth of search results

TABLE 3.2: Listing of result merging algorithms. Several algorithms have been proposed from supervised methods to algorithms that use collection, and query and documents statistics.

Author and Year	Approach	Evaluation
Shaw and Fox, 1994	combMin, CombMax, Combsum, CombMNZ	CombMNZ Better
Bartell, 1994	Neural Networks	Results better than individual retrieval systems using Cranfield collection
Voorhees, Gupta, and Johnson-laird, 1995	model the distribution of relevant documents in each collection collection	10% effectiveness of single collection
Callan, Lu, and Croft, 1995	CORI	TREC volume 1, results promising
Lee, 1997	combMin, CombMax, Combsum, CombMNZ	CombMNZ better Average Precison on TREC-3
Yager and Rybalov, 1998	(Supervised) Fusion parameter estimation	No retrieval experimentation and evaluation
Vogt and Cottrell, 1998	Weights based on linear regression	Better results than single system using TREC 5
Steidinger, 2000	Round Robin and variants, raw score and collection weighting	Round robin better
Aslam and Montague, 2001	BordaFuse, BayesFuse, Weighted BordaFuse	Borda fuse comparable with COMBMNZ and Bayes-fuse better results
Si and Callan, 2005	Returned utility maximisation	Improved retrieval effectiveness
Lillis et al., 2006	Probfuse	better than ComMNZ

merging and the trend has moved back to supervised merging of results with focus towards Learning To Rank (LTR) approaches. LTR methods will be discussed in Section 3.2.3. LTR provides opportunities to merge results not only based on relevance but on new objectives such as understandability or intelligibility in our case.

3.2.2 Multilingual Search Results Ranking

Multilingual retrieval systems retrieve and rank documents written in several languages to meet a user's information need expressed in a single language. The returned documents, which are written in different languages, may be presented to the user as a single result list to assist the user to find relevant content quickly, or as multiple result lists separated by language. Accordingly, there are two common multilingual search results presentation formats: interleaving results among different languages; or separating results by language and

presenting them in different tabs or pages (Steichen and Freund, 2015). Additionally, three architectures for multilingual retrieval systems have been proposed, including distributed, centralised and tagged centralised. Distributed architecture based MLIR systems are modelled after federated search systems, i.e, different information sources contain documents in different languages (Lin and Chen, 2003a). Retrieval systems using centralised architecture work as monolingual search systems where documents in several languages are indexed and retrieved as though they were written in a single language (Lin and Chen, 2003a). Tagged centralised systems use a single indexing system but document and query terms are tagged with language identifiers (Lin and Chen, 2003a).

The distributed MLIR approach has been widely investigated; the approach consists of three major components (Lin and Chen, 2003a): query translation, document retrieval and result merging. The primary challenge for distributed MLIR is to merge results from multiple sources in an order that will be useful to the user, i.e., including relevance and preference. Similarly, centralised MLIR systems face the same challenge of meeting user information needs as well as preferences. An example of user preference could be user language competencies in terms of their ability to understand text written in a particular language and user's preference to read certain topics in specific languages, which may affect their experience with the system, i.e., the utility of a highly relevant document may be affected by the user's understandability of the document. Therefore, merging and ranking of results is necessary if results are to be presented as a single result list. The same challenges of merging search results resurface, since collection specific issues such as size and number of relevant documents for each query in each collection may lead to similarity scores to be incomparable (Shokouhi and Si, 2011). However, merging multilingual search results has language specific issues such as differences in quality of translation, orthographic, and morphology issues as well as user language preferences. Earlier work in multilingual systems ignored user preferences and focused on retrieval effectiveness in the context of relevance.

Several algorithms for result merging have been proposed: (i) using normalised and raw similarity scores (Lin and Chen, 2003a); (ii) downloading and translating retrieved documents (Si et al., 2008); (iii) translation of retrieved documents to the query language (Chen and Gey, 2004); and (iv) machine learning approaches that learn to rank documents using user preferences (Gao et al., 2009).

Lin and Chen (2003a) investigated several merging algorithms including raw score, normalised score, round-robin and normalised by top-k with translation penalty and collection weight. Similarity scores generated by the ranking algorithm for each source or language using methods such as cosine similarity and Okapi BM25 have been proposed for merging multilingual search results (Lin and Chen, 2003a). This method assumes that the similarity scores of documents from different language collections are comparable. Multiple results lists are merged into a single ranked list using the original scores. Round Robin interleaves

documents by ranks produced in the original lists, i.e., documents on position one from all languages are ranked first, then on the second position and so on. Normalised score transforms scores to be comparable – for example, dividing each score in the list with the maximum score generated for the result list and, after adjusting scores, all results are put into a pool and sorted by the normalized score. Normalised by top-k algorithm was proposed to overcome challenges in MLIR such as translation ambiguity, out-of-vocabulary words, and the availability of relevant documents in a particular language given a user query. In this approach, similarity scores are normalised by dividing the scores in each list by the average score of the top-k documents. The documents are merged using w_i, which is the weight calculated as a translation penalty, and collection weight, which estimates the probability of retrieving relevant documents in that collection. The study produced mixed results – comparison of results with raw scores and Normalised by top-k produced comparable results; while better results were obtained for comparison of Normalised by top-k with round robin and normalised scores (Lin and Chen, 2003a).

Savoy (2003) used logistic regression to calculate the probability of documents being relevant given a rank (logarithm of the rank) for each collection and Retrieval Status Value (RSV) or a scoring function. The approach was applied on MLIR using a German, English and French CLEF collection and better results were obtained compared to normalised scores, raw score and round robin.

Chen and Gey (2004) proposed a MLIR approach that translates both query and documents: user query is translated to the collection languages and documents from the collections are translated to the query language. Results obtained using document translation experiments on CLEF 2001 and 2002 using English, French, German, Italian, and Spanish were better than the translation of queries only.

Nie and Jin (2003) proposed an architecture for indexing multilingual documents by tagging terms with language tags or ids in a centralised index. The language of the document is first identified, then the text of the document is pre-processed according to the rules and resources of the identified language, i.e., stemming, adding synonymy, and removing stop words, later language tags are added to the terms and, finally, the terms are indexed in a single index regardless of language. Querying is done in the same order: the query is translated and pre-processed using language specific resources and tools and language tags are added to the terms. The query is run as in monolingual retrieval. The approach is compared with Round Robin and raw scores using French, German, Italian, and English documents from the CLEF 2002 test collection. The approach produced slightly better results than Round Robin and raw scores.

Martínez-Santiago, Ureña-López, and Martín-Valdivia (2006) used an approach called 2 step scoring function. The approach first matches queries with their respective monolingual

collections; extracts concepts based on query translations, i.e., a concept is a term with its corresponding translations in other languages; builds a new centralised index based on the top 1000 documents from each collection using the concepts(term frequencies and index are based on concepts), and ranks documents using new similarity scores. Document frequency is calculated differently based on sum of documents with terms in each concept. The results from the 2 - phase approach based on 2001 to 2003 CLEF collections using English, French, German, and Spanish were better than such methods as round robin, normalised scores, and logistic regression.

Similarly, Si et al. (2008) proposed a merging algorithm that downloads a small set of documents from each collection, translates the documents, indexes the documents on a centralised index and calculates new similarity scores. A logistic transformation model was used to learn the transformation between the source scores and comparable scores, i.e., the model estimates comparable similarity scores of all retrieved documents. The approach was evaluated on the CLEF multi-8 merging task (Dutch, English, Finnish, French, German, Italian, Spanish and Swedish). The approach reported better results than the simple logistic merging approach.

Research on result aggregation emerged in the early 90s due to the invention of search engines such as Google, Alta Vista and AskJeeves. Multilingual search results merging became a prominent topic in the early 2000s and evaluation forums such as CLEF in Europe and NTCIR in South East Asia introduced tracks on this topic. Table 3.3 provides a list of studies on multilingual results merging algorithms. Performance of the proposed algorithms has been inconsistent with raw scores emerging to be better or comparable or worse with many algorithms. This may be due to factors such as translation quality and ambiguity, collection specific issues in terms of collection size and number of documents for each query, and query complexity. However, in many studies the normalised scores approach has performed better than raw scores, probabilistic approaches have performed better than normalised scores and document translation has performed better than any of the methods. Due to limitations of translation tools and resources, retrieval effectiveness of scarce resourced languages would be low. Supervised methods have been proposed in the early stage of research for result merging but was not explored further until late 2000s. LTR approaches have been proposed as a ranking method that can learn how to rank search results using hundreds of features. LTR algorithms have been extended to handle Multilingual search and federated search result aggregation, i.e., learning to aggregate search results.

3.2.3 Learning to Rank Beyond Relevance

LTR in the context of multilingual retrieval has not been widely investigated. Tsai et al. (2007b) used the FRANK algorithm to learn the weighting for different features such as

TABLE 3.3: Listing on Multilingual Results Merging Algorithms

Author and Year	**Languages**	**Approach**	**Evaluation**
Savoy, 2003	English, Italian, French and German	Logistic regression	Better than raw score, round robin and normalised scores
Lin and Chen, 2003b	English, Chinese and Japanese	Normalised top-k with translation penalty	normalized-by-top-k merging better but comparable to raw score
Martínez-Santiago, Ureña-López, and Martín-Valdivia, 2006	Spanish, German, French, Italian, English	2 step RSV	Better than round robin, raw score, normalised score, logistic regression
Nie and Jin, 2003	English, French, Italian, German	raw and round robin with tagged index terms	tagged better and robin slightly better than raw scores
Si et al., 2008	CLEF multi-8	Download, Transformation model	

translation quality, query specific and document specific features. The weights were linearly combined with ranking model scores, i.e., BM25, to compute ranking scores, which were then used to sort returned documents to create a single ranked list. The approach was used on NTCIR3, 4, and 5 collections and was compared with raw score, round robin, normalised-topk/l, 2 step merge and logistic regression. The proposed approach results were better than other previously proposed methods. Gao et al. (2009) approach LTR for multilingual retrieval as a new ranking problem – topics of the query are learnt from documents across languages and a joint relevance probability of documents is estimated. The approach is compared with other SVM based LTR algorithms (SVM-MAP and RSVM) and gives significantly better results. Table 3.4 provides a summary of LTR approaches for ranking multilingual results.

Several algorithms have been proposed for rank aggregation using LTR, such as Cranking (Lebanon and Lafferty, 2002), Markov Chain (Liu et al., 2007), LambdaMerge (Sheldon

TABLE 3.4: Listing of Work Using LTR for Merging Multilingual Search Results

Author and Year	**Approach**	**Evaluation**
Tsai, Wang, and Chen, 2008	FRANK	Better than 2-stage
Gao et al., 2009	Joint Probability Ranking	Better than SVM-MAP

et al., 2011) and LambdaMerge extension (Lee et al., 2015). LTR algorithms for multilingual/federated search results aggregation have performed better than the previously proposed methods. This indicates that as more research is done using LTR, new improvements to rank aggregation could be achieved. Previous work on search result ranking focused only on relevance. To meet the diverse information needs of users in different contexts, other factors such as understandability need to be considered. The current direction for LTR is to meet user information needs while also considering user subjective needs or behaviour. Such approaches attempt to rank documents by learning user preferences in terms of ranking of certain features to improve user satisfaction.

Learning to rank approaches that consider relevance and other criteria such as diversity (Gollapudi and Sharma, 2009), freshness (Dai, Shokouhi, and Davison, 2011; Dong et al., 2010) and efficiency (Wang, Lin, and Metzler, 2010) have been reported in literature. Such studies have been investigated in the context of multiple criteria optimisation and have used approaches for solving the multi-criteria optimisation problem (Adomavicius, Manouselis, and Kwon, 2011) : i) finding Pareto optimal solutions; (ii) aggregating multiple criteria, i.e, finding a weighted sum, and optimise the hybrid criterion, also called scalarization; iii) optimise the most important criterion and converting the other criteria to constraints; and (iv) optimising one criterion at a time and using the optimal solution as constraints.

Recency has been used to re-rank queries with temporal elements (Dai, Shokouhi, and Davison, 2011; Dong et al., 2010). Dai, Shokouhi, and Davison (Dai, Shokouhi, and Davison, 2011) used hybrid labels of documents by combining relevance and freshness scores computed using Harmonic mean. Divide and Conquer (DAC) ranking framework is used to learn how to rank documents. Results obtained using this approach were better than those that used demoting of relevance of a document with respect to time, proposed by Dong et al. (2010). Using hybrid labels linearly combines multiple criteria into a single relevance label, and therefore one single objective function is optimised. Gollapudi and Sharma (2009) used LTR to rank documents based on relevance and diversity using constraints. Constraints were specified as axioms that an objective function was required to satisfy to achieve diversification of results. A distance measure was defined to estimate pairwise similarity between any pair of documents from the returned results, and an objective function was trained to maximise the sum of relevance and dissimilarity.

Kang et al. (2012) compared the approach of aggregating labels or models for a multi-criteria problem in a vertical search scenario and found that model aggregation is better than label aggregation. Svore, Volkovs, and Burges (2011) investigated the problem of LTR with multiple objective functions using multiple graded labels from multiple sources, namely, clickthrough data and expert judgement labels. The authors extended the LambdaMART algorithm to solve the multiple objective ranking problem ,i.e., the ranking problem was transformed into an optimization problem of ranking using multiple relevance labels.

Similar to the work reported in this thesis is the task of ranking based on relevance and understandability of documents studied by Palotti et al. (2016). The authors used LambdaMART to rank health Web pages for topical relevance and understandability. Relevance features focused on query and document similarity scores and the readability features captured surface level properties of text. Likewise, with the goal of personalising search results based on user reading level, Collins-Thompson et al. (2011) re-ranked documents based on estimated reading level of the user. Topical relevance features are integrated with user reading level estimated scores and are used to rank documents. These studies are similar to our ranking approach, with the difference that the studies focused on understandability of documents (Palotti et al., 2016) and reading level of a user (Collins-Thompson et al., 2011) of text of the same language. Our work uses text based similarity features to estimate intelligibility of related languages to re-rank documents. Table 3.5 lists studies using LTR methods to meet several user needs or preferences.

Evaluation of retrieval systems is also following the same trend of meeting multiple user objectives with new criteria, such as understandability, usefulness and utility, being proposed. Relevance has been argued to be multi-dimensional with notions such as topicality, reliability, scope, novelty and understandability (Xu and Chen, 2006). However, the evaluation of retrieval systems with respect to relevance has been shown to be limited to topicality (Mizzaro, 1997; Cosijn and Ingwersen, 2000; Xu and Chen, 2006; Mao et al., 2016). Zuccon (2016) proposed understandability as an evaluation criteria integrated with topicality, i.e., understandability biased evaluation, based on the Gain Discount Framework proposed by Carterette (2011). This family of measures is based on an assumption that a relevant document is not useful if the searcher cannot understand the contents of the document. This assumption is important to the line of research presented in this paper, i.e., it is necessary to know the threshold of intelligibility a user is able to handle to have successful intercomprehension. The evaluation measure has since been proposed for evaluating consumer health search engines (Zuccon and Koopman, 2014).

LTR approaches that attempt to maximise multiple criteria, i.e., relevance and other criteria such as diversity and recency, have been explored to improve user search behaviour.

TABLE 3.5: List of Work Optimising Multiple Objective Using LTR

Author and Year	**Objectives**
Dai, Shokouhi, and Davison, 2011	Temporal and relevance
Gollapudi and Sharma, 2009	Diversity and relevance
Svore, Volkovs, and Burges, 2011	Multiple relevance label sources
Collins-Thompson et al., 2011	Reading level and relevance
Palotti et al., 2016	Readability and relevance

Only a few studies have been done in this area (see Table 3.5). These techniques have registered improved retrieval effectiveness, and it is evident that as more data becomes available, the number of criteria will increase and methods such as deep learning will advance the field. LTR has been used in our work with the same understanding that ranking to meet users' diverse features is possible with LTR by combining features that capture the attributes of the desired objective.

3.3 User Perspectives in Retrieval

Early MLIR focused on system oriented attributes such as query translation, indexing and ranking (Peters, Braschler, and Clough, 2012). However, recently there has been a growing interest to study multilingual search from the user perspective. Such work falls into three categories: (i) user centered design to find the right interaction design for multilingual search (Petrelli et al., 2006b; Petrelli, 2008); (ii) user focused comparative studies to understand user multilingual search behaviour to inform the building of adaptive, personalised multilingual search systems (Steichen and Freund, 2015); and (iii) multilingual Web search systems to support and understand needs of multilingual users (Capstick et al., 2000; Capstick et al., 1998).

3.3.1 User Centered Design for Multilingual Search Systems

Earlier work on multilingual search systems focused on supporting multilingual search for users with little knowledge of one of the languages being used . However, MULINEX, a Web search engine supporting English, German and French, was designed with the goal of supporting polyglots and understanding their information needs – this was uncommon in that era (Capstick et al., 2000; Capstick et al., 1998). One of the well documented research projects on interactive search using multiple languages and User Centered Design(UCD) is Clarity (Petrelli et al., 2006b; Petrelli, 2008). Clarity was a CLIR system that was built to

support users to find information using English and other European resource scarce languages such as Finnish, Swedish, Latvian and Lithuanian. Iterative design and evaluation were used in combination with qualitative and quantitative methods to build a CLIR system with an appropriate user interaction. The design of Clarity focused on interface features for query translation and feedback as well as the presentation of multilingual search results.

3.3.2 MLIR Interface Features

The question of the best multilingual search presentation layout has been explored. Three presentation layouts have been identified (Steichen and Freund, 2015); namely: single page merged results, tabbed single language results and panel results for specific language. Steichen and Freund (2015) studied five multilingual search interface designs: tabs, side-bars, panels, interleaving and a universal search interface. They found that although previous research focused on integrating results of different languages, this approach was the least favourable approach while the panels interface had the highest preference score. Similarly, Ling, Steichen, and Choulos (2018) conducted a comparative study on the same interfaces and found that listing results based on languages was the preferred interface and that users behaved differently when using each interface. These studies show that user interface features can give different user interaction behaviour. These results are important because the underlying algorithms used in search systems may be affected by the preferences of users, which may change the direction of research and technologies. Chu and Komlodi (2017) used UCD to design an interface for multilingual search system , and found that simplicity, visibility and customizability were some of the attributes users wanted for multilingual search interfaces.

3.3.3 User Interactions and Multilingual Search

Users may prefer results to be written in certain languages because of their varying language competencies or inherent subjective preference for certain languages. This entails that multilingual search results may be personalised to meet individual user preferences. Lowe and Steichen (2017) studied multilingual search behaviour in terms of language use, the effect of language proficiency, task and domain. Their results show that users use other languages that they speak when searching on the Web but that usage is influenced by proficiency and the task at hand or topic. Similar results were reported by Steichen et al. (2014); they concluded that multilingual users need to be presented with search results based on their preferences in terms of task, domain and language proficiency.

The shift in interest for MLIR from system oriented evaluation focusing on algorithms to user behaviour has brought in new insights on ranking, results presentation and interface

features for multilingual search results. However, there is still a gap in MLIR on user interactions and experiences for different user search contexts, which will inform the design of new algorithms and systems that meet user needs. Our work contributes to this knowledge by investigating the user preferences of the ranking of search results and their emotional responses as they interact with these results. Additionally, our work contributes to interface design of search systems for the retrieval of documents written in related languages.

3.3.4 Emotions

People experience emotions in all their interactions and therefore, it is unsurprising that several models for Interactive IR (IIR) have proposed affect or emotion as one of the factors affecting IR interactions (Ford, 2015; Saracevic, 1997; Savolainen, 2014). Moreover, previous studies have shown that emotions affect how people search and use information; the research community has focused on what emotions are experienced in search tasks and causes of the triggered emotions, the role of the experienced emotions on search behaviour (Lopatovska and Arapakis, 2011), and how prior emotional state of a searcher or emotiveness of information objects such as music and images can influence his/her choice (Poretski, Lanir, and Arazy, 2019).

Varying emotions are triggered in different search tasks – the emotional polarities experienced in search sessions have been found to correlate with positive attributes of the interaction, including successful search completion, easiness of tasks and readability of the document while negative emotions were associated with negative attributes such as frustration and difficulty to find answers (Lopatovska and Arapakis, 2011). The objective of our work is similar to the following studies: firstly, Arapakis, Jose, and Gray (2008) studied emotions associated with search tasks of varying difficulty and found that emotional polarity moved from the positive to the negative side of the emotion spectrum when task difficulty changed from low to high and secondly, Lopatovska and Mokros (2008) found that simplicity of writing style caused positive emotions with participants who were asked to rate retrieved documents.

In the linguistics community, the research objectives have been to identify linguistics and non-linguistics factors or features and statistical metrics that contribute to successful intercomprehension or predicting intelligibility of languages, but no studies exist investigating emotions in relation to intercomprehension. In the light of these findings and setting, it is vital to know what emotions are associated with users interacting with search results in which intercomprehension is expected.

TABLE 3.6: Listing of Stemming Techniques

Author and Year	**Approach**
Lovins, 1968	Suffix Stripping
Hafer and Weiss, 1974	Variety Successor
Porter, 1980	Rule based suffix stripping
Harman, 1991	Light weight suffix stripping
Krovetz, 1993	Dictionary based lemmatisation
Oard, Levow, and Cabezas, 2000	Character Co-Occurrence
Bacchin, Ferro, and Melucci, 2002	Graph-based
Mayfield and McNamee, 2003	n-grams
Bacchin, Ferro, and Melucci, 2005	Probability based stemming
Majumder et al., 2007	String similarity distance

3.4 Stemming

Stemming reduces morphological variants, i.e., words with the same stem, to a common form. Stemming was first proposed as a pre-processing step of index and search terms in IR systems to overcome term mismatches due to morphological variation of words. Morphological variation of the same concept/word may occur due to processes such as inflection, derivation, compounding and reduplication. Stemming is known to improve the effectiveness of retrieval (Hull, 1996). Stemming is known to boost recall because common forms will have more matches between query and document terms. However, stemming has at times been reported to harm IR performance with respect to recall - the improvement in retrieval seemed to have been offset by degradation in performance of some queries (Harman, 1991). This may be due to morphological variants being mapped to different common forms, which may harm precision or unrelated forms being mapped to the same form. Stemming has been reported to improve retrieval for short queries and documents (Hull, 1996). Stemming may be useful for under-resourced languages because many of the resources are short documents and at the same time more documents would be matched and recall can be improved.

The section first describes stemming techniques that have been proposed in literature. The section ends with a string distance similarity measure approach used in language independent stemming techniques.

3.4.1 Stemming Techniques

Stemming was the first effort to improve the performance of retrieval (Lovins, 1968). Stemming was first used to improve recall by mapping morphological variants to a single base form. Later, Krovetz (1993) proposed three motivational perspectives for stemming: (i) query expansion (ii) clustering, and (iii) normalising of query terms to concepts. Krovetz's

proposed perspectives of stemming are based on the principle that stemming would reduce a word to a linguistic root to preserve the meaning of words. Unfortunately, many stemming approaches reduce morphological variants to an arbitrary stem, which is not linguistically motivated, and thus only satisfy the first function.

Several stemming approaches have been proposed in literature, namely: table lookup, successor variety, rule-based, n-grams and statistical stemmers (Frakes, 1992) and graph based. The most basic stemmer uses a table or machine-readable dictionary to store indexed terms and their corresponding stems or a dictionary with stems as part of the entries. Search terms or terms to be indexed are stemmed by looking up their possible stem from the table or dictionary. Similar to this approach, Krovetz (1993) proposed a rule-based stemmer that looks up the word before stripping off affixes to obtain linguistic roots or stems. Unfortunately, data such as machine-readable dictionaries are unavailable for many languages and storing and retrieving pre-computed values is computationally expensive.

The successor variety approach proposed by Hafer and Weiss (1974) for suffix stripping is a language-independent stemming approach that does not require linguistic rules and resources. The approach is graph-based and attempts to estimate morpheme boundaries based on a distribution of text from a large corpus of text. The successor variety of a given sequence of characters is the number of characters that come after it in a given corpus. This number decreases as the string gets longer until a morpheme boundary is reached. Hereafter, the value of the successor variety increases. Unfortunately, this approach only works well with unidirectional affix stripping, i.e. , suffix removal, and cannot work with words with both prefixes and suffixes.

Rule-based stemmers use rules specifically for a given language to remove prefixes and affixes. These rules are written to remove specific affixes from a word – the approach is flexible as one may choose to remove inflectional affixes only, i.e., light stemmer (Harman, 1991), or derivational affixes as well, i.e., a strict stemmer (Porter, 1980). Several approaches have been proposed for rule-based stemmers. Most stemmers use the longest iterative match technique (Lovins, 1968) to remove affixes or the longest sub-string using a set of language-specific rules.

Rule-based algorithms have been used in several languages, including Bantu languages (Malumba, Moukangwe, and Suleman, 2015). The most widely implemented rule-based algorithm is the Porter algorithm. Porter's algorithm uses a set of preconditions and rules. Conditions are divided into three categories, namely: conditions on the stem, conditions on the suffix, and conditions on the rules. The most prominent condition is based on the element measure m, which is computed based on alternate vowel-consonant sequences. Measure m is given as: $[C](VC)^m[V]$. Different types of rules are applied on a word in steps based on the value of m, i.e., only one rule in each step is selected in order to remove the

longest possible affix. The English language has five steps rules. Unfortunately, Porter's algorithm was designed for suffix stripping and usually fails to accommodate languages with prefixes and infixes. Rule-based algorithms have produced better results than other alternative approaches proposed in the literature so far. Moreover, rule-based stemmers require well-crafted rules that require morphology knowledge of the language. This knowledge may not be available for languages that have not been well documented. The proposed work uses rules based on the morphological structure of words in many Bantu languages (Nurse, 2001; Meeussen, 1967). However, the affixes to be removed are generated using a statistical approach.

Statistical stemmers are language independent and do not require language-specific knowledge – they learn the possible affixes or morpheme boundaries from raw text collection. Several approaches for statistical stemmers have been proposed, notably, approaches that use (1) morphological paradigms to cluster morphological variants together and compute the stem of the words (Bhat, 2013; Majumder et al., 2007; Adamson and Boreham, 1974) and (2) probabilistic models to estimate morpheme boundary or stem in a given word (Bacchin, Ferro, and Melucci, 2005). Statistical stemmers are able to handle corpus specific idiosyncrasies but are unable to handle cases not seen in the corpus. Therefore, most of the approaches need thresholds and parameters (Bhat, 2013; Majumder et al., 2007; Adamson and Boreham, 1974) that are corpus dependent and require to be re-computed when the collection has changed. Our proposed approach uses rules and affixes that have been learnt from a corpus of a given size using a morphological clustering method. The approach renders that it can work on any Bantu language text that does not completely divert from the proposed linguistic structure.

Finally, n-grams – repeated consecutive sequences of n characters – have been proposed by Mayfield and McNamee (2003) as an alternative to stemming with comparable performance to rule-based approaches. n-gram indexing and matching have been tested on several Indo-European languages such as English, Spanish, and French (McNamee, Nicholas, and Mayfield, 2009) and Indian languages such as Marathi, Bengali and Hindi (McNamee, Nicholas, and Mayfield, 2009; Dolamic and Savoy, 2010). 4-grams have been found to provide relatively better results than other n-gram sizes. Also, results seem to indicate that improvement in results is correlated with the morphological complexity of the language. Unfortunately, due to the repetition of characters, the size of the index grows faster with n-gram indexing and there is no direct correspondence between any given word and the strings kept in the index. A similar approach that uses most occurring word final n-grams as suffixes to derive suffixation rules was proposed by Oard, Levow, and Cabezas (2000). The approach was applied on 50,000 words and used with Linguistica – unsupervised word segmentation tool using Minimum Length Description (MLD) (Goldsmith, 2001) to overcome the problem of corpus dependency of statistical methods. The approach registered

better average precision over unstemmed words and using Linguistica only (Oard, Levow, and Cabezas, 2000).

Several approaches have been proposed for stemming, from rule-based stemming to corpus-based stemming. Table 3.6 lists studies on stemming using different techniques. Unfortunately, many of the stemming approaches have been proposed for Indo-European languages, which are highly suffixal. On the contrary, Bantu languages are prefixal with some derivational processes that are suffixal. Therefore, more research needs to be done to design stemming approaches that both accommodate the word structure prevalent in Bantu languages as well as the complex processes such as reduplication, and are friendly to less described languages.

3.4.2 String Distance Similarity Measures

The majority of systems that use unsupervised methods to learn the morphology of natural languages consist of two components (Hammarström and Borin, 2011), namely: a bootstrapping heuristic that comes up with candidate strings of morphemes and an explicit model that has either of the following formulations: (1) what constitutes an adequate morphology for a set of data; or (2) an objective function that must be optimized, given a corpus of data, in order to find the correct morphological analysis. One of the approaches is to group morphologically related words using methods that use the orthographic word, e.g., string distance measures.

String distance similarity measures have been used to estimate morphological relatedness of word pairs using shared consecutive sequences of characters or n-grams, as well as by applying directly on individual characters of words. The work of Adamson and Boreham (1974) is the first to report the use of string distance similarity measures in stemming – dice coefficient based distance on bi-grams was used to calculate the similarity between pairs of words of document titles (Adamson and Boreham, 1974). String similarity measures have also been used for other tasks such as determining words with the same root (Roeck and Al-Fares, 2000), query matching, and indexing for morphologically related languages (Majumder, Mitra, and Pal, 2008; Majumder et al., 2007; Bhat, 2013; Snajder and Basic, 2009).

Statistical stemmers based on string similarity measures group morphologically related words into morphological classes, i.e., words sharing the same stem, and select a stem for each class. Morphological classes are created using a clustering algorithm, such as Hierarchical Agglomerative Clustering (HAC) with string distance as a distance metric for splitting clusters. Several classes of string distance measures have been proposed for different linguistic typology (Majumder et al., 2007). Table 3.7 shows the listing of string distance similarity measures for clustering morphologically related words in stemming algorithms or related tasks.

TABLE 3.7: Listing of Studies using String Distance for Morphological Clustering

Author and Year	**String Distance Measure**	**Evaluation**
Adamson and Boreham, 1974	Dice	successful clustering
Jacquemin, 1997	Simple truncation	successful clustering
Gaussier, 1999	Simple truncation	successful clustering
Roeck and Al-Fares, 2000	Jaccard	Better clusters
Yarowsky and Wicentowski, 2000	Edit distance	Successful word alignment
Schone and Jurafsky, 2000	Edit Distance	Successful
Baroni, Matiasek, and Trost, 2002	Edit Distance	Successful
Hu et al., 2005	Edit distance	Successful
Majumder et al., 2007	D_1, D_2, D_3, D_4 and Dice	D_3
Snajder and Basic, 2009	D_3, D_4, Dice and Levenshtein	D_4
Bhat, 2013	D_2, D_4	D_2

Different approaches including simple truncation methods, traditional edit distance, dice coefficient and a set of D_x distances have been proposed in literature for similarity distance measures to estimate morphological relatedness of words. Simple truncation estimates the similarity between two words by comparing the first k characters of words (Jacquemin, 1997; Gaussier, 1999). Formally, two words w_1 and w_2 of a given language are p similar if

$$truc(w_1, p) \equiv truc(w_2, p) \tag{3.1}$$

where the first $trunc(w, k)$ consists of first k characters of w and there is no q such that $q > p$ and $trunc(w_1, q) \equiv trunc(w_2, q)$

Edit distance measure between a pair of words is the minimum number of insertion, deletion, and substitution operations (Levenshtein, 1966) required to change one of the strings to the form of the other. Mathematically, Levenshtein distance between two strings X,Y of length $|X|$ and $|Y|$ is given by $lev_{x,y}(|X|, |Y|)$

$$lev_{x,y}(i,j) = \begin{cases} max(i,j), & \text{if } min(i,j) = 0 \\ min \begin{cases} lev_{x,y}(i-1,1) + 1 \\ lev_{x,y}(i,j-1) + 1 \\ lev_{x,y}(i-1,j-1) + 1(a_i \neq b_j) \end{cases} & \text{otherwise} \end{cases} \tag{3.2}$$

Wagner and Fischer (1974) later defined the edit distance problem as a problem of computing a series of edit operations with minimum cost to transform one string to another. This version of edit distance has been used in studies for morpheme discovery, segmentation and word productions (Hu et al., 2005). Hu et al. (2005) used the String Edit Distance (SED) to

perform alignment of words to discover templates which are transformed into Finite State Transducers (FST) and used to parse Swahili words.

Dice coefficient uses character n-grams and calculates the number of distinct common n-grams between two strings and divides by the sum of n-grams for the two strings (Adamson and Boreham, 1974).

$$Dice_n(X,Y) = \frac{2C}{X+Y} \tag{3.3}$$

Snajder and Basic (2009) generalised the Dice coefficient for effective clustering as follows:

$$Dice_n = 1 - \frac{2C}{X+Y} \tag{3.4}$$

Jaccard similarity coefficient compares members of two sets to see which members are shared and which are distinct (Snajder and Basic, 2009).

$$J(X,Y) = |X \cap Y| / |X \cup Y| \tag{3.5}$$

Majumder et al. (2007) proposed a set of distances for estimating the similarity between two words, primarily focusing on suffix stripping. The approach compares a pair of strings by comparing characters at each position. Similar characters in the same position score a zero and non-matching characters score a 1. Mathematically, given a pair of strings $X = x_0, x_1 ... x_n$ and $Y = y_0, y_1, ..., y_n$, the string distance measures D_x is defined using a Boolean function p_i:

$$p_i = \begin{cases} 0 & if x_i = y_i (0 \leq i \leq min(n, n\prime)) \\ 1 & otherwise \end{cases} \tag{3.6}$$

D_1 matches full word pairs but the subsequent metrics matchs up to the first mismatch. These distances aim to reward long matches on the left side and are targeted on suffixation: D_1 penalises all following character positions after first match, D_2 rewards long matching prefixes, D_3 rewards long matching prefixes and penalises long non matching prefixes and D_4 measure penalises long non-matching suffixes. These measures have been defined as follows:

$$D_1(X,Y) = \sum_{n=0}^{n} \frac{1}{2^i} p_i \tag{3.7}$$

$$D_2(X,Y) = \frac{1}{m} \sum_{n=0}^{n} \frac{1}{2^{i-m}} if m > 0, otherwise \infty \tag{3.8}$$

$$D_3(X,Y) = \frac{n-m+1}{m} \sum_{n=0}^{n} \frac{1}{2^{i-m}} if m > 0, otherwise \infty \tag{3.9}$$

$$D_4(X,Y) = \frac{n-m+1}{n+1} \sum_{n=0}^{n} \frac{1}{2^{i-m}} if m > 0, otherwise \infty \tag{3.10}$$

D_x measures have been evaluated extensively on a number of languages: English (Adamson and Boreham, 1974; Yarowsky and Wicentowski, 2000), Arabic (Roeck and Al-Fares, 2000), English, French and Marathi (Majumder, Mitra, and Pal, 2008), Hungarian and Czech (Majumder et al., 2007), Croatian (Snajder and Basic, 2009)and Kannada (Bhat, 2013). These studies have compared the performance of these metrics to language specific stemmers like Porter's (Porter, 1997) and other word segmentation and stemming techniques. Evaluation results reported so far suggest that Dice coefficient and Levenshtein distance are the least effective string distance similarity measures (Majumder, Mitra, and Pal, 2008; Majumder et al., 2007; Bhat, 2013; Snajder and Basic, 2009). However, stemmers created using Dice and Levenshtein metrics appear to perform better than simple truncation techniques and no stemming. D_x measures seem to perform better than other metrics (Majumder, Mitra, and Pal, 2008; Majumder et al., 2007; Bhat, 2013; Snajder and Basic, 2009). This is not surprising as this set of measures (Bhat, 2013) are designed to extract suffixes from words of single slot morphology languages and these results are from tests on such languages so far. Despite the success of D_x measures, there has not been a single D_x family metric that has given the best results in all evaluations done.

Using string distance similarity measures to create morphological clusters is attractive because it is easy to implement for any language and requires a small corpus of naturally occurring words. The results the approach has generated are promising, i.e., improved retrieval effectiveness compared with no stemming and sometimes comparable results with more sophisticated techniques – those that require huge amounts of data or hand-coded rules. Most importantly, the approach has been tested on languages of different families, which are largely under-described languages. This provides new opportunities for bootstrapping language resources for low-density languages (Hammarström and Borin, 2011). Equally important, some of the languages used so far in the evaluations use different alphabets than the Latin alphabet (e.g., Indic languages) (Bhat, 2013; Majumder et al., 2007). Accordingly, the approach is usable across languages with little changes. Although using string distance similarity measures have been applied in several languages, there is still some work that needs to be done in this area to fully understand their potential. Firstly, in terms of evaluation, the approach has been evaluated on languages that are predominantly suffixal (suffix stripping only required). This is also evident in the type of distance measures proposed so far. It would be beneficial for similar evaluations to be done on other languages that have different types of morphology. For example, Bantu languages provide a different scenario as they have a complex morphological system that goes beyond one-slot morphology (at least one affix to add or remove). Also, these studies would provide insight into whether new metrics are required for other languages with a different typology as well as

what heuristics to use in these metrics. Furthermore, evaluations need to provide similar multiple evaluation metrics in order to compare results across studies.

Secondly, most of the evaluations are done to identify words with similar stems for the purpose of stemming in retrieval systems without analysing the quality of clusters or identifying the possible stem in the cluster. For instance, most of the studies used the central word in the cluster as a stem (Majumder et al., 2007). Using this approach, it is unclear how ad-hoc queries can be managed in a retrieval scenario, i.e., how the stem of the incoming queries can be found. Consequently, the quality of clusters created has not been of much importance in the approach – only that smaller clusters generate stronger stemmers and larger clusters create weak stemmers. This ignores the fact that clusters generated this way may be processed further to produce better results (morpheme segmentation techniques learn patterns in word-formation to identify morpheme boundaries).

Finally, the use of string distance similarity measures in estimating morphology has intrinsic weaknesses: the use of character sequences to estimate both form and semantic relatedness may lead to clustering words that are morphological and semantically different. In this context, approaches that incorporate semantics such as deep learning may be appropriate but requires a lot of data, which is unavailable for most languages. Moreover, affixes not appearing in the corpus may not be recognised as belonging to the language, i.e., the method is inextensible to what has not been seen in the corpus. Unsupervised methods for semantic clustering are applied to raw text to come up with semantic groups. Semantics-based approaches use local context to come up with words that appear in the same context in the corpus, i.e., words that appear in the same context have the same meaning. Several studies have shown some improvements when semantics are combined with orthographic similarity techniques. Yarowsky and Wicentowski (2000) combined three measures based on semantic (cosine similarity), frequency and orthographic similarity (edit distance) to perform analysis and observed significant performance improvement. Similarly, Schone and Jurafsky (2000) used orthography(edit distance), semantics (latent semantic analysis), and syntactic distributions to improve performance. Baroni, Matiasek, and Trost (2002) used edit distance and a semantic measure based on the proximity that words appear to each other in a corpus and registered some improvements in the accuracy of segmentation.

Recent advances in machine learning have made semantic analysis easier. Üstün and Can (2016) used word2vec word embeddings to come up with semantic vectors from raw text. Cosine similarity was used to estimate the orthographic similarity of word pairs. Other studies have used only the syntactic context to induce morphological clusters successfully (Can and Manandhar, 2010). Unfortunately, comparative studies for these different approaches have not yet extensively been done but several studies have combined these different approaches to improve results.

3.5 Summary

Information retrieval is an active research field with several theories and algorithms being proposed, from document ranking algorithms that use a combination of documents and query features, to estimate document relevance to complex document ranking function learners that use user(s) search history and features beyond relevance. Prior research on retrieval for related languages focused on matching queries with documents across languages, more specifically using vocabulary similarity without any translation in the retrieval pipeline. Further, the prior evaluation focused on whether such approaches would retrieve relevant documents – comparable results with respect to other approaches such as dictionary-based query translation were reported. We have discussed related work in retrieval covering these areas including merging and ranking search results, user perspectives, and interaction issues surrounding multilingual search results. In the context of merging and ranking search results, supervised methods have shown better performance than other methods using heuristics such as rank positions. The major challenge for this research direction for resource scarce languages is lack of data. Studies focusing on understanding user search behaviour and their interactions with multilingual search systems have not been done widely, including search involving related languages. Therefore, there are still gaps in understanding user behaviour interacting with multilingual search results. Stemming may improve results of resource-scarce languages as there would be several documents matching. We have discussed several techniques used in stemming. Results of stemming showed mixed results – some stemming techniques worked better on certain languages and corpus. However, the n-gram approach has provided consistent results.

Chapter 4

Test Collection Development

System oriented evaluation is the most widely adopted IR evaluation approach, used mainly to assess the effectiveness of new retrieval models or techniques. Test collections and evaluation measures are commonly used in system oriented evaluation. We adopted test collection based evaluation to assess our ranking and stemming approaches. The retrieval effectiveness evaluation of the proposed stemming approaches used the Cranfield approach (Cleverdon, 1991) – our test collection consisted of a set of documents, topics and relevance judgements (qrels) (Sanderson, 2010) and a set of appropriate metrics for evaluation. Our ranking study used a dataset that consisted of query and document features such as relevance similarity scores, query and language intelligibility features and relevance labels for query and document pairs.

The chapter discusses the procedure for creating a test collection used to evaluate our proposed methods. Firstly, we discuss the process of collecting documents, extracting text from the collected documents, structuring of new documents created from the extracted text and adding metadata to the new documents. Secondly, we describe the procedure that was used to formulate, translate and format topics that were used as statements of information needs for the collection. Thirdly, we describe how relevance assessments were done, from recruiting judges to the procedure used to assess documents. Lastly, we provide an analysis of intelligibility features used in the study: a description of intelligibility features that were considered in our research and how the features were calculated.

4.1 Corpus

A test collection is made up of a set of documents, topics and relevance judgements. The documents or corpus for the collection represent the type of documents likely to be found in a real world setting (Sanderson, 2010). We created a new test collection for two reasons. Firstly, our work involved RSLs, i.e., languages that have limited or no language resources and tools like corpora, machine readable dictionaries and machine translation readily available. As such, there are no test collections available for the studied languages. Secondly, the study required specific features, i.e., user preferences of documents based on the L_1 and

intelligibility features. We created a small test collection to enable experimentation with the proposed techniques. Several tasks were completed, including documents gathering, query formulation and translation, judgements as well as feature extraction and engineering.

4.1.1 Documents Gathering

Due to the limited amount of information readily available in digital format and copyright constraints, we obtained information from two media houses in the form of newspaper articles and news bulletins. Topics in these documents include current and development news, health, and religious topics. The radio news bulletins are written in Chichewa and English, while the newspaper articles are written in Chichewa and Citumbuka. The rest of the documents that form part of the collection and are written in Chichewa, Citumbuka, Citonga, Cinyanja and Cisena were obtained from the Web.

Documents were converted from such formats as PDF and HTML (Web pages) into text files. The text file documents were then cleaned by removing irrelevant information, such as header or footer text. Metadata, including title of the document, language of the content, document identifier and source, were extracted from the original documents. Missing data fields were added to documents that lacked such information. The text files and metadata were used to create XML documents following the TREC and CLEF style. Each XML file contains the following fields: DOCument IDentifier (DOCID), DOCument Number (DOCNO), title of document (TITLE), Description of the content (DESCRIPTION), language of the text (LANGUAGE), origin or source of text (SOURCE) and the text content of the document (TEXT) (see Figure 4.1).

Zodiak Radio News

Zodiak collection consisted of files of news items broadcasted on Zodiak FM, a radio station in Malawi, in the year 2015. The original collection contained news files written in English and Chichewa, consisting of both full bulletins and news headlines, which were read at different times of the day in either English or Chichewa, or both. Each bulletin was written in its own file and the files were provided in MS Word format, i.e., .doc or .docx. Since our corpus used Chichewa files only, English bulletins and mixed language bulletins were ignored.

Documents were created from collected MS Word files written in Chichewa. The procedure for creating documents consisted of the following steps: (i) file pre-processing, (ii) extracting stories from bulletins, and (iii) cleaning and formatting of documents.

- **Pre-processing of files:** In the first step, we created folders for each day and moved all news files of a particular day to its folder. We then grabbed news files from three

```
<add>
<doc>
<field name="DOCID">ZK182015-17</field>
<field name="DOCNO">ZK182015-17</field>
<field name="TITLE">Ebola</field>
<field name="DESCRIPTION"> Zotsatira za kafukufuku wa katemera
wa matenda a Ebola zikusonyeza kuti mankhwalawa akhoza kuteteza
ku matendawa</field>
<field name="LANGUAGE">Chichewa</field>
<field name="SOURCE">Zodiak</field>
<field name="TEXT">Zotsatira za kafukufuku wa katemera wa
matenda a Ebola zikusonyeza kuti mankhwalawa akhoza kuteteza ku
matendawa.  Pomwe matenda a Ebola anavuta m'mayiko a ku zambwe
mu Africa panalibe mankhwala oteteza anthu kumatendawa.  Bungwe
la za umoyo padziko lonse lapansi World Health Organization
lati mankhwalawa adzetsa chiyembekezo ndipo akhoza kusintha
zinthu.</field>
</doc>
</add>
```

FIGURE 4.1: Example Document from Zodiak News Bulletin in Chichewa

bulletins in each day and put them in their own folders: 7:30 am, 1 pm and 7 pm – bulletins identified to have Chichewa text only. The selected files were converted to .txt from .docx (unoconv and antiword) or .doc (antiword).

- **Extracting stories:** The text files were parsed to extract stories from each file. We developed some heuristics to get stories – page headers and story titles were used as heuristics to break a text file into stories.

- **Cleaning of stories:** The extracted stories were cleaned to remove empty lines, blank stories and other unnecessary elements.

  - Each story had some repeated words and other elements that were removed (cue in, insert, cue out) or other characters and words that were not part of the story such as page labelling from footer and news break announcement.
  - Due to some news being repeated during the course of the day, some of the stories appeared in multiple files. Cosine similarity of each story against all stories of that day was computed and any duplicate stories were removed. Any stories with cosine similarity score of 0.8 and above were treated as duplicates.

- **Formatting stories to documents:** Each story was formatted into CLEF (XML) format, but extended for Solr indexing by introducing *add* and *doc* tags. More metadata necessary for the research was added, such as language of the document and source of the document.

The Zodiak collection consisted of very short stories mainly because they are radio news items that are usually short. There are still some repeated stories mainly because some of the stories were repeated over long periods, from days to weeks.

Fuko Newspaper

Fuko is a local newspaper in Malawi, published fortnightly in two languages, namely, Chichewa and Citumbuka. Fuko covers current, political and developmental topics. Twelve (12) publications in each of the two languages were made available by the publisher. Each article in the newspaper was used to create a document for the collection. The following procedure was used to extract data from the documents:

- **Pre-processing:** Fuko volumes were obtained from the publisher in PDF format and the files were first converted to images using pdftoppm. Tesseract, an OCR program, was used to convert the image files to text files.
- **Extracting stories:** Using end of article heuristics (a combination of 'I', followed by a dot, blank lines and capital letters for title of new story), articles were extracted from the text files. Metadata such as title, description, DOCID, DOCNO were also extracted from the text files. DOCID and DOCNO were generated based on date of publication, title was the title of the article, and description was generated by combining article title and the first two sentences of the article.
- **Formatting:** the extracted articles were formatted using the format in Figure 4.1.
- **Cleaning:** The stories were checked manually for bad characters such as wrongly encoded characters.

Wikipedia

Wikipedia provides opportunities for access to information that is available in multiple languages. Two Wikipedia sites for Citumbuka [1] and Chichewa [2] were used to generate documents for the collection – done in two steps. First, the urls of the pages were extracted: this was done using Beautiful Soup, where a GET request was used to access the page that listed

[1] https://tum.wikipedia.org/wiki/Main_Page
[2] https://ny.wikipedia.org/wiki/Tsamba_Lalikulu

all the pages available in that language. The page entries (names of the Wikipedia page entries) were extracted and used to send new requests. Second, new GET requests were sent and the body content of the returned pages were extracted. The documents were formatted using the style in Figure 4.1.

Health Text

The health corpus was extracted from a publicly available Chichewa translation of 'Where there is no Doctor' by Umoyo Trust, titled 'Pamene Palibe Dokotala' [3]. The book sections or topics were extracted from the table of contents chapter. In each chapter, section titles were used as heuristics to identify the beginning and end of a new section. Each section in the book became a document. Firstly, the PDF documents were changed to text files. The start of the next section was used as the end of the current section. Metadata was also generated for each document. The generated documents were written to file and formatted using the style in Figure 4.1. The created documents were edited manually to correct text errors introduced by the conversion program.

Religious Text

Beautiful Soup was used to send GET requests to specific pages and used to extract the text content of the pages. The text extracted from Web pages included religious writings, such as the Bible and related content. The text from each page was formatted into a text document. Religious content covers all the five languages that are being investigated in the study.

Table 4.1 shows the statistics of the corpus, including number of words and documents arranged by language and source. Zodiak documents dominated the Chichewa corpus but the documents are shorter compared to documents from other sources. Documents from Wikipedia are the shortest, while most of the religious documents are longer than documents from other sources. Documents from Fuko are well written, and most of the stories have translation equivalents.

4.1.2 Text Properties

The quality of the corpus was evaluated, particularly by checking if the corpus follows the distribution of words in a natural language text, which may affect different retrieval aspects such as indexing. We investigated Zipf's linguistics law of word frequency. Given a corpus, Zipfian analysis is used to model distribution of words, i.e., Zipf's law states that the frequency of a word is inversely proportional to its rank in the word frequency table for a collection of words of natural language utterances (Powers, 1998). Thus for written text,

[3]https://hesperian.org/books-and-resources/resources-in-chichewa/

TABLE 4.1: Corpus Statistics. Chichewa corpus has the most number of documents.

Source	No of Docs	Distinct Words	Word Total
Chichewa			
Zodiak	6,519	35,433	421,032
Fuko	210	10,370	41,607
Wiki	674	16,305	72,735
Health	369	19,292	81,992
Religious	1,609	63,767	512,422
Total	9,380	114,369	1,092,518
Citumbuka			
Fuko	203	8,559	406,053
Wiki	688	5,907	38,144
Religious	1,367	48,124	19,792
Total	2,258	58,390	459,789
Citonga			
Total	1,367	48,124	297,793
Cisena			
Total	449	19,161	159,660
Cinyanja			
Total	173	12,796	59,935
Corpus Total	**13,627**	**252,840**	**2,069,695**

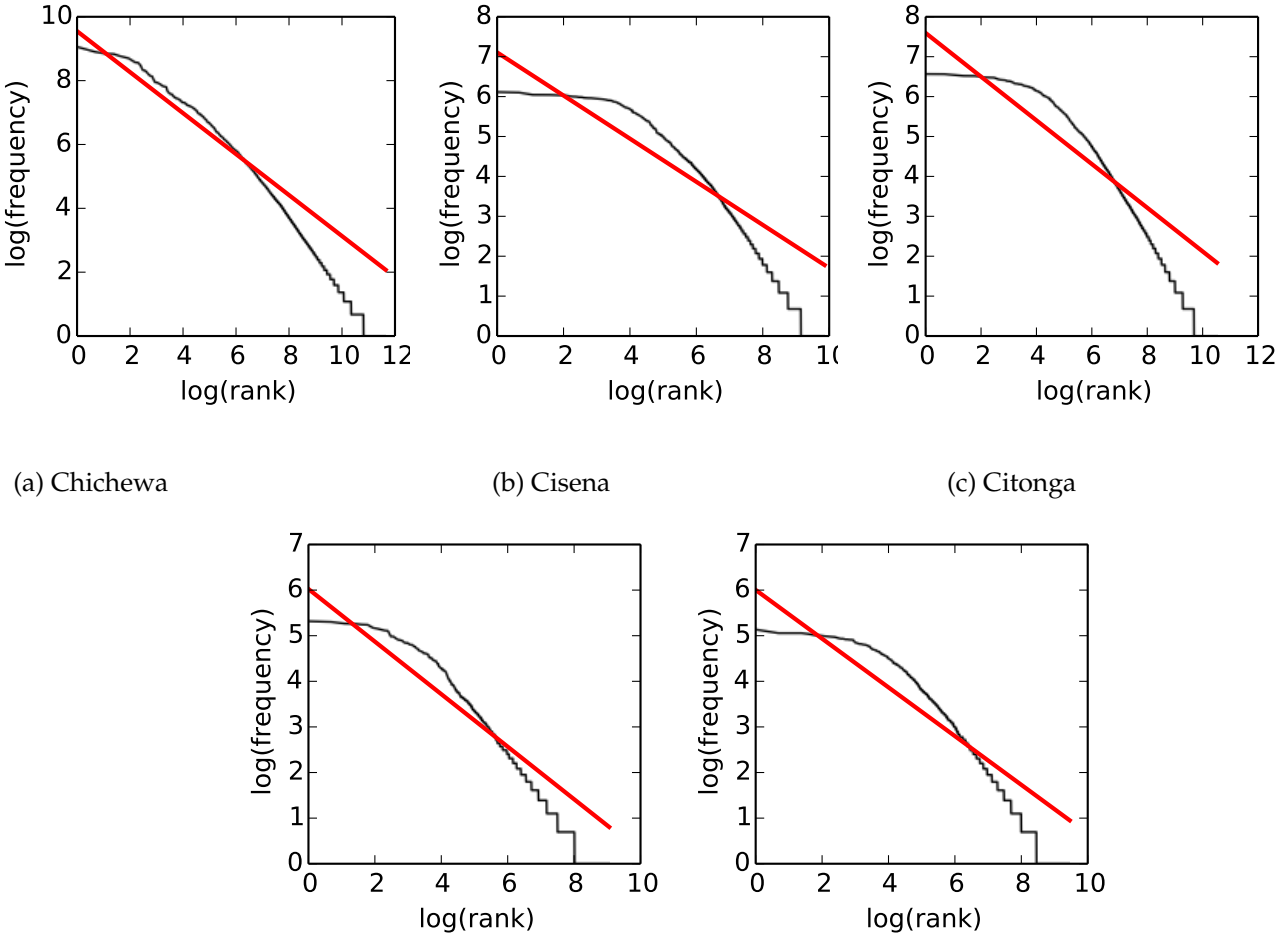
rank ordered distribution of word frequencies follows the power law:

$$f(r) = \frac{C}{r^{\alpha}} \tag{4.1}$$

where r is the rank for every word in the corpus, C is the normalising constant and $\alpha \approx 1$ is a free parameter for specifying the degree of skewness. We computed the frequencies of words in the corpus and ranked the words according to their frequency of appearance. The results are plotted on a graph of log of frequencies ($Log_e Fr$) against the log of the ranks ($Log_e R$). The log-log plot for rank and frequency are expected to be approximately linear. The plots for Cisena, Citonga, Chichewa, Citumbuka and Cinyanja in Figure 4.2 are more closer to the fitted line and thus follow the word frequency distribution observed in many natural languages.

4.2 Topics

Topics are information needs statements used to evaluate different techniques of retrieval in system centered evaluation. Following TREC/CLEF style format of topics, a topic consists of an identifier, title, description and narration. The title of the topic is usually used



(a) Chichewa (b) Cisena (c) Citonga

(d) Citumbuka (e) Zambian Cinyanja

FIGURE 4.2: Zipfian plots for the (from top left to right and bottom left to right) Chichewa, Cisena, Citonga, Citumbuka and Zambia Nyanja Corpora respectively. The red line shows the fit of Zipfian power law on the corpus data.

as a query for the topic. The description gives an explanation of the information need providing contextual information. The narration gives the explanation as well as description of what documents are relevant for that particular topic. Assessors use topics to evaluate retrieved documents. We recruited assessors to formulate, translate topics and judge documents against the topics.

We distributed a call for participants via email to all university students at the University of Cape Town and through social media outlets for Zambia and Malawi student societies. We used self reported language knowledge to recruit the assessors. Topics were created by monolingual assessors who spoke one of the languages of interest, i.e., Chichewa, Citumbuka and Cinyanja. Topics included representative information needs in all the three languages being considered. We did not include topics for Citonga and Cisena due to the problems with finding resources to compute some of the required features. Citonga documents are still used in the re-ranking experiments. Assessors were given themes drawn from the sources of documents to be used as a starting point to browse the collection for dominant topics that can be explored further to formulate topics. A custom Web based interface was created for interacting with the documents.

4.2.1 Topic Formulation

Five assessors were recruited to formulate queries. Each assessor was given a task sheet, which had a list of themes for the topics to be created and a template for creating the topics, including an example topic. The task sheet also had instructions on how to browse the system to find documents, how to formulate topics, where to write the topics and how to write the translation in English.

The assessors came up with topics after browsing the collection using a Web based retrieval system running Solr with BM25 scoring function on the back-end. Each assessor reviewed the top one hundred documents for relevant documents. A topic with at least five seen relevant documents was admitted to the list of topics. After deciding to include the topic in the collection, the assessor translated the topic to English. Each assessor submitted at least twenty topics.

4.2.2 Topic Translation

Assessors translated topics formulated in the initial phase. Each assessor translated at least fifty (50) topics to their mother tongue using an English equivalent of the topic. A translation was accepted only after another assessor had confirmed the translations. Each assessor was paid R130 for a session of three hours.

During each session, assessors were given a Word Processing document with a table for each topic. The assessors then entered their translations in a column for their language.

```
<?xml version="1.0" encoding="UTF-8"?>
<topic>
<identifier>17</identifier>
<title lang="en">Disease Prevention</title>
<title lang="ny">Kudziteteza ku Matenda</title>
<title lang="nya">Kupewa kuli matenda</title>
<title lang="tum">Kujivikilira ku Matenda</title>
<description lang="en">What should I do to prevent
diseases?</description>
<description lang="ny">ndingatani kuti ndidziteteze ku
matenda?</description>
<description lang="nya">Nnikuchita chani kuti ninkale alikupewa
kuli matenda</description>
<description lang="tum">Ningachita uli kuti nijivikilire ku
matenda</description>
<narrative lang="en">The document should provide ways of disease
prevention</narrative>
<narrative lang="ny">Tsambali likuyenera kupereka njira
zodzitetezera ku matenda</narrative>
<narrative lang="nya">Zolemba ziyenekela kupeleka njila ya
kupewa kuli matenda.</narrative>
<narrative lang="tum">Hamba likwenera kupereka nthowa
zakujivikilira ku matenda</narrative>
<query lang="en">disease prevention</query>
<query lang="ny">Kudziteteza ku matenda</query>
<query lang="nya">Kupewa kuli matenda</query>
<query lang="tum">kujivikilira ku matenda</query>
</topic>
</xml>
```

FIGURE 4.3: Sample topic from the collection. Each field has a translated equivalent in each of the three languages and English.

After translating their topics, the assessors submitted their translated topics. Thereafter, another assessor evaluated their translations and made changes where necessary. In total, one hundred and twenty nine (129) topics were translated to two of the other languages depending on the original language. Finally, the topics were formatted as TREC and CLEF topics (Figure 4.3).

4.3 User Relevance Judgements

An invitation to judge documents was distributed via email to all UCT students and through social media outlets for Zambia and Malawi student societies. Relevance judgements were done by monolingual users, more specifically L_1 speakers of the topic language, i.e., participants were asked to judge documents written in their mother tongue as well as related languages. Prospective participants responded by signing up using a link shared in the call. Six (6) assessors were recruited and assigned to tasks based on their L_1. Before the assessors started the task, the researcher run a tutorial to the assessors to show and explain the procedure to complete the task. Thereafter, the assessors signed a consent form. Each assessor judged thirteen (13) topics in each session of two hours. Assessors were presented with one hundred documents at a time to assign a relevance grade on the scale of 0 to 3: 0 Not relevant, 1 Marginally relevant, 2 Fairly relevant and 3 Highly relevant. The full description of the scale for relevance is given in Table 5.2.

4.3.1 Relevance Assessment Procedure

The assessment task started with two pilot sessions. In the pilot sessions, assessors judged fifty top ranked documents, but it was realised that many of the relevant documents were missed. Also, assessors judged each document for topical relevance and intelligibility. However, it was realised that documents were being judged for understandability in terms of the vocabulary used in the documents. Therefore, documents written in the same language had a wide range of intelligibility scores. In the actual sessions, intelligibility was not included in the relevance judgement task; assessors were trained and asked to judge documents according to the usefulness or utility of the document. Leaving out intelligibility affects how the dataset can be used – for example, ranking experiments would only optimise one objective of document usefulness or utility (the assumption is that intelligibility is implicitly incorporated in the user judgement. Additionally, assessors judged up to 100 hundred documents for each topic.

Relevance assessments were done using a Web interface on top of Solr. Topics were stored in a database and divided into tasks. Each task had thirteen queries and assessors were assigned to tasks. Each assessor was given login details to be used with the system.

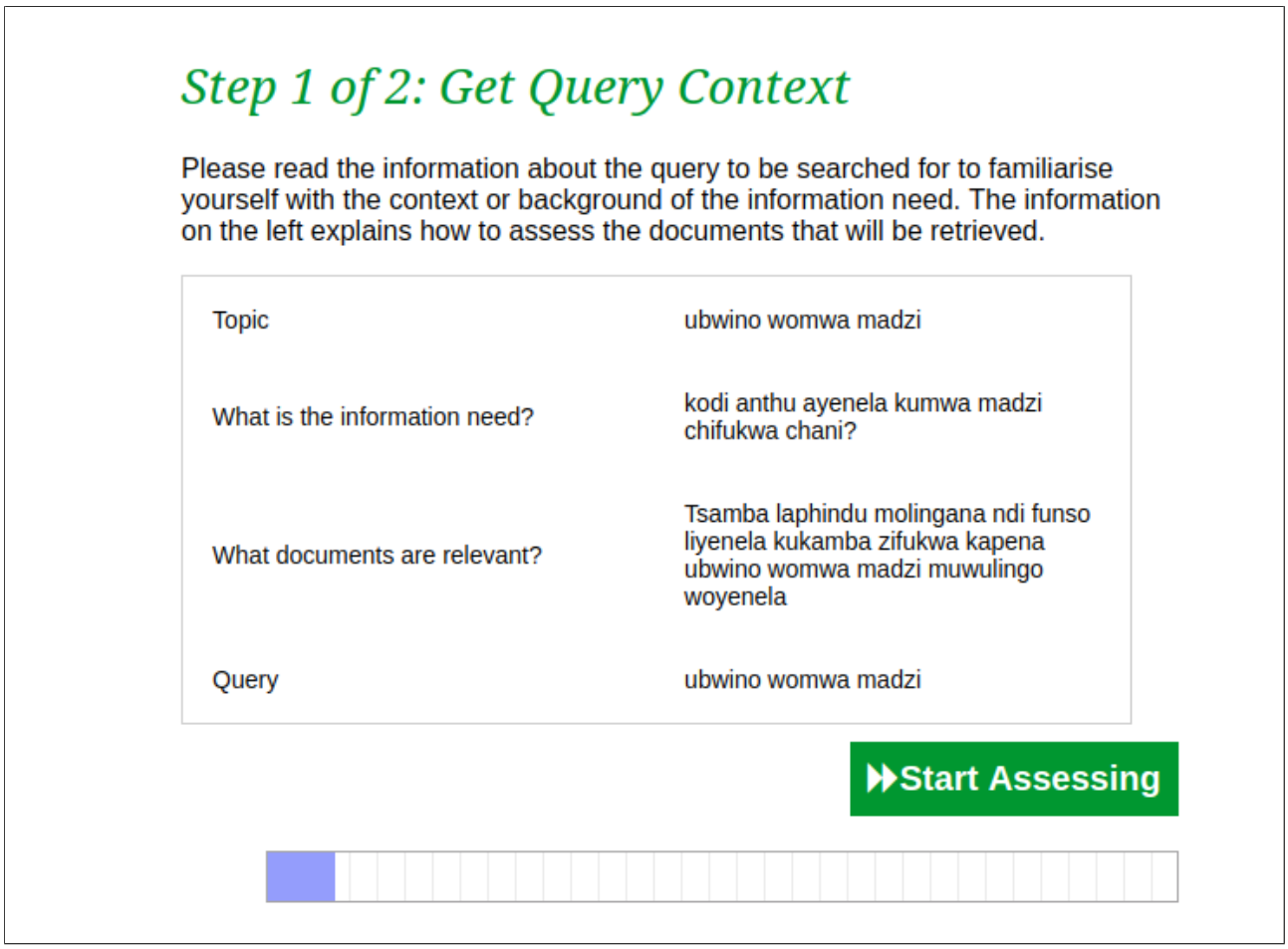


FIGURE 4.4: Example for topic to be assessed

After logging in, the system gave the assessors information about tasks to be completed. In each task, topics were run one after another until all 13 topics were judged. Assessors did the judgements in sessions in the researcher's laboratory. After proceeding to a task, the system showed the assessor the topic to be assessed and the status to completion of the session. One hundred (100) documents were retrieved for each topic. Assessors were paid R130 for each session. Figure 4.5 shows the interface for judging documents and showing topics.

4.3.2 Relevance Assessments System Set-Up

Four different Solr scoring functions were used to increase the diversity of documents retrieved, namely: (i) default BM25 on words, (ii) default BM25 with 3 and 4 character n-grams, (iii) probabilistic based model using Divergence from Randomness (DFR) (Amati and Van Rijsbergen, 2002) and (iv) language modelling based retrieval model using Bayesian

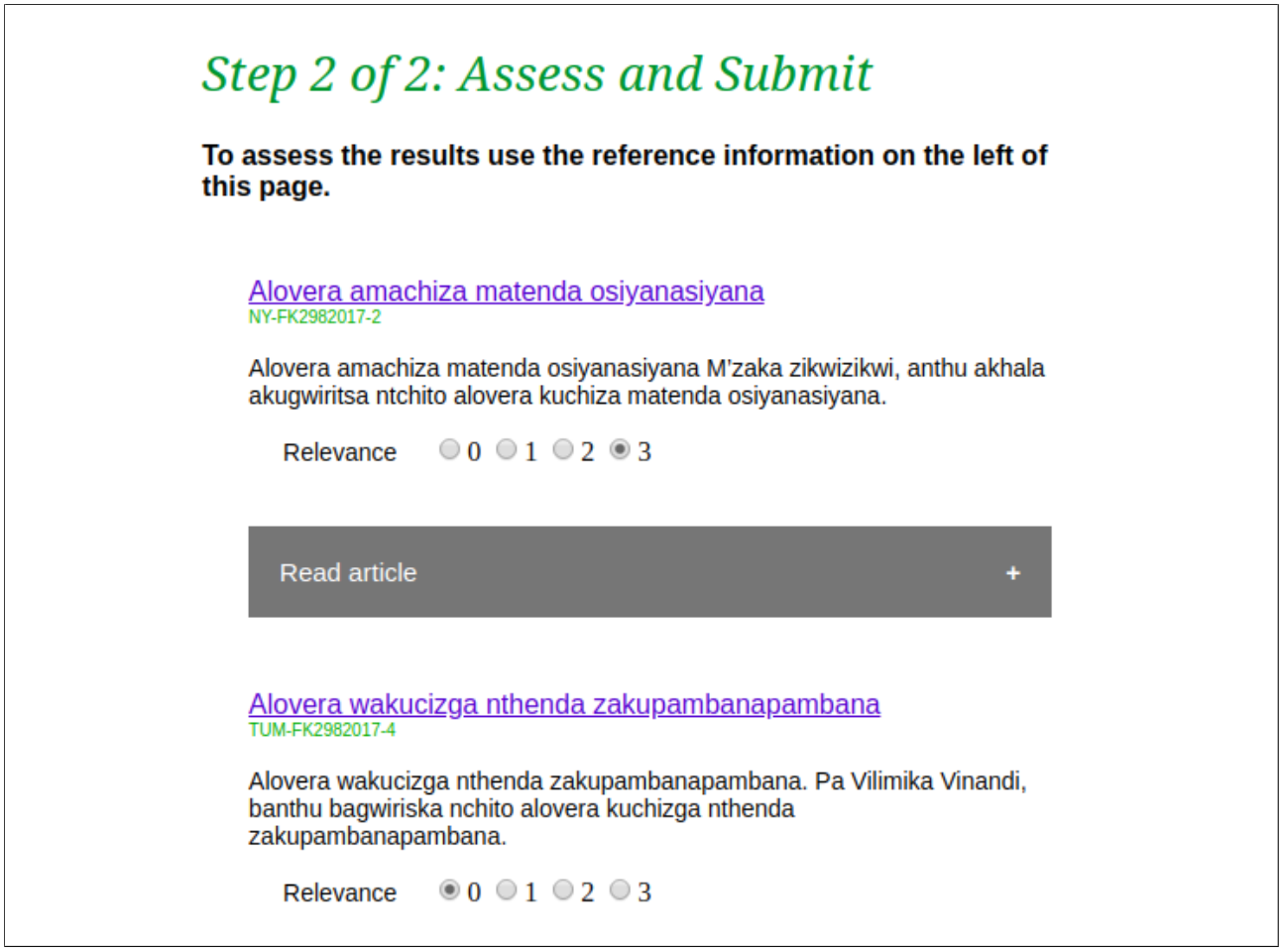


FIGURE 4.5: Interface for judging documents showing documents being judged.

smoothing with Dirichlet priors (Zhai and Lafferty, 2001) with $\mu = 2000$. BM25 used standard parameter values – $k_1 = 1.2$, $k_3 = 7$ and $b = 0.75$. All the four approaches used centralised indexing for all of the corpus languages, i.e., each index had the same documents but indexed differently based on the model used (see Figure 4.6). Only one hundred top documents were presented to the assessor.

Documents retrieved by each retrieval system were merged and ranked by the aggregating node,i.e., the node that received the query, sent it to the other three nodes, received search results, removed duplicates and sorted the results of all four nodes based on similarity scores. A score in this case is a measure of how well a document matches a query using a particular scoring function such as BM25. Each query sent to Solr consisted of all three versions of the query for assessors to judge as many documents as possible written in any of the three languages. The assessors were presented with ranked documents in decreasing order of estimated topical relevance. However, the assessors were told that the order of the documents was not important and that all documents had to be judged uniformly.

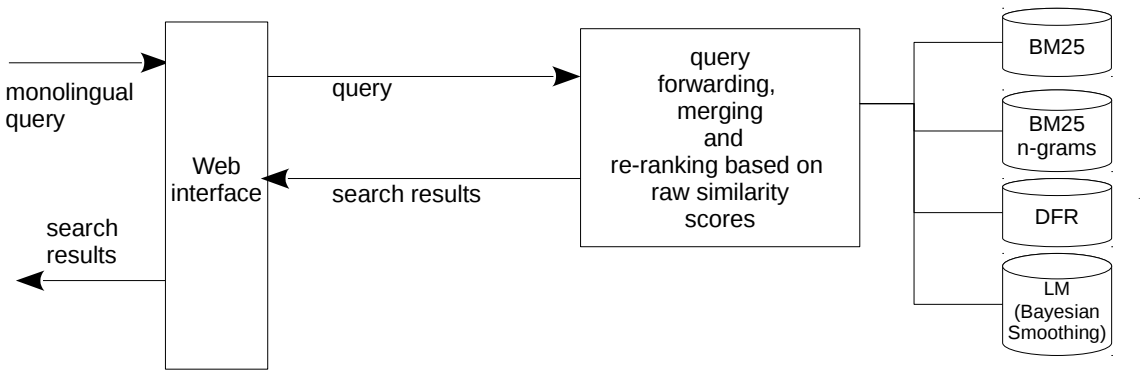


FIGURE 4.6: Architecture of the search system used for relevance judgements.

4.4 TREC Style Dataset

The complete dataset consists of the corpus, 129 topics in three languages, and relevance assessments of the topics, with each topic judged to a depth of 100. In our work, two types of studies were investigated and each required a different set of data in terms of features. Our stemming study used the Cranfield approach for evaluating retrieval based on effectiveness, and this required document query features and relevance judgements.

We formatted the judgements and runs using TREC Style. This consists of qrels provided by assessors, including document and query features. Relevance judgements or qrels followed the TREC format: `query identifier`, `iteration`, `document number or identifier` and `relevance score or judgement`. This dataset was used with evaluation runs also formatted based on TREC style: `query identifier`,`iteration`,`document number or identifier`, `rank`, `similarity value` and `run identifier`. We formatted the dataset in this way to take advantage of the tools available for calculating evaluation metrics and for reuse in other experiments that may need such a data set.

4.5 LTR Dataset

LTR is a supervised learning task and therefore, requires a labelled dataset (Li, 2011), i.e., both a training set and a test set. LTR is an ongoing research area and several algorithms have been proposed. To objectively compare algorithms and to speed up the progress in IR research, by removing the task of developing data sets, several evaluation benchmark datasets have been created, e.g., LETOR and OHSMED (Qin and Liu, 2013; Qin et al., 2010). Most of the benchmark datasets available are provided by companies specialising in search such as Microsoft, Yahoo and Yandex (see Table 4.2).

Such datasets follow a single format or standard: a dataset consists of several queries and their associated documents with their features describing the relationship between the

TABLE 4.2: List of Available Benchmark Datasets for LTR

Name	Dataset	Queries	Documents	Features
LETOR 4.0	MQ2007	1692	69623	46
LETOR 4.0	MQ2008	784	15211	46
LETOR 3.0	OHSUMED	106	16140	45
LETOR 3.0	Gov03td	50	49058	64
LETOR 3.0	Gov03np	150	148657	64
LETOR 3.0	Gov03hp	150	147606	64
LETOR 3.0	Gov04td	75	74146	64
LETOR 3.0	Gov04np	75	73834	64
LETOR 3.0	Gov04hp	75	74409	64
Yahoo	Yahoo1	29921	709877	519
Yahoo	Yahoo2	6330	172870	595
MSLR	MSLR10k	10000	1200192	136
WCL2R		79	5200	29
Yandex		21701	97290	245
IMAT				

query and the document. For each training example, there is a relevance label y_j^i, followed by a query id, then a list of features and may end with other auxiliary information such as document IDs. The relevance label may be Boolean, i.e., 0 or 1, indicating that the document is relevant or irrelevant to the query or multilevel graded indicating different levels of relevance such as *not relevant, slightly relevant, relevant, useful and perfectly relevant*. Feature vectors are specified using a feature ID and their values. The dataset, i.e. features (most datasets do not include Web pages or documents – only the values of the features are released in text format) are released in a text format or text file. Figure 4.7 illustrates the format of entries used in the datasets and followed in our study.

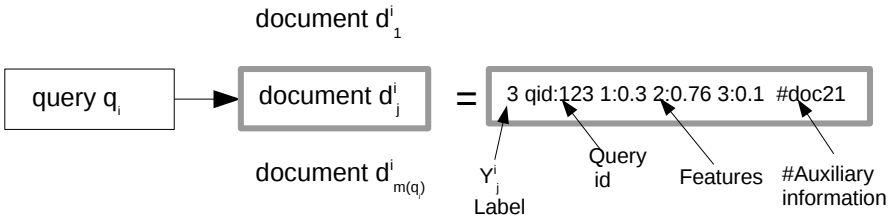


FIGURE 4.7: Learning Instance Format. Each instance is associated with a query and document features.

The LTR benchmark datasets developed and released so far have focused on monolingual retrieval and do not contain features that are required for the current study. Specifically, current available datasets do not have ranking preferences necessary for the current research

– relevance judgements based on topicality and intelligibility of related languages as required by the experiments and therefore, a new dataset is created. The dataset is labelled with relevance scores with respect to queries in both monolingual setting and multilingual retrieval.

4.5.1 Dataset Features

The proposed LTR approach requires features drawn from the documents with respect to the query. Selection of features included in the dataset is based on the principles used in LTR datasets such as LETOR (Qin et al., 2010), i.e. using features that have been used in previous IR studies. Features are grouped into three categories: query–document features, document features and document – language features. Query-document features are calculated based on the extent to which the query matches the document. These are calculated for each document and query pair, and estimate the topicality or aboutness of a document with respect to a particular query. Document features are features that specify document attributes. Document–language features use language intelligibility features. These features are calculated using equivalent documents obtained from the Web or in the public domain such as religious or Wikipedia pages.

4.5.2 Topical Relevance Features

For query–document features, relevance similarity scores between the document and query were used. BM25 is used as a feature for representing the matching degree of relevance between query terms and document terms (Robertson and Walker, 1994). We used standard values for BM25 parameters, i.e., $k_1 = 1.2$, $k_2 = 7$ and $b = 0.75$. We also included TF, IDF and TF-IDF scores as features. Additionally, language modelling with Jelinek-Mercer Smoothing scores were employed. The description of these concepts has been given in Chapter 2. We calculated the features using untranslated queries based on 3-gram tokens and unprocessed tokens from the document – words as they appear in the corpus. Table 4.3 lists the topical relevance features.

4.5.3 Intelligibility Features

Reading text in an unfamiliar language may cause some frustration for the user as they attempt to understand the written text. This may considerably affect the relevance of the document, e.g., a document perfectly matching the information need in a not very related language may be less useful than a marginally relevant document in the language of the user or in a closely related language, subject to how successful intercomprehension can be performed by the user. To this end, the ranking function needs to be able to balance between

TABLE 4.3: Features used to represent query-document similarity and document features.

Feature	**Description**
	Relevance Features
TF	Term Frequency in title
TF	Term Frequency in body
TF	Term frequency in body and title
IDF	Inverse Document Frequency in title
IDF	Inverse Document Frequency in body
IDF	Inverse Document Frequency in body and title
BM25	BM25 in title
BM25	BM25 in body
BM25	BM25 in body and title
Normalised BM25	normalised BM25 score based on body and title text
TF–IDF	TF-IDF in title
TF–IDF	TF–IDF in body
TF–IDF	TF–IDF in body and title
LM	Query likelihood language model with Jelinek-Mercer Smoothing
	Document Features
$\|D\|$	Length of title
$\|D\|$	Length of body
$\|D\|$	Length of body and title.

relevance and intelligibility of the returned content. Therefore, reading intercomprehension should be quantified to describe how monolingual speakers of one language can understand text written in a related language. Intercomprehension is determined by linguistic costs such as linguistic distance measured at different levels of language description (lexis, orthographic, morphological distance and morphosyntax) and extra-linguistics costs such as foreign language learning attitude, exposure, age, prior linguistic background and other socio-linguistics factors. There has been mixed results on the correlation of intelligibility and extra-linguistic costs (Stenger et al., 2017). Moreover, it is not clear how these elements can be quantified and used in intelligibility measures. Linguistic determinants of intelligibility usually measure how two languages are related based on shared vocabulary, visual representation correspondences, i.e, orthography and more recently, word recognition cognitive measures based on information theory – conditional entropy and surprisal (Stenger et al., 2017), and language models (Fischer, Vreeken, and Klakow, 2017; Gamallo, Pichel, and Alegria, 2017). Table 4.4 lists the linguistic features used in our experimentation while Table 4.5 lists the extra-linguistic features.

Linguistic Distance Based Measures

Lexical distance is based on the number of shared cognates between two languages on a common set of words such as the Swadesh list (common lexicalised or universal concepts) (Heeringa et al., 2013). The assumption is that cognates act as links between two languages and the more shared vocabulary, the more likely readers can understand text written in the other language. Orthography is the primary interface between a reader and language in a reading intercomprehension scenario. Methods based on orthography, i.e., orthographic distance using string similarity measures such as Levenshtein distance, have been widely used to measure mutual intelligibility between language pairs (Heeringa et al., 2013; Stenger et al., 2017). Levenshtein distance is a string similarity measure calculated as the number of operations required to transform one string into another through character insertions, deletions, and substitutions. Each of these operations are given weights. However, weights are not used for the basic implementation of the algorithm. Levenshtein distance calculated based on text from two languages gives similar values regardless of the direction of the relationship, i.e, symmetric relationship (Levenshtein, 1966). We provided a description of how to calculate the Levenshtein distance between two strings in Chapter 3.

Intelligibility relationship between two languages is usually asymmetrical – a language L_i can be intelligible to speakers of a particular language L_j but L_j may not necessarily be intelligible to speakers of L_i to the same degree. LD values are symmetric since the values are calculated directly based on the two words without considering direction of the relationship. The advantage of using LD is that it has been widely used in similar tasks for several languages and the results have been consistent.

Complexity based Measures

More recent research has focused on modelling the cognitive processes in intercomprehension scenarios using surprisal theory (Stenger et al., 2017). Surprisal theory attempts to account for the mental processes when humans process sentences. The theory stipulates that reading time of a word has a linear relationship to how predictable a word is in a given context or sentence (Hale, 2001; Goodkind and Bicknell, 2018). The less surprising a word is in a given context, the less time a reader takes to read it. The processing cost of a word w_i, is the surprisal of a word given its position i, calculated as :

$$surprisal(w_i) = -log_2(p(i)), \tag{4.2}$$

where $p(i)$ is the probability of getting w_i as the next word given the sentence's previous words $w_1, w_2, w_3..., w_{i-1}$. The theory has been adapted to a word level analysis by considering the cost of having the next character of a word, known as Character Adaptation Surprisal (CAS) (Stenger et al., 2017):

$$surprisal(L1 = c1|L2 = c2) = -log_2 P(L1 = c1|L2 = c2) \tag{4.3}$$

L1 – native language, c1 – character of L1
Ł2 – stimulus language, c2 – character of L2

Word Adaptation Surprisal (WAS) is calculated by adding CAS of characters in a word. The approach has been shown to have results that are comparable with Levenshtein (Stenger et al., 2017). Additionally, adaptation surprisals between two languages are different depending on the direction of the relationship, i.e., asymmetric. The metric provides a more realistic relationship between languages and attempts to model the cognitive load that readers of text in unknown languages have while attempting to understand the content.

Conditional Entropy (CE) measures the uncertainty in a random variable when another is known. In intercomprehension, it is used to estimate the complexity of a mapping from a non-native language to a native language. CE is asymmetric and has been shown to model the difficulty of the adaptation process between cognates of the known language and the target language (Jens et al., 2007). Therefore, low entropy between word pairs from two languages signifies high intelligibility between the languages. CE of a pair of words is calculated as follows:

$$CE(X|Y) = \sum_{x \in X, y \in Y} P(x, y) log_2 P(x|y) \tag{4.4}$$

CE is usually measured at phonetic level for intelligibility in spoken language but has also been applied at orthographic level for reading intelligibility. In order to have stable values of CE, it is estimated that 800 word pairs are required (Jens et al., 2007).

Distributional Similarity Metrics

The objective of language modelling is to capture and exploit properties and regularities in a language, i.e., to learn a function that captures statistical characteristics of the distribution of sequences of words and assigns probabilities of the next word given previous context or words. Language models based on words from text are widely used but have proven to be limited in scenarios where there is word variation due to morphologically related forms, spelling variation and word relatedness due to language similarity. Additionally, training models require a lot of data, which is expensive for many resource scarce languages. This

is a challenge for the languages under consideration because Bantu languages are morphologically rich and have limited resources. Moreover, the study involves related languages. Low level language modelling based on smaller units such as sub–words or characters may provide better results.

The perplexity of n-gram models extracted from text corpora measures how well a probability distribution or probability model fits a sample. Character n-grams encode both lexical as well as morphological information. Mathematically, perplexity (PP) is the inverse probability of the test text given the mode. A low perplexity indicates the probability distribution is good at predicting the sample. The perplexity (PP) of a collection of text in relation to a language model LM of a related language, CH ($CH = ch_1, ch_2, \ldots, ch_n$) is calculated as (Gamallo, Pichel, and Alegria, 2017):

$$PP(CH, LM) = \sqrt{\prod_{i}^{n} \frac{1}{P(ch_i|ch_1^{i-1})}}, \tag{4.5}$$

where n-gram probabilities is given as:

$$PP(ch_n, ch_1^{n-1}) = \frac{C(ch_1^{n-1}ch_n)}{C(ch_1^{n-1})} \tag{4.6}$$

A perplexity-based distance between two languages is calculated by comparing the n-grams of a text in one language with the n-gram model trained for the other language. The perplexity of the test set text in L_2 given the language model of L_1 is used to find the perplexity (Nakov et al., 2017; Gamallo, Pichel, and Alegria, 2017):

$$Distance_{PP}(L_1, L_2) = PP(CH_{L2}, LM_{L1}) \tag{4.7}$$

Kullback-Leibler (KL) Divergence measures the difference between two probability distributions (Kullback, 1959). KL Divergence can be used as measure of how the distribution of tokens in one language, e.g., n-grams, is different from the distribution in another language. KL divergence is calculated as:

$$D_{KL}(P\|Q) = \sum_{i=1}^{t} p_i log \frac{p_i}{q_i} \tag{4.8}$$

where P and Q are two probability distributions (Kullback, 1959). KL divergence is asymmetric and takes a positive or zero values. The two probability distributions are identical when the KL divergence is equal to zero. KL divergence is used to measure the similarity of the distributions of tokens between pairs of corpora to perform tasks such as document classification. We calculated the KL divergence between the language models of the tri-grams

of every pair of languages being investigated.

Jensen – Shannon divergence is a measure of similarity between two probability distributions (Lin, 2006). The measure is a symmetric and takes finite values derived from KL divergence $D(P\|Q)$. It is mathematically defined by:

$$JSD(pq) = \frac{1}{2}D(P\|M) + \frac{1}{2}D(Q\|M) \tag{4.9}$$

with $M = \frac{1}{2}(P + Q)$, (Lin, 2006). We calculated JSD distance – the square root of JSD divergence on tri-grams of corpora of every pair of languages.

Syntactic Measures

Related languages also share syntactic features. Gooskens and Swarte used three syntactic features that analyse word variation for parallel text to estimate syntactical similarities of two related languages (Gooskens and Swarte, 2017). The measures are movement measure, indel measure and tri-gram measure.

Movement measure calculates the number of words that move when literally translating text from one language to another. The lesser the number of equivalent words that change position in a sentence, the closer the syntax of the two languages. It is calculated by counting the number of words that move when translating a sentence from one language to another closely related language.

Indel measure counts the number of inserted or deleted words when translating text from one language to another language. The lesser the number of words removed or deleted when literally translating two sentences, the more similar the syntax of the two languages.

Tri-gram correlation measures the similarity of the frequencies of word tri-grams between corpora of two languages. The more similar the frequencies of shared tri-grams, the more similar the languages. We calculated tri-gram correlation for both words and n-grams.

Extra-linguistic Features

Extra-linguistic features are based on factors that are non linguistic in nature but affect intelligibility of languages, such as frequency of contact and learning. We designed an online experiment for L_1 speakers of our focus languages by adopting methods for measuring intelligibility proposed by Bayley et al. (2013). We collected extra-linguistic features from the participants and intelligibility scores by asking L_1 speakers to translate text written in the languages. We recruited participants through calls made by e-mails and social media messages to the general public as well as mailing lists of university students in Malawi and Zambia. Participants were recruited into the study without any competency test on the languages to be investigated. One hundred and four participants took part in the study.

TABLE 4.4: List of intelligibility features calculated based on corpus and list of words

Feature	**Description**
	Intelligibility Features
Levenshtein Distance (LD)	Average distance measuring the number of operations required to transform a cognate in one language to a word in another language
Levenshtein Distance (LD) – stem	Average distance measuring the number of operations required to transform a stem cognate in one language to a stem in another language
Conditional Entropy	The uncertainty or difficulty of mapping a word in a non-native language to a word in a native language
Perplexity Distance	Measures how well a probability distribution of n-grams from a corpus predicts a model
Cosine Similarity	measures the similarity between two vectors of an inner product space of tri-grams of two documents
Surprisal	measure of uncertainty in a word being transformed to a cognate
Kullback-Leibler divergence	measure of how the language models based on character n-grams of one language is different from that of another language – asymmetric
Jensen–Shannon divergence	measure of similarity between the distribution of n-grams between two languages – symmetric
Lexical distance	the percentage of the number of words that are not cognates for any two given languages.
Document Language	the language of the document
query language	the language of the query
	Syntactic Features
Movement measure	number of words that are moved when translating a sentence from one language to another closely related language
Indel	number of inserted or deleted words when translating text from one language to another language.
Tri-gram	correlation of the number of frequencies of word tri-grams in a corpus of two languages.

TABLE 4.5: List of extra-linguistic Features

Feature	Description
Age	Age of participant. Values were transformed into four classes, i.e., 1 to 4.
Gender	Gender of participant. The data was transformed to binary values, i.e., 1 for male and 0 for female
Qualification	Highest level of academic qualification. The values were transformed to integers, i.e., 1 to 4.
Contact	Representing whether the participant had contact with the language. Binary values represented whether the participant had previous contact with the language or not.
Attitude	Participant perception of the beauty of the language. Scores were on a scale of 1 to 5.
Familiarity	Represented the contact frequency of the language. Scores were on a scale of 0 to 4.
Learning	Represented whether participant had learnt the language before participating in the study.

The experiment had three sections. The first section consisted of general demographic questions such as age, gender, qualification, languages learnt so far and all the places that they had ever stayed. The second section consisted of opinion testing in which participants self-reported their proficiency of all the five languages in reading, writing, listening and speaking tasks. Five levels were used: 0 none, 1 novice, 2 intermediate, 3 confident and 4 excellent. Participants were also asked about their attitude towards each of the languages, i.e., how ugly or beautiful a language sounded on a scale of 1 to 5. The third section consisted of text comprehension tests for each of the languages and each participant completed all five tasks in each language. Participants were given a paragraph of about 200 words to read and provide a summary in their L_1. After completing each task, participants self-reported their level of comprehension for the text on a scale of 0 to 4 namely: 0 understanding nothing, 1 for recognising some words, 2 for understanding some of the sentences and being able to gauge what was being said, 3 for understanding almost everything but missing some of the words and 4 for understanding everything completely.

Online experiments provided two types of data: first, subjective data about the language experience and other socio-linguistic factors associated with our languages of interest, and second, intelligibility scores given their subjective language prior knowledge. The latter were combined with linguistic attributes to create features used in the feature engineering phase and the former as a target variable. The final set of features were derived from the computed linguistic features and subjective attributes provided in the online experiments.

Feature Extraction

Linguistic features were derived from word-lists and parallel corpora. The word list was a list based on the Swadesh list. Swadesh list is a list of about 207 concepts, which are deemed to be universal and culturally independent. Initially we obtained the English version of Swadesh list and asked two L_1 speakers of each of the five languages to translate the list to their L_1. These word lists were used to compute lexical distance, (word, stem and affix) Levenshtein distance, surprisal and conditional entropy. We used Incompy [4] (Mosbach et al., 2019) to calculate Conditional Entropy, Surprisal and Levenshtein using the created wordlists.

Some of the features required parallel corpora to be calculated. We used religious text from five articles – the articles were originally written in English but translated by the publisher to Citonga (4,389 tokens), Chichewa (3,895 tokens), Cinyanja (4,124 tokens) and Citumbuka (4,137 tokens). We used this corpus to calculate features, namely, perplexity distance, Kullback-Leibler Divergence, Jensen-Shannon and cosine similarity.

We also aligned five paragraphs of text from Fuko newspaper extracts to investigate positional correspondences of grammatical features in the investigated language pairs. This text was originally written or obtained from the publisher in Chichewa. We recruited translators to translate the paragraphs to their L_1. This corpus was used to calculate Indel, movement measure and tri-gram correlation.

4.5.4 Data Cleaning

We corrected spellings and merged data provided the participants. For example, highest educational level choices were merged – O level was merged with A level, professional certificate and diploma were merged to tertiary qualification. Additionally, a few participants did not complete their gender and educational level. Some of the participants did not complete some of the sections of the tests and average values from another study with the same languages were used to fill the missing values.

4.5.5 Data Transformation

Categorical variables were converted to numerical representation as some of the algorithms expect numerical values only. Ordinal variables such as age(18 – 25, 25 – 35, 35 – 50 and above 50) and education level (O-level, Diploma, Bachelors, Masters and PhD) were encoded using numerical encoding. Attitude of a participant towards each language was encoded with integer values associated with participant choices made during the experiment. Familiarity and exposure were derived from participants' responses on how frequently they

[4] https://github.com/uds-lsv/incompy/blob/master/utils.py

come into contact with the language in terms of listening, reading, speaking and writing the language. For example, participants answered the following question for the exposure – How often do you hear/speak or read Cinyanja? The choices to select included never, once in week, twice in a week, once in a day and that they use it everyday. Each familiarity choice was associated with a numerical value from 1 to 5.

Binary values were used for studying the language in school. Those who studied the language were assigned the value of 1 and those who did not the value of 0. Gender (male and female) was encoded with binary values, namely, 1 for male and 0 for female. Geographical location for the first ten years and last ten years were coded as binary values for different places in Malawi, feature instances were clustered into regions and, for places outside Malawi, country codes were used. Regions with contact with the language were given 1 and those with no contact were assigned 0.

Each training instance consisted of feature values for both linguistic and extra-linguistic features' values and an intelligibility score. Due to the asymmetric nature of intelligibility, each language combination, i.e., (L_1 Chichewa and target Citumbuka or L_1 Citumbuka and target language Chichewa) had an entry for each participant, hence there were 20 entries for each participant. Therefore, for each training instance, we added two more features: L_1 and target language. We derived intelligibility scores from self-reported text comprehension scores after participants completed a text comprehension test for each language. The test comprehension scores were divided into five categories from 0 to 4, or none to excellent.

4.5.6 Data Exploration

We first wanted to know the distribution of the data and the association relationship of the features against one another and intelligibility score – these aspects may affect the performance of the features in learning tasks. We explored our data by visualising it in terms of frequency distribution using histogram, and correlation of features against the target variable. We used Pearson correlation to measure the similarity between a feature and the target outcome or class label, i.e., an intelligibility score. Figure 4.8 shows the frequency distribution of values of each feature. The distribution of values for linguistic features are not even in each feature and not similar across features – features may be capturing different aspects of the languages. The distribution of learning a language and contact seems to be similar. Intelligibility score is not evenly distributed: score 4 has the most count, followed with score 3.

Figure 4.9 shows the correlation of each pair of features using Pearson Correlation Coefficient. There is high correlation between the target feature $SCORE$ with all features except for age (AGE), gender (GEN), and qualification (QUA), and syntactic features ($INDEL$, TRI, MOV). Linguistic features have high correlation amongst themselves. Age, gender

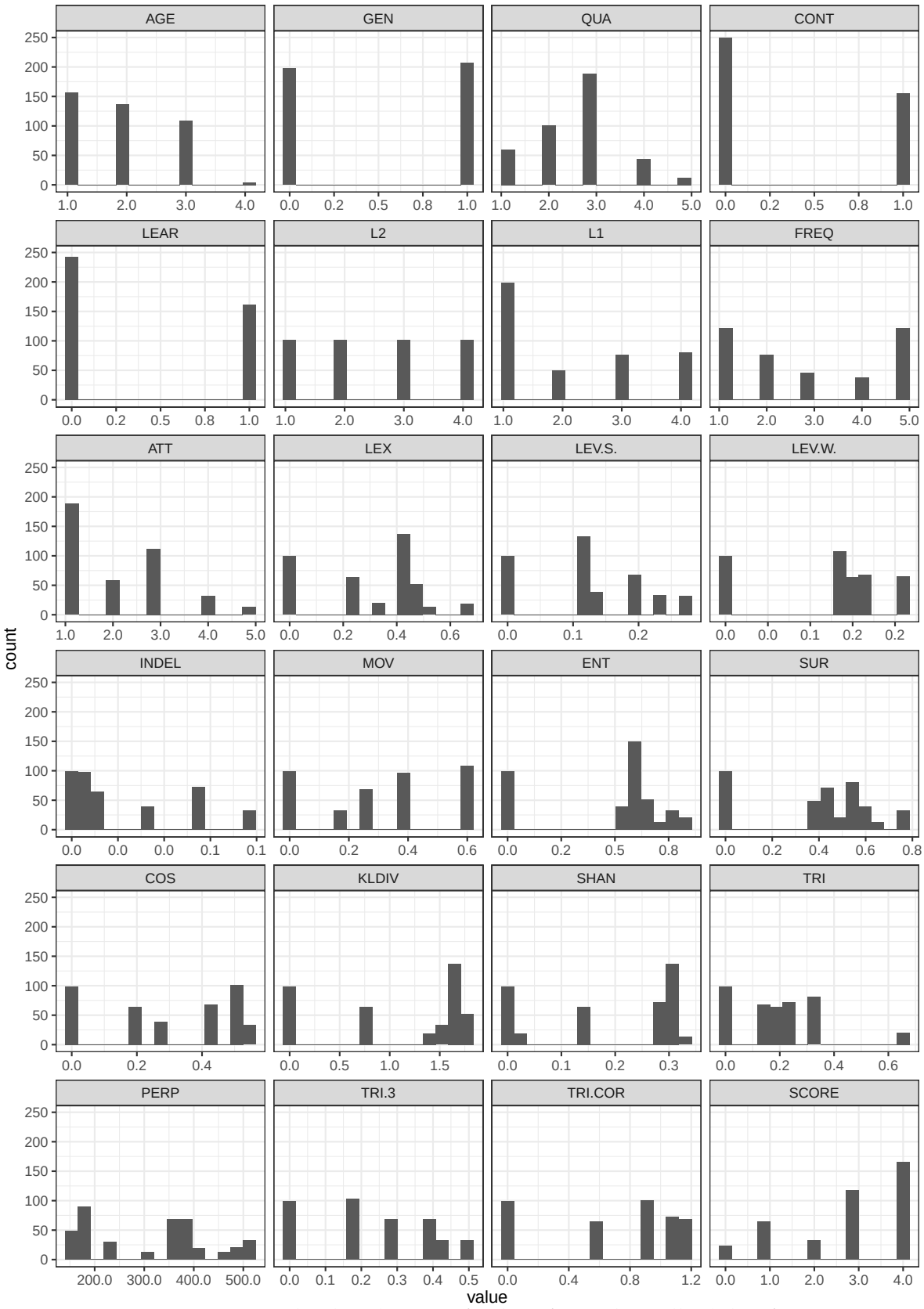


FIGURE 4.8: The distribution of values for each intelligibility feature.

and qualification do not have any correlation with any of the other features. We present the task of feature selection for intelligibility features, as well as intelligibility prediction in Chapter 6.

4.6 Summary

System-centered evaluation of retrieval systems is the most common experimentation model in IR. Test collections together with evaluation measures are used to evaluate retrieval effectiveness of retrieval systems. The chapter outlined the procedure used to build a test collection to test our thesis. The collection consisted of documents, queries and relevance judgements in combination with extra relevance features and intelligibility features. We conducted feature selection for the intelligibility features to select the optimal subset of features that would give the best intelligibility prediction accuracy.

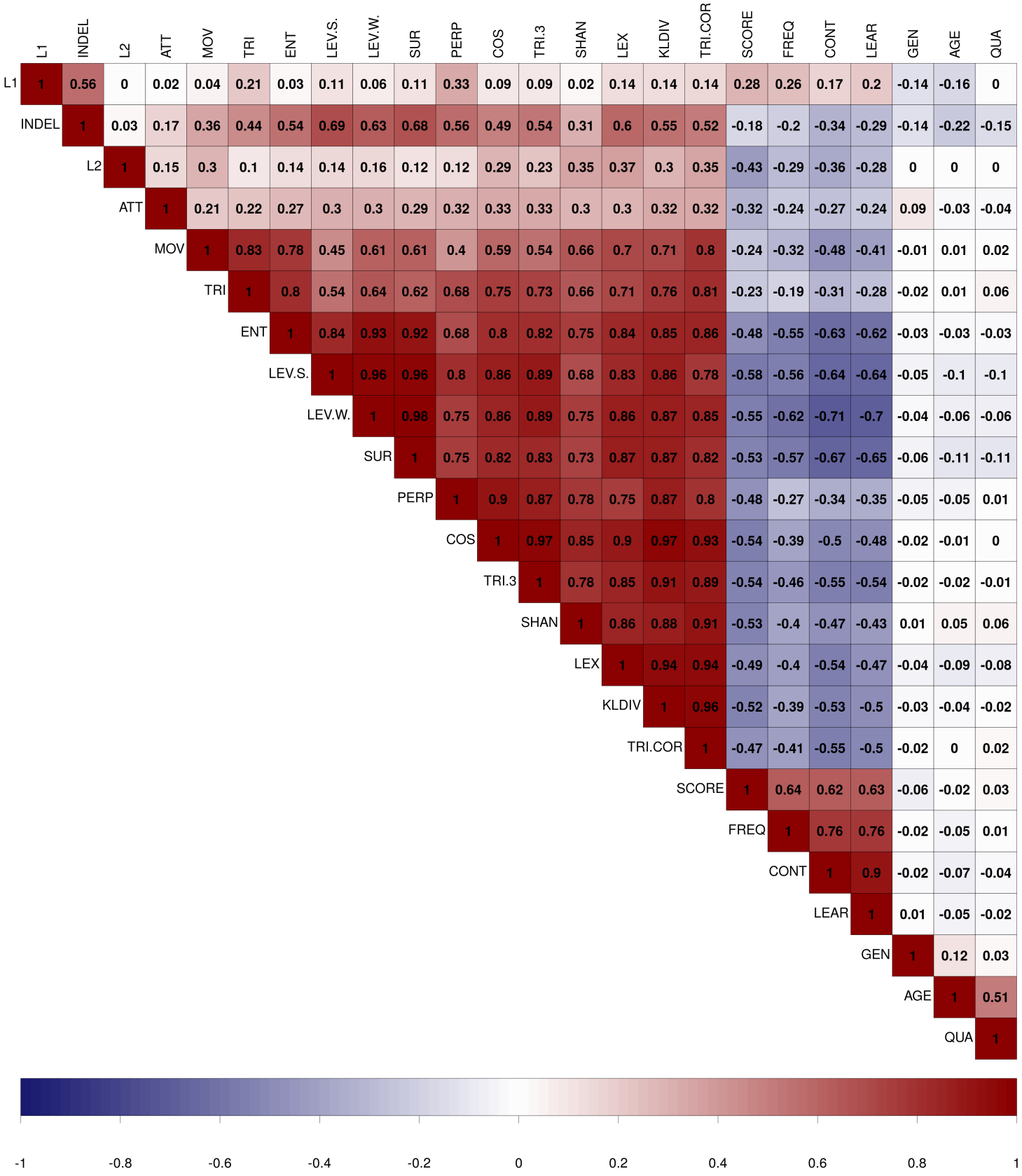


FIGURE 4.9: Plot of Pearson Correlation Coefficient of pair of features. The figures are the Correlation Coefficients for corresponding pair of features. Colour represent the intensity and the type of correlation – negative or positive.

Chapter 5

User Perspectives

The purpose of retrieval systems is to provide users with information that will meet their information needs and help them complete their tasks. Therefore, retrieval systems must be evaluated on whether they are providing information that is relevant and useful to a user given a search context or scenario. Furthermore, IR systems should provide satisfactory user experience to impact user attitudes and behaviours positively – to make searching enjoyable. Our approach for evaluating proposed techniques used a holistic approach, with using multiple methods for evaluation – using both system and user-centered search system requirements and evaluating them accordingly.

This chapter provides the research design, methods and procedures used for our user-centered approach, as well as experimental results and their discussions. Firstly, we investigated user interaction behaviour in retrieval environments where users are presented with search results written in related languages. Specifically, we investigated the usefulness of such results, user preference of ranking search results, and emotions associated with interacting with such results. Secondly, we investigated interface features, including appropriate presentation of such results that can be used to support users to interact with search results written in related languages.

Presentation of search results written in multiple related languages, rather than only results written in the query language, may affect user search experience. It is therefore necessary to study the emotions that the user may experience during search episodes, the behavior that a user engages in while interacting with the system, and the thoughts that a user may be experiencing while interacting with the system. User-oriented IR evaluation aims to understand the thoughts of a user during and after a search process, and the emotions a user experiences in search episodes, as well as the behaviour a user engages in during a search period, or as a result of using a system.

To understand user perspectives on interacting with search results written in related languages, the following questions were investigated:

RQ1 Are search results written in closely related languages useful to the user?

RQ2 What are the ranking preferences of users for search results written in related languages with varying intelligibility? Does intelligibility matter in the rank preference of such results?

RQ3 What types of emotions do users experience when interacting with search results that require intercomprehension?

RQ4 What is the appropriate presentation style for related languages' search results?

A user study was done to answer the first three research questions. Participants conducted several tasks associated with the research questions in a single session. To answer the fourth question, user centered design was used to design, build and evaluate an interaction appropriate for users to search and interact with search results written in related languages.

5.1 User Study on Intercomprehension

To investigate user interaction behaviour and usefulness of search results written in related languages, a controlled user study was done. The study had four tasks, namely: ranking of search results, completing search tasks, emotional reflection and text comprehension. Specifically, we studied the usefulness of search results that assume intelligibility and relevance. The user study also investigated ranking preference of users for search results that have different levels of intelligibility and relevance. The type of emotion associated with different search episodes involving different intercomprehension demands were investigated through retrospection.

5.1.1 Research Design

The study was designed as a within-subject research study. Participants completed the same tasks regardless of their prior language knowledge. Each participant completed the tasks in a single session with an average completion time of 60 minutes. We used a Graeco-Latin square to rotate tasks to avoid task sequence interference and to minimise fatigue effects. The study had a single independent variable, namely, intelligibility and four dependent variables based on the subtask: (i) emotional experience, (ii) rank, (iii) search task completion and document usefulness, and (iv) comprehension. Pre-defined sets of documents were presented to the participants regardless of their search queries. The languages of relevant

documents were varied and rotated around four languages, namely Citumbuka, Chichewa, Cinyanja and Citonga. Additionally, two documents in Luganda and Cisena were included in the retrieved documents but none of these documents was relevant.

5.1.2 Participants

An invitation to participate in the study was distributed via email to university students and through social media outlets for Zambia and Malawi student societies. All participants were living in South Africa at the time of the study. We were particularly interested in the participants' language competencies. As such, participants self-reported mother tongue competency on the registration forms was used to assign participants a language for the study. Twenty four respondents (13 male and 11 female) were enrolled into the study. No competency tests were performed for participants to qualify for the study. Participants signed a consent form before taking part in the study.

5.1.3 Steps and Apparatus

The study was divided into three stages: (i) participants first filled a demographic questionnaire, (ii) performed four search tasks and (iii) translated four documents written in other languages and submitted a score on how they understood the contents of the document. Figure 5.1 illustrates the procedure used to conduct the study.

Participants first completed an entry questionnaire that consisted of demographic questions, language competency questions on five languages, i.e., Citumbuka, Citonga, Cisena, Chichewa and Cinyanja and questions on their search experience using their L_1. A participant's L_1 was used to assign languages in which the study was to be conducted. A post-task questionnaire was administered after completion of each search task to ascertain the emotional episodes associated with each search task. The questionnaire had four questions taken from the Geneva Appraisal Questionnaire (GAQ) (questions 4, 5, 8 and 33). We also adapted GAQ's question 34 for use with Plutchik's wheel (Plutchik, 1980) to provide more choices of emotions than those listed in GAQ. Plutchik's wheel lists several emotions with varying intensity and is used in studies where emotional intensity is important (Poretski, Lanir, and Arazy, 2019). A post-session interview was used to obtain data to understand more about participants' general attitude and behaviour towards intercomprehension in retrieval.

5.1.4 Search Tasks and Topics

The search task consisted of four sub-tasks, namely: ranking of search results, completing search tasks, retrospection of emotion episodes and testing of text comprehension. Four search problems were used in the study and all participants completed the same search

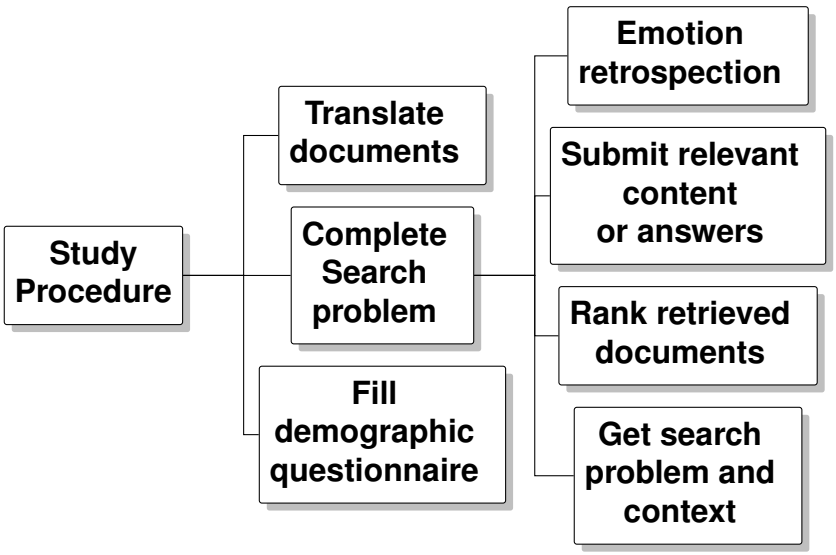


FIGURE 5.1: Procedure Used to Conduct the Study

problems. These search problems were formulated by three assessors selected from the respondents interested to take part in the study. These assessors did not participate in the user study. Table 5.1 shows the search tasks participants completed in the study. The search tasks

TABLE 5.1: Description of tasks completed in the search task

Task	Title	Description
Task 1	Benefits of Drinking water	What are the benefits of drinking water? What is the minimum quantity of water an adult should drink in a day?
Task 2	Prevention of diseases	How can people protect themselves from sicknesses?
Task 3	Life after death	What theories do different communities teach about the place where those who have died go?
Task 4	Origin of life	What theories have people formulated about the beginning of life?

were written in English, and the assessors translated the tasks to three languages, namely: Chichewa, Citumbuka and Cinyanja. Thereafter, the assessors judged the documents for relevance. The assessors used graded relevance to assess the documents based the grades provided in Table 5.2. In total, 24 documents were presented to participants in the search task and five documents in the text comprehension task. Only six documents were returned in each search task. All participants were presented with the same documents regardless of the language used in the study. However, the search problems were presented to participants in their L_1.

TABLE 5.2: Relevance Grades Description

Score	Grade	Description
0	Not relevant	The result does not contain anything useful or related to what I am looking for
1	Marginally relevant	The result is related to what I am looking for but does not give me the exact information.
2	Fairly relevant	The result does contain some information related to what I am looking and is overall useful.
3	Highly relevant	The result perfectly matches my information need and I am satisfied with it.

5.1.5 Procedure

Participants conducted the study individually. Each participant was welcomed and was led to the researcher's laboratory where experiments were being conducted. The researcher explained the purpose of the study and tasks to be completed. Thereafter, participants were asked if they were willing to proceed with the experiment. If the participants were happy to proceed, then they signed a consent form. The researcher explained the procedure for completing the experiment and participants were given login details for the Web application custom built for the study. After a successful login, a tutorial page about tasks to be done was loaded and participants proceeded to fill a demographic questionnaire after reading the tutorial.

5.1.6 Search Problem

After completing the questionnaire, participants proceeded to complete the search tasks. Firstly, participants were presented with a page explaining an information need. The information need description had the following sections following TREC topic style: title, description and a summary narration of what a relevant document should contain.

This information was given in the language assigned for the study for each participant. The information provided at this stage was used to complete the following tasks:

1. **Ranking:** Participants proceeded with a search task and six documents were retrieved pre-selected for the search problem, and presented to participant. Participants ranked the retrieved documents by dragging documents to preferred positions – participants were asked to order documents the way they would have preferred a search engine to rank them.

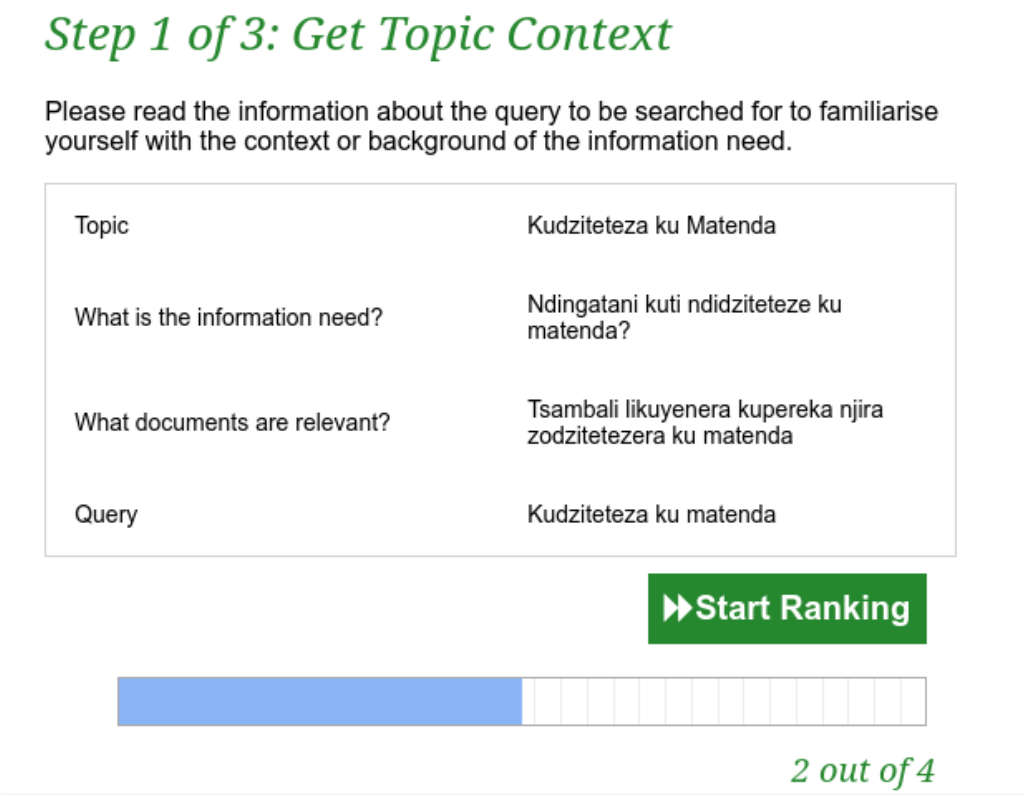


FIGURE 5.2: Example of a topic presented to participants

2. **Search Task completion:** After submitting the preferred order, participants received the same set of documents, but ordered using their rank preferences. In this page, the participant was asked to find the relevant content that would satisfy the information need. Once the participant was convinced of the answer, the participant clicked on a button to submit: i) the relevant information obtained in the documents and formulated in the language used in the study; and (ii) the titles of documents where the answer or relevant content was found.

3. **Emotion Introspection:** After completing each search task, participants answered questions about emotions experienced when they found the answers to the information need. The questions aimed to understand the emotional response of the participant as a consequence of reading documents in certain languages to complete the tasks.

After completing each search problem and all the three sub-tasks associated with it, a new search problem was loaded. After completing four search problems, participants were given a page describing the translation task.

TABLE 5.3: Text Comprehension Scale. Scores were used to specify how each comprehensible text was to someone with a specific L_1

Score	Label	Description
0	Not comprehensible	I understand nothing
1	Marginally comprehensible	I recognize a few words
2	Fairly comprehensible	I understand a few sentences or some sections
3	Comprehensible	I understand everything except a few words
4	Excellent	I understand everything

5.1.7 Text comprehension

Four documents, each written in a different language, were used in the text comprehension task depending on L_1 of the participant. Each participant translated the title and the first two sentences of a paragraph written in a language different from their L_1. Also, the participant gave a score on how they understood each of the documents. Table 5.3 provides the reference used for evaluation in the text comprehension task. Participants exited from the system after completing four translation tasks. The participant notified the researcher that the study was complete. The researcher asked participants for their comments about the tasks, especially their perception to completion as well as how they were satisfied with the provided search results. Finally, the participant signed a payment form and was given a compensation of R100 (US$7).

5.1.8 Results of Intercomprehension User Study

Interacting with a search system that provides results written in closely related languages may produce user interaction behaviour that is specific to such a system. To provide an enjoyable search experience that users struggle less in their interaction, it is essential to understand how users may interact with such results and leverage this knowledge to build search interfaces that provide appropriate affordances for such interactions. In this section, we present experimental results for user interaction with search results retrieved based on relevance and intelligibility with the query language. Particularly, we present results on a user study concerning usefulness of search results, user ranking preference and emotions associated with this kind of interaction in a search session.

We designed a user study to understand user experiences and behaviour in retrieval scenarios where users apply intercomprehension to meet their information needs. Specifically, we investigated the ranking preference of search results by users in this context. We

also studied emotions associated with intercomprehension in retrieval through retrospection. We first report results on participant characteristics in terms of their general population information, search experience and language competencies. This is followed by ranking preferences task results and their analysis. Finally, we provide search results on emotions and search completion.

Participant Characteristics

Participant Demographics: The twenty four participants were from various age groups, including : 18 to 25 (11), 26 - 35 (7) and 36 - 50 (6). They also had diverse educational background as follows: Science (7), Law (3), Humanities (5), Commerce (2), Engineering (3) and Health Sciences (4), and studied at different levels, namely PhD(7), MSc(6) and Undergraduate(11). Participants were requested to include their search experience with the languages being studied. Sixteen out of twenty four (67%) participants claimed to have used their L_1 to search or read information on the Web on topics such as current affairs, music, poems, plays and videos, religious material and translation of words. Some of the participants indicated that they use English to search for local content as a strategy to find relevant content, which cannot be found using their mother tongue.

Language Competencies: Participants who speak the following languages as their L_1 were recruited: Citumbuka(7), Cinyanja(6) and Chichewa (11). Participants also provided self reported language competency scores before and after completing the search tasks. Figure 5.3 shows a bar chart for self reported competency levels of participants before the study. All the participants had good knowledge of Chichewa. Most of the participants had some knowledge of Cinyanja and Citumbuka. Most of the participants had no knowledge of Citonga and Cisena. After finishing the search tasks, participants performed a text comprehension task – four documents written in Citumbuka (tum), Chichewa (ny), Citonga (tog) and Cinyanja (nya) were given to participants to read and score themselves on how they understood the documents on a scale of 0 to 4. Figure 5.4 shows a bar chart of text comprehension scores for the participants.

The scores reported in the text comprehension task were higher than self reported scores obtained at the beginning of the study – some of the languages may have been unfamiliar to the participants. We measured the variation of competency scores between the text comprehension task and self-reported or opinion scores and calculated Intraclass Correlation Coefficients (ICC). We calculated ICC estimates and their 95% confidence intervals based on single measure, absolute-agreement and 2-way mixed-effects model. The ICC value was 0.641 and its 95% confidence interval was between 0.257 and 0.811, which means there is 95% probability that the true ICC value is on any point between 0.257 and 0.811. The ICC values show that there is poor to strong agreement between the two methods. Previous evaluations have found similar relationships in scores. Gooskens and Swarte (2017) have

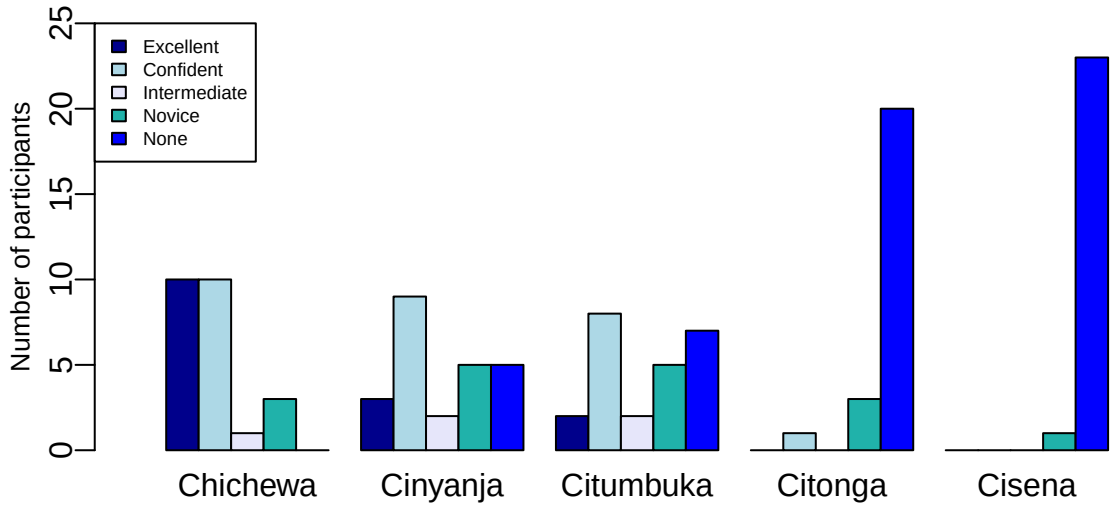


FIGURE 5.3: Self-reported competency scores for participants before completing any task in the study

proposed the use of functional testing as its scores have shown to correlate more to true intelligibility than opinion testing. Therefore, we used text comprehension scores in analysis results for the other tasks.

User Ranking Preferences

Ranking is one of the core tasks in retrieval – the effectiveness of a search engine is usually evaluated by how it ranks documents (i.e., a good system should rank the most relevant documents highly). Obviously, this depends on the task at hand as well as user preferences. Users ranking preference may be based on several attributes depending on specific user search context. Our work explored retrieval for related languages and we investigated the interplay of relevance and intelligibility from the perspective of users interacting with search results written in related languages. Our study aimed to answer the following questions:

RQ2 What are the ranking preferences of users for search results written in related languages with varying intelligibility? Does intelligibility matter in the rank preference of such results?

Participants' explicit rank preferences of six documents for the four tasks were used to conduct correlation of rankings, ranking distribution and goodness of fit analysis for the ranks against our hypothetical ranking. We transformed user submitted position of a document

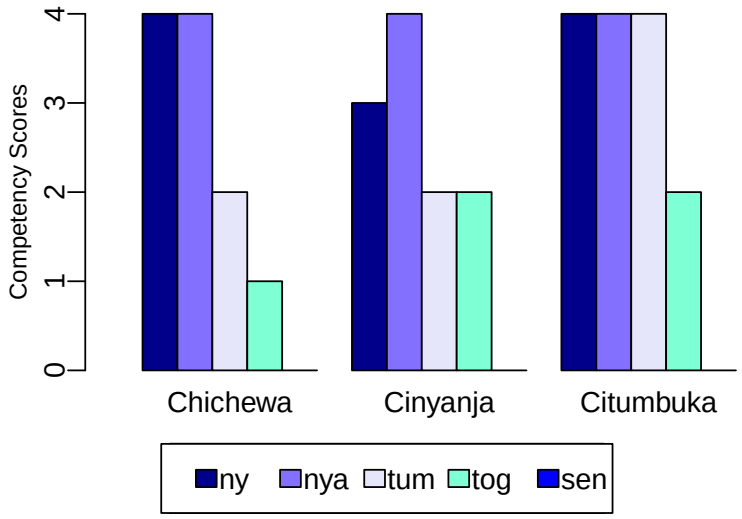


FIGURE 5.4: Average reading comprehension scores by L_1 speakers.

to a numerical value as a rank, i.e., the document on the first position was given the value 1 and the rest of the documents were processed in this way.

Correlation of Ranking

We analysed the user submitted rankings for each participant against all the other participants regardless of their L_1 to find out whether the rankings were similar. We used Kendall Rank Correlation Coefficients (Kendall Tau τs) for each ranking provided by each participant against every participant ranking. Figure 5.5 shows the plot of correlation coefficients of the rankings. The plot shows that there is some similarity in rankings of different degrees for most participants; this may be due to participants trying to rank relevant documents highly. However, the rankings of two participants are not similar to the rest of the participants.

Rank Distribution by Language

Further, we also wanted to know if participants' L_1 may have influenced how they ranked the documents. Accordingly, we plotted box plots for each task, grouped by L_1 of the participant to observe how participants of each language ranked documents. Figure 5.6 shows the box plot of participant ranking based on their L_1. Each of the tasks had relevant documents at different positions and written in different languages.

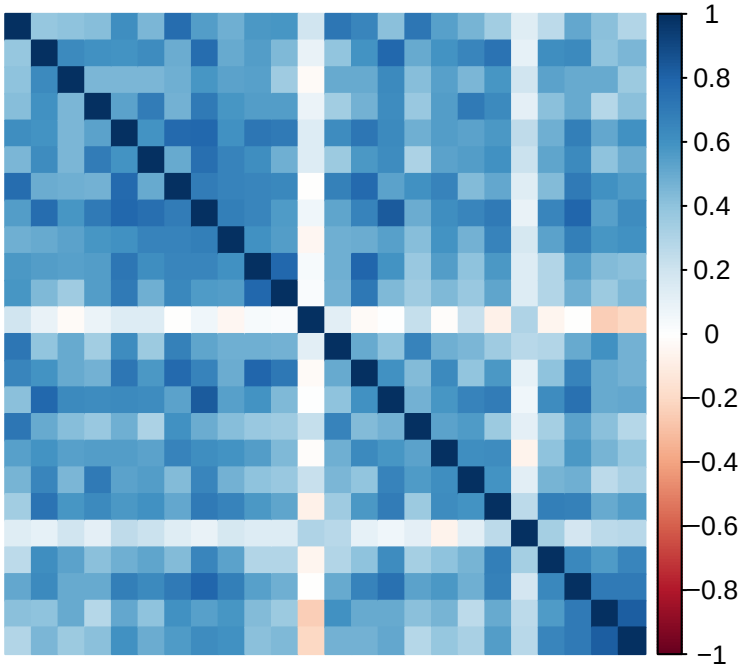


FIGURE 5.5: Plot of Kendall Rank Correlation Coefficients showing the degree of association of rankings of each participant against every participant.

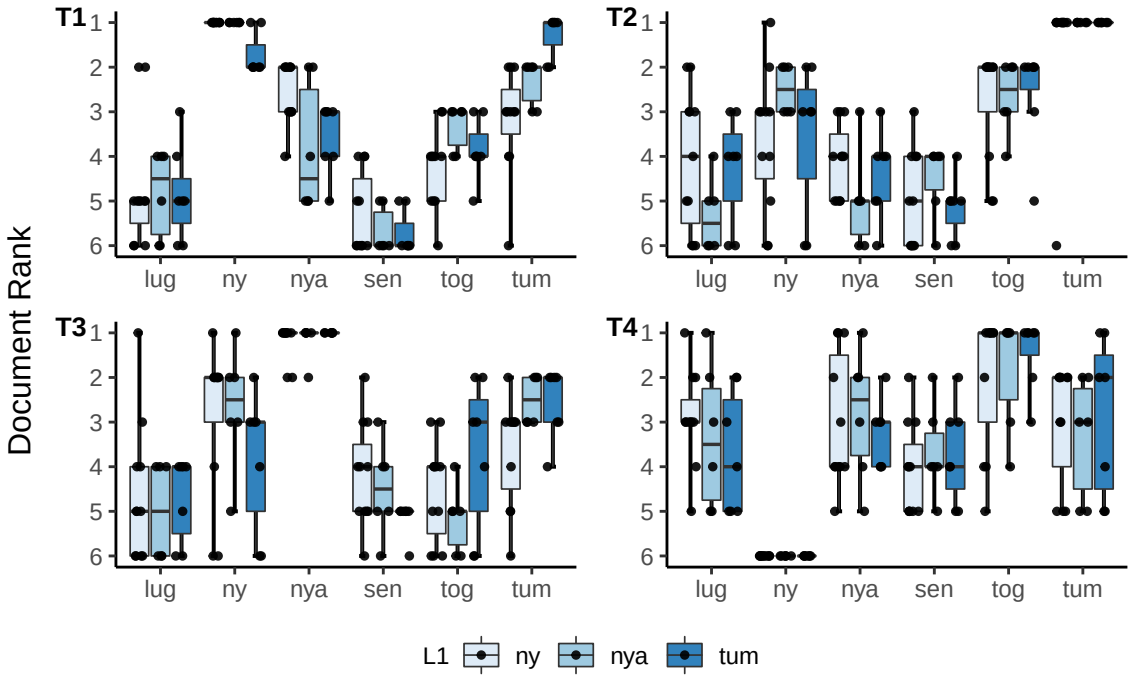


FIGURE 5.6: Rank preferences distribution for task 1 to task 4 (T1, T2, T3 and T4 respectively) grouped by L_1. The codes lug, ny, nya, sen, tog and tum represent documents written in Luganda, Chichewa, Cinyanja, Cisena, Citonga and Citumbuka respectively.

The first task had three relevant documents written in Chichewa, Citumbuka and Cinyanja,

with Citumbuka and Chichewa documents being highly relevant while the Cinyanja document was fairly relevant. Cinyanja and Chichewa speakers ranked the Chichewa document highly while most of the Citumbuka speakers ranked the Citumbuka document highly. This shows some evidence of users preferring documents written in their own languages in the presence of multiple relevant documents. Cinyanja and Chichewa participants were not familiar with Citumbuka, and relied on intercomprehension to decide on the ranking, i.e., these participants mostly ranked the Citumbuka document on either position 2 or 3. Surprisingly, the Luganda document was ranked more highly than Cisena document although both documents were irrelevant and Cisena is in the same family of languages with the rest of the other languages. This might be due to participants not interested in ranking documents they know or guessed to be irrelevant but also incomprehensible to them.

The relevant document for the second task was written in Citumbuka. Almost all participants ranked the Citumbuka document on first position except one Chichewa participant who ranked it at position 6. The ranking of Citonga document was also consistent on position two with a few participants ranking it at different positions.

The relevant document for the third task was written in Cinyanja. Three documents written in Chichewa, Citumbuka and Citonga discussed related content but their topics were far from that of the task. Most of the participants ranked the Cinyanja document highly. Citumbuka and Chichewa documents were ranked relatively higher by most participants unlike the Citonga documents, which may have been due to the low relevance as well as intelligibility, i.e., Citonga is not widely spoken and almost all participants were not familiar with the language.

The relevant document for the fourth task was written in Citonga, which is one of the languages most of the participants were unfamiliar with. Participants used intercomprehension to rank the Citonga document. Sixteen out of twenty four (16 out of 24) participants ranked the Citonga document on the first position. Moreover, Luganda and Citumbuka documents discussed broad topics related to the search task topic and were ranked highly as well.

Our analysis demonstrates that participants preferred documents written in their language or closely related languages to be highly ranked when multiple relevant documents were available. Fairly relevant documents with higher intelligibility were ranked lowly. Most of the participants did not rank documents that were not relevant even in cases when the documents were highly intelligible to them. For example, a Chichewa document in the fourth task was ranked at the fourth position – this may have been the last document in the list of documents when participants first got the search results.

Ranking by Relevance and Intelligibility

The idea to provide search results written in closely related languages of users assumes that users would prefer relevant documents written in their language and then, in closely related languages. Accordingly, documents written in a language that is completely incomprehensible are not desirable and should be ranked low. Similarly, documents that are irrelevant but comprehensible are not useful. This assumption led to the following hypothesis:

- Users preference of ranking is primarily based on relevance and then intelligibility.

This hypothesis stipulates that users would want documents to be ranked based on relevance and intelligibility if intercomprehension is assumed as follows: i) Relevant and comprehensible documents should be ranked highly, ii) relevant documents but less comprehensible should follow, iii) if relevance is the same, priority in ranking should be given to more comprehensible documents to the participant. Essentially, ranking should be based on relevance first and intelligibility should be used as a secondary attribute.

To investigate these assumptions, we created a hypothetical ranking for each L_1 using the assumed constraints. We used average scores from the text comprehension task for participants of a specific L_1 and relevance judgements of documents provided by monolingual assessors to construct a single ranking for each task for each group of L_1. We then investigated the similarity between the sample ranking provided by participants and our hypothetical rankings. We used the approach proposed by Melucci to compare the rankings using Kendall Tau τ and Kolmogorov-Smirnov D statistics and tests (Melucci, 2007).

Data Transformation: Since there were multiple rankings for each task for participants of a specific language, the rankings were aggregated using Borda Count voting model. Borda Count is an election method in which voters rank candidates by preference and the winner is chosen based on the points accumulated from number of candidates ranked lower than this candidate. Borda Count has previously been used to aggregate search results from multiple search engines, and the results obtained so far are comparable to more advanced techniques using supervised learning (Aslam and Montague, 2001; Liu et al., 2007; Wu, 2012).

We merged the rankings of documents in each task by L_1 into a single ranked list using the Borda count rank aggregation approach as follows: Given a set of rankings for task i, R_i = $R_{i1}, R_{i2}, \ldots, R_{im}$ (where m is the number of participants using $L1_k$), of a set of documents D_i = d_{i1}, d_{i2}, …, d_{in} where $n = 6$. For each ranking R_i, assign to document d_{ij} points equal to the number of documents ranked lower than itself + 1, i.e, a document ranked on first position gets n or 6, second position gets $n - 1$ or 5, third position gets 4 and last position gets 1. The total count for document d_j is the number of points it accumulates from all its rankings seen in the sample for participants with this L_1 on this task. The accumulated points are used to rank documents in descending order for task i for participants using $L1_k$.

Ranking Correlation using Kendall Tau: Kendall Tau measures the strength of association between two sets of ranks given to a same set of objects. We test the null hypothesis that Kendall Tau is zero ($\tau = 0$), i.e., the two sets of ranks are not similar. Our alternative hypothesis is that the ranks are correlated or similar, i.e., Kendall Tau is non-zero ($\tau > 0$). The Kendall Tau statistic values in Table 5.4 indicate that there is a strong correlation be-

TABLE 5.4: Kendall Tau Correlation between our hypothetical and sample rankings grouped by language. H_0:($\tau = 0$). Reject null hypothesis ($p <= 0.05$).

L_1	**Kendall Correlation Coefficient**	**p-value**
Citumbuka	0.61	0.00018
Chichewa	0.64	8.9e-05
Cinyanja	0.63	9.9e-05

tween the observed user ranking and the expected rankings from our hypothetical ranking algorithm. The p-values are very small ($p < 0.05$) and we reject the null hypothesis ($\tau = 0$). Therefore, we conclude that there is evidence to support that the rankings provided by the participants are similar to the hypothetical rankings.

Goodness of Fit Test: We also wanted to investigate if the rank distribution between the hypothetical ranking and empirical ranking provided by the participants come from the same distribution or follows the distribution of our hypothetical rankings and we used the Kolmogorov–Smirnov (K-S) Test. The K-S statistic quantifies the distance between the empirical distribution function of the sample and the Cumulative Distribution Function (CDFs) of the reference distribution, or between the empirical distribution functions of two samples. K-S Test is suitable for ordered categorical data (Melucci, 2007; Jann, 2008) with a large sample size. Accordingly, we used a variation of K-S Test with bootstrap method to estimate the best p-value for D (Jann, 2008).

TABLE 5.5: Kolmogorov–Smirnov Statistic and test by L_1

L_1	**Kolmogorov–Smirnov** D	**p-value**
Citumbuka	0.042	1
Chichewa	0	1
Cinyanja	0	1

Our null hypothesis is that the two sets of rankings come from the same distribution. The obtained D values in Table 5.5 indicate that the two samples come from the same distribution, i.e. D is close or equal to zero (D gives the maximum distance between distributions) of the two samples). Therefore, there is evidence to support that the two distributions are the same, i.e., our p-value is 1 (p > 0.05 for 95% significance level).

Through our analysis, it has been shown that users ranked relevant documents highly for search results written in related languages. Intelligibility was used as a secondary criterion.

Search Task Completion

We investigated whether participants were able to complete each task to find out whether users find the information provided in closely related languages useful. Ultimately, we wanted to answer the following research question:

RQ1 Are search results written in closely related languages useful to the user?

Participants were asked to complete a search task by submitting the title(s) of relevant document(s) and topic answers that they found from the documents. The first task had three relevant documents written in Cinyanja, Chichewa and Citumbuka. The topic was an informational task and information required were facts. All the 24 participants were able to complete the task. The second and third topics were also fact topics with one relevant doc-

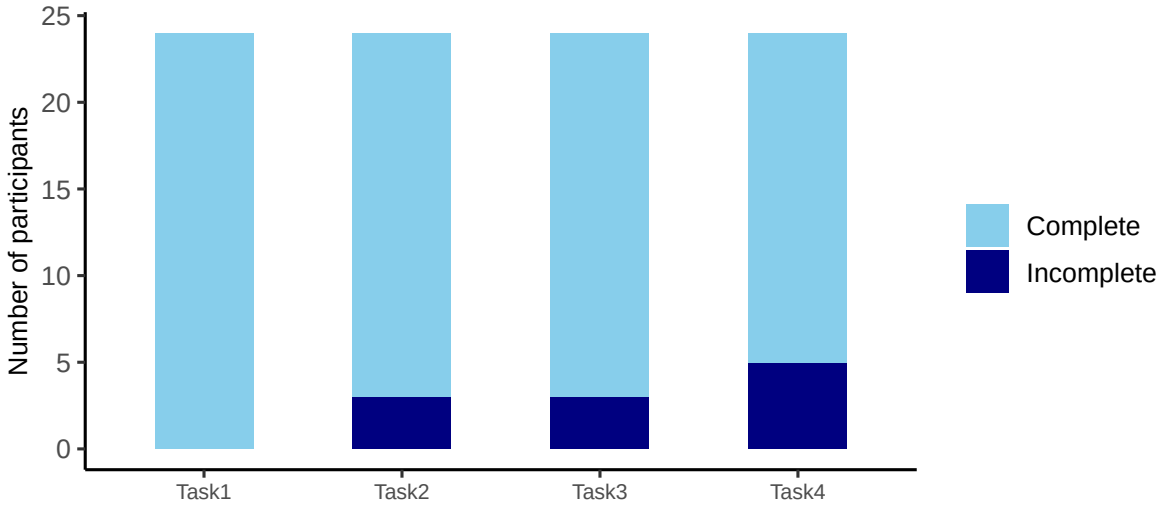


FIGURE 5.7: Task Completion Status. Number of participants completing tasks decreased as intelligibility decreased.

ument each written in Citumbuka and Cinyanja respectively. Twenty one (21 out of 24) participants were able to complete Task 2, i.e., two Chichewa and one Cinyanja participants were not able to submit answers for the task. Twenty one (21 out of 24) participants completed Task 3, while three (3) Chichewa speakers were not able to complete. Analysing the data further showed that the same two participants did not complete tasks in both tasks and their rankings were out of agreement with those provided by other participants sharing the same L_1. The relevant document for the fourth task was written in Citonga, and the task required analysis of content to submit an answer. Moreover, the document was written in a language participants were not familiar with. 19 out of 24 participants completed the task.

Our analysis shows that most of the users were able to complete the task and provided an answer to the search problem. Focusing on the individual task, our observation was that the number of participants completing tasks decreased as intelligibility decreased.

Emotions and Intercomprehension

Emotions affect behaviour and may cause users to approach or avoid a system. Therefore, our user study is designed to explore the affective aspects of intercomprehension in a retrieval scenario. Our aim is to understand the emotional states of users when interacting with search results that require application of intercomprehension. The effort required to understand a document written in an unfamiliar language may lead to frustration and stop a user from completing their search task completely. Figure 5.8 shows the classification of emotions grouped by the L_1 of the participant and task. More participants reported negative emotions in scenarios where intercomprehension was required to complete a search task.

After each search episode, participants recorded their emotion experience using a Geneva Appraisal Questionnaire (GAQ) and Plutchik's wheel. The Plutchik's wheel allowed participants to explicitly specify the type of emotion that they had just experienced in the task. We classified the emotions provided into two classes, namely, negative and positive emotions for each participant of each task (Plutchik, 1980).

A few participants reported negative emotions in the first task. There were three relevant documents written in three languages that were L_1 languages for the participants. Participants whose first language was Chichewa reported more negative emotions in tasks involving relevant documents written in languages not closely related to their own L_1. The relevant document for the third task was written in Cinyanja. Most of the participants reported negative emotions in the fourth task, which required intercomprehension for all participants.

Participants were asked to indicate if they struggled to complete a task. We wanted to explore if there is any association between the type of emotion and whether a participant struggled or not. We conducted Fisher's exact test of independence, which provides a p-value for the test and Confidence Interval (CI) for odds ratio. The odds ratio shows the strength of association between two variables, with the null hypothesis being that struggling status and type of emotion are independent, i.e, odds ratio = 1. The results for the first task

TABLE 5.6: Fisher's Exact Test Results for each task

Task	**p-value**	**odds**	**CI (95% CI)**
Task 1	0.001976	0	0 to 0.3225
Task 2	0.01087	0	0 to 0.7204
Task 3	0.5212	0.4882	0.02014 to 34.6330
Task 4	0.357	0	00 to 23.4001

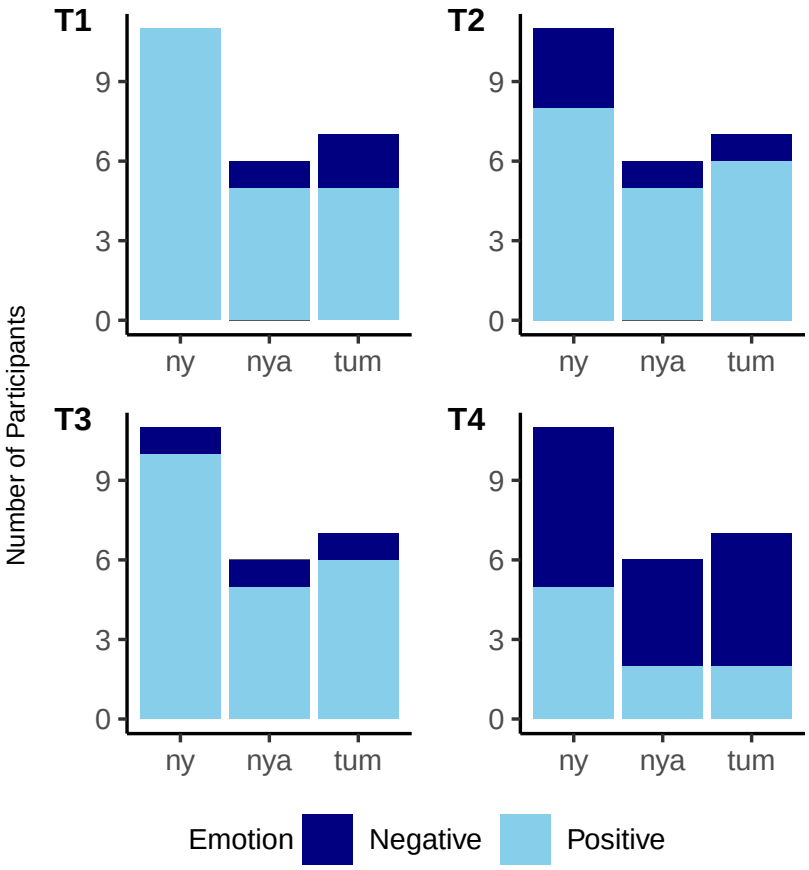


FIGURE 5.8: Classification of emotions: plot shows the number of participants experiencing negative and positive emotions while completing Task 1 (T1), Task 2 (T2), Task 3 (T3) and Task 4 (T4). ny, nya and tum represent Chichewa, Cinyanja and Citumbuka respectively. The number of participants who experienced negative emotions increased as intelligibility decreased.

indicate that struggling and emotion status are not independent. The p-value is very small ($p < 0.05$) and 95% CI for the odds ratio spans from 0 to 0.32 and does not coincide with 1, i.e., we reject the null hypothesis that odds ratio = 1. The same result is observed for the second task. Therefore, the results for first and second tasks are significant.

Surprisingly, for the third task, p-value is very big and therefore, we have insufficient evidence to reject the null hypothesis ($p > 0.05$ for 95% significance level). Similarly, the CI for the third task contains 1. However, the CI is large, i.e., the odds ratio is not precisely estimated. Similar results are observed for the fourth task. Therefore, we conclude that there is not enough evidence to support our hypothesis that the type of emotions one experience is independent from struggling in the search task. These results suggest that participants who experienced negative emotions may also have struggled to complete tasks.

Overall, our findings suggest that participants found search results in related languages

useful: participants were able to complete search tasks using content in another language. As intelligibility decreased, more participants struggled to complete tasks and experienced negative emotions. Our observation on rank preference suggests that participants preferred documents to be ranked based on relevance as a primary criterion and then intelligibility. This simplistic view is impractical in real search systems where the relationship between intelligibility and relevance may be complex – a ranking function that balances both objectives to improve the utility of the user may be appropriate. Additionally, we investigated the problem of ranking search results using multiple criteria (e.g., relevance and intelligibility) – and supervised ranking was used as one of the methods. We further explored user interaction in the context of the user interface to determine the appropriate search engine features to avoid overwhelming users (see Section 5.2).

5.2 User Interface Design

A search engine that matches, retrieves and presents the user with information written in related languages may require special interface features to make the user aware of the characteristics of the results and to appropriately present them to the user to improve the user's search experience. We adopted User-Centered Design (UCD), an iterative design process in which user needs and context of use are fully investigated and users are involved throughout the development process. We describe the process in Section 5.2.1. We conducted several design and evaluation iterations and the process is described in Section 5.2.2. The evaluation procedure of the interface designed is given in Section 5.2.3. Section 5.2.4 presents the results of the evaluation.

5.2.1 User Centered Design

User-centered design was used to discover appropriate features and interactions for the proposed search system. Interviews and participatory design sessions were used to come up with ideas about features, design interactions and interfaces to be included. Design and evaluate steps were done in a iterative manner. At the end, the emerging design was evaluated using a usability testing instrument .

In each iteration, users were involved in the design of interfaces using the following steps: i) understanding context of use; ii) user requirements, iii) design of the interfaces and iv) evaluation of user requirements. Background interviews were done to collect data on user context and needs in relation to related languages as well as usage scenarios. Ten (10) participants who speak IsiZulu, IsiXhosa and IsiNdebele were interviewed. The questions included their language competency, issues faced by multilingual speakers of RSLs as well as preference on relevance and understandability in relation to topics or type of content.

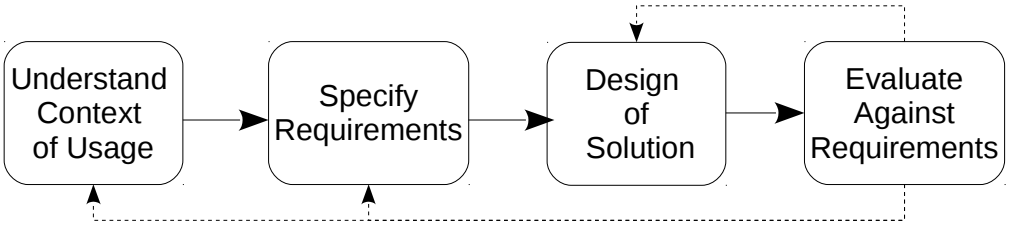


FIGURE 5.9: User Centered Approach used in the User Search and Result Presentation Interface Design

5.2.2 Design Iterations

The initial sketches of the user interfaces were based on literature from related studies. Four design sessions were conducted, and in each session, users were given a task sheet describing the purpose of the session. The initial task sheet contained a description of the project and two interfaces from literature, i.e. tabbed and interleaving interfaces, and example components that could be used. Participants were asked to discuss the prototypes and suggest changes to the presented interfaces. Participants played with paper prototypes to change the design interfaces. Participants' designs initially started with paper prototypes, whereby participants would create their own interfaces without limits. The initial phase was exploratory and divergent thinking was encouraged to gather as many ideas as possible. For each interface designed, participants were asked to explain the context and why they would prefer such an interface. Accordingly, the submitted interfaces were diverse and the subsequent sessions were done to design a better interface from the participants' proposed ideas.

In the subsequent sessions, participants worked on a single design in a participatory design session. The session had set objectives to ensure that the basic feature requirements to accommodate information search behaviour steps were met. Two stages were identified based on a previous similar study on CLIR (Petrelli et al., 2006a):

- **Query formulation and submission**: User submitting query that will be matched with documents in related language. The interface needs to have features that will allow the user to submit the query as well as enhanced features to assist the user with intercomprehension interaction.
- **Results presentation or layout**: Presentation of results to assist the user to navigate the results and complete the search task using intercomprehension.

Participants discussed what features and design layouts could enhance user experience. All

design choices were discussed amongst the participants before being added in. The designs were done on the computer using Axure 3[1], so that participants could visualise their suggested designs. At the end, participants evaluated the features based on the original objectives and their expectations. After the second session, a low fidelity prototype was created to be used in the next session. The third and fourth sessions were done using the same procedure and a prototype interface was realised.

5.2.3 User Interface Evaluation

The final interface design was evaluated in the final stage. An advert was sent out to potential participants through e-mails and social media. Fifteen (15) participants were recruited to take part in the study. Recruited participants spoke one of the languages of the content.

Search engine Architecture

The search system comprised of search and result presentation interfaces that were designed in the UCD activities, and a search engine with documents written in nine African languages. The search engine re-ranked search results based on relevance scores, i.e. normalised BM25 scores and cosine similarity scores of a corpus written in several languages (e.g., an estimation of lexical similarity between languages). The search engine had documents written in nine South African languages namely: isiZulu, Sepedi, Setswana, Tshivenda, isiXhosa, isiNdebele, isiSwati, Sesotho and Xitsonga. Content in several languages was collected to provide for a multilingual search environment as well as a variation in terms of language similarity. Documents were gathered from the Web using Web crawlers. South African Language Identifier [2] (SALId), was used to identify the language of the documents for a document to be admitted to a collection of a specific language.

The search engine used the Solr distributed architecture to index documents for each language collection. Cosine similarity scores were calculated using the South African Constitution (SAC) corpus written in nine languages. Text was used to estimate intelligibility between languages because lexicons were not available in each of the studied languages. SAC text represented a body of text that was readily available in each of the languages with high quality translations as they are public government documents. Cosine similarity was calculated based on pairs of languages to find an estimation of orthographic similarity between languages. In addition, the search system used Google Translate to translate queries and a language identifier to identify the language of the query submitted by participants. Figure 5.10 depicts the architecture of the system used for usability.

[1] Axure is a software for creating prototypes and offers drag and drop feature as well as formatting of widgets.

[2] https://rma.nwu.ac.za/index.php/resource-catalogue/nchlt-south-african-language-identifier.html

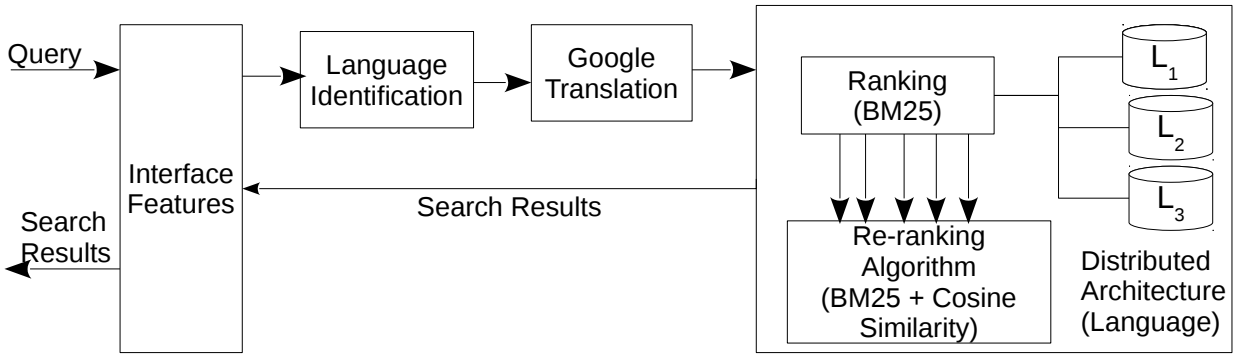


FIGURE 5.10: Architecture for System Used for Usability Evaluation

The search engine worked as follows. When a user submitted a query, the query language was first identified, and translated using Google Translate. Translations were available for only two languages: IsiXhosa and IsiZulu. Queries were not translated for the languages that had no translation tools publicly available. Documents were indexed separately by language. Queries of specific languages were run on specific collections. The matched documents were ranked using BM25, and a new normalised BM25 score was calculated based on the top ten (10) documents retrieved from each collection. Finally, the documents were re-ranked using the product of the normalised BM25 Similarity and Cosine Similarity – estimating orthographic similarity of the language of the query and document language.

User Interface Evaluation Procedure

Participants were welcomed by the researcher and taken to the experiment room. The researcher explained the study and tasks that had to be done in the experiments. Participants were asked if they wanted to proceed to take part in the study, and then they signed a consent form. Thereafter, the researcher handed out tasks to be completed and questionnaires. Participants first completed a demographic questionnaire that asked participants for such information as their age, gender, education, L_1, language knowledge, language use (read, write, what content) and search experience i.e., ever searched using any of the languages included in the study. Participants were then presented with three search tasks to complete using the search engine.

Participants searched for relevant content to complete given tasks. After finding relevant content, participants submitted their answers to ensure that they completed the tasks. This process was done to simulate a real world scenario, whereby users would use a search engine to find information that would meet their information needs. After using the search engine, users were informed of the features they did not use to complete the task. Participants proceeded to complete a System Usability Scale (SUS) questionnaire.

Participants also provided feedback as to whether they had noticed certain functionality before it was shown to them, and whether they found it useful or not. Furthermore, an in-built program in Windows Operating System called "steps recorder", was used to record user interactions with the interface. This was used to analyse whether users returned to specific functionality or not. After an informal interview with the participant, the session was complete. The average completion time was 30 minutes and participants were paid a compensation of R45 (US3). In the next section, we provide results on the choice of interface features to support users in interacting with a search engine that provides them with results in related languages.

5.2.4 User Interface Design Evaluation Results

Using resource scarce languages to search for information may be frustrating because of the poor quality of search results as well as search engines not returning search results at all. Frustration interrupts the cognitive flow of users and may negatively affect users' perception of the experience using the system. Returning search results written in related languages may introduce new interaction behaviour for users as users are required to interact with languages may not be familiar with. Therefore, users may need interface features that help them to complete their search tasks and make search experience enjoyable by minimising incidents that would negatively affect their interaction.

Through iterative UCD activities, several features and principles were identified to be crucial for interaction with the system:

- **Simplicity:** Participants wanted interfaces with fewer features but with information that would help them quickly decide on the usefulness of a search result.
- **Flexibility:** Users wanted to have control on how the results are displayed – arranging the results by language and having all the results written in several languages in one page.
- **Familiarity:** Users wanted an interface that was familiar to what they use in universal search. Also, participants suggested or chose features they were familiar with.

Several features were identified through participatory design sessions as follows:

- **Search Results Presentation Layout:** Toggling of displaying of results in tabs by language and displaying results in a single list. Figure 5.12 shows the tabbed and single interface designs used in the early stage of the UCD process.
- **Language identification:** If tools and resources are available for some languages, language identification for queries and documents should be available.

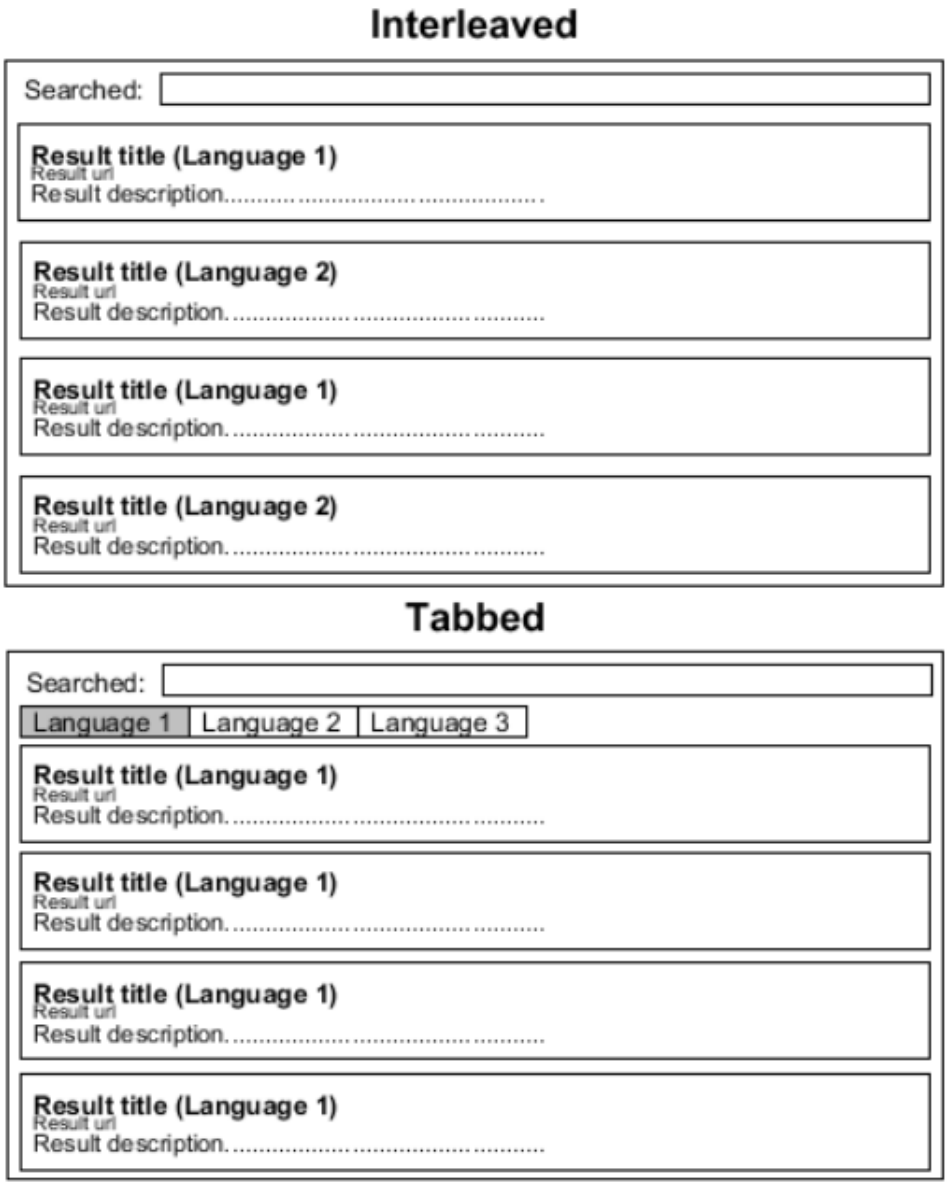


FIGURE 5.11: Tabbed and interleaved Interfaces

- **Translation:** For languages that have the resources, translation for queries should be done.

Given the context of the research, working with scarce languages, most of the languages do not have resources available to perform language identification and translation. Our evaluation system used such resources were they where available. Usability aims to discover whether an interface is easy to use. The interface is deemed usable if a user is fully aware of all features and functionalities a system provides. A usability evaluation was done on the designed search interface using the System Usability Scale (SUS) (Brooke, 1996). SUS is an instrument used to measure the usability of systems and was chosen because of its simplicity for both analysing usability of features of an interface and obtaining feedback

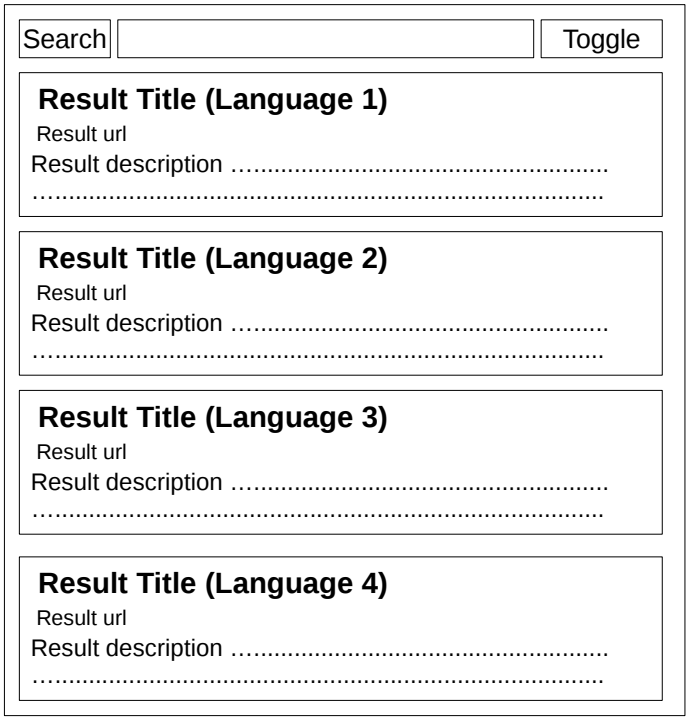


FIGURE 5.12: Evaluated Interface Design

from participants. SUS has 10 questions and participants answer using a Likert scale of five points. Table 5.7 shows the SUS questions.

TABLE 5.7: Questions in SUS Questionnaire

Number	Question
1	I think that I would like to use this system frequently.
2	I found the system unnecessarily complex.
3	I thought the system was easy to use.
4	I think that I would need the support of a technical person to be able to use this system
5	I found the various functions in this system were well integrated.
6	I thought there was too much inconsistency in this system.
7	I would imagine that most people would learn to use this system very quickly.
8	I found the system very cumbersome to use.
9	I felt very confident using the system.
10	I needed to learn a lot of things before I could get going with this system.

Fifteen (15) participants with different L_1 languages took part in the usability evaluation tasks. Figure 5.13 shows the score of each question and participant. The results show that all participants except one (scored below average of standard score for web based applications, e.g., 67.2 out of 100) were comfortable with the designed interface (Bangor, Kortum, and

Miller, 2008). The minimum SUS score was 67 out of 100, and the average score was 86. An average score of 67.2 and above is taken as a score for an interface that is usable. An average score of 86 shows that the interface was excellent. Most of the participants interacting with

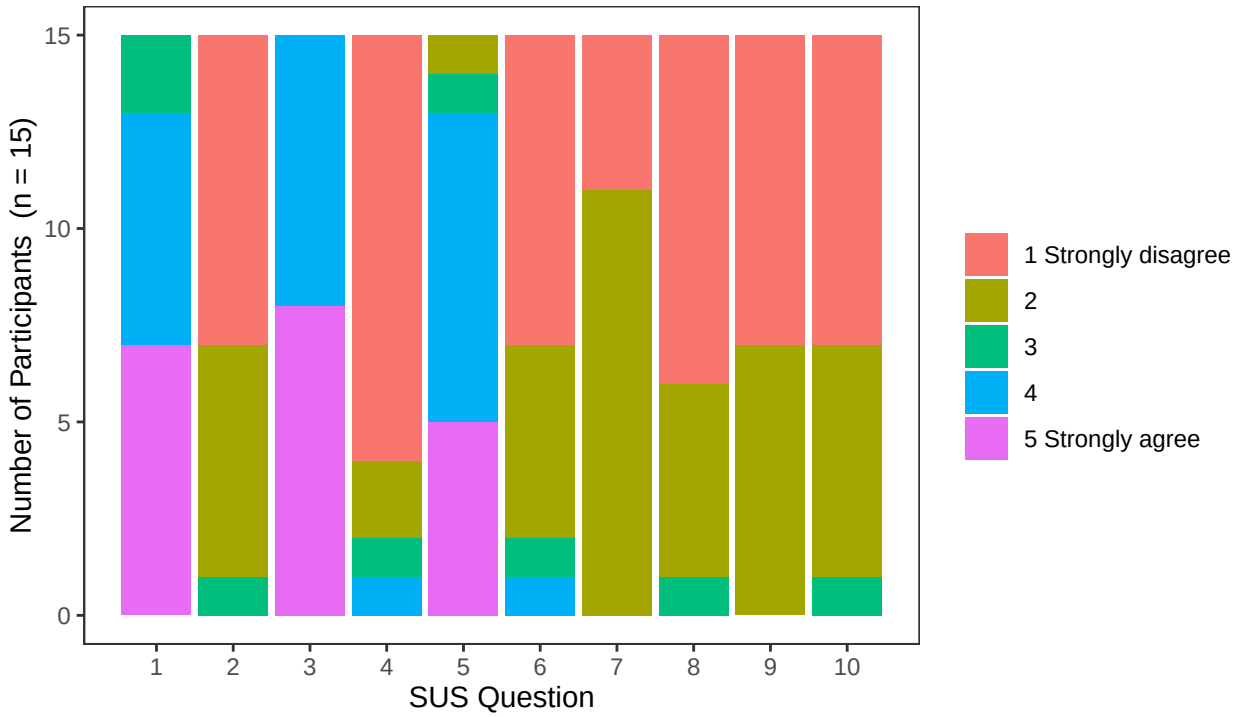


FIGURE 5.13: SUS Scores for each question.

the system did not use the switching interface feature – only two (2) out of fifteen (15) made use of the feature. Users may have had this type of interaction because of their familiarity with monolingual search interfaces used widely on the Web. The subjective evaluation also shows that participants were not confident using the system (question 7 and question 9). We think that this also due to familiarity with the layout.

After the experiment session, participants were asked about the feature to ensure that they were aware of the feature. Users were shown features available in the search system and were asked to interact with them. In the final evaluation, participants indicated that the features were useful. All the participants indicated that the toggling of results presentation was very useful (fifteen out fifteen (100%)). In addition, participants who were native speakers of the languages indicated that the translations were not helpful to them as the translations were of very low quality (only IsiZulu and IsiXhosa had translations through Google translate) and they were able to understand content written in closely related languages. Most participants indicated that they would have loved the language of the document to be annotated with the results, for them to know what language the document was in before opening the link – some users may not be able to identify the language of the document from the result snippet and may ignore the result as irrelevant. The study shows some promising results in terms of identifying interface features and layouts that may assist users to

interact with search results written in related languages. Participants in the design process requested familiar layouts combining merged results layout with tabbed layout, which they could switch depending on the results presented to them. User participants also requested additional features such as language identification to help them explore the results.

In the light of these results, the interface was changed by introducing facet like categories as a way of providing flexibility of exploring results written in different languages together or separately. Facets have been shown to assist in finding relevant documents when ranking of documents is not effective (Käki, 2005) as well as helping users learn more about the topic in complex tasks (Yee et al., 2003). Figure 5.14 shows the slightly changed interface.

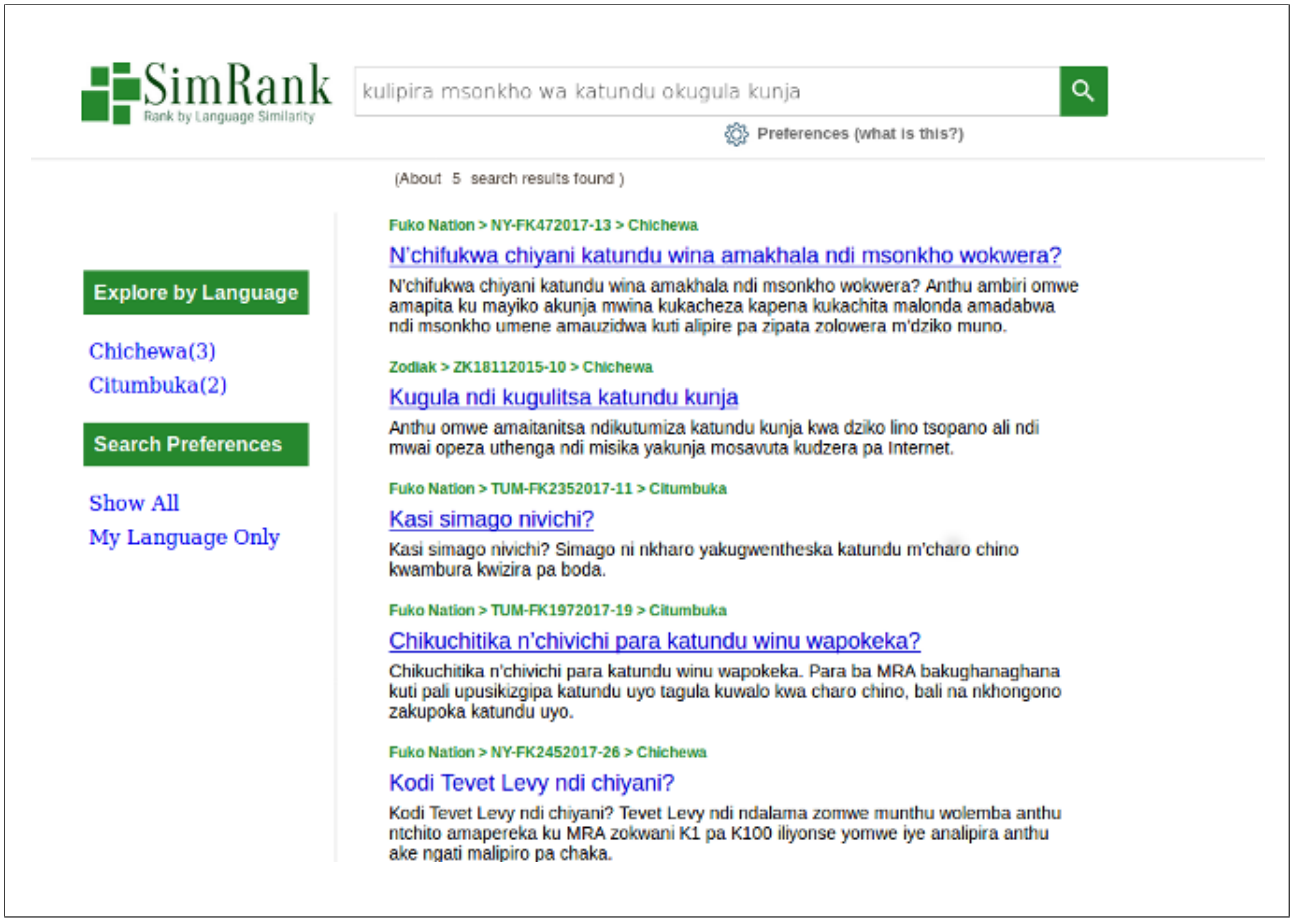


FIGURE 5.14: Revised search interface using the UCD principles and features.

5.3 Discussion

The objective of this section is to present a discussion of our findings and their implications, as well as limitations of the approaches employed in our investigation. Struggling search is typical for speakers of resource scarce languages when they search for information written in their own languages on the Web. To help such users, it is important to understand their

search experiences while interacting with search systems, and to design search systems that may support them to complete their tasks. We have explored presenting users with content written in related languages to enable users access alternative relevant content. Through a user study, we have shown that such search results are useful when resources are limited – participants of the study were able to complete search tasks with different levels of difficulty, depending on the nature of the search problem and intelligibility. Success rate of completion decreased as intelligibility went down. Our findings supplement previous studies focusing on matching of queries and documents written in related languages without translation (Buckley et al., 2000; Gey, 2007; Chew and Abdelali, 2008). Nonetheless, this thesis provides more insights on the usefulness of matched documents, as well as user search experience and perceptions while interacting with the results. This outcome is important for improving search results for speakers of resource scarce languages.

We also focused on understanding user ranking preferences for search results written in related languages but with varying intelligibility and relevance. Our observation from the results is that participants used relevance of the document as a primary criterion for ranking, and intelligibility was used as a secondary criterion. Accordingly, these results have implications for the evaluation and design of search engine algorithms. Search algorithms may consider matching and ranking documents written in closely related languages when resource scarce languages are used; the evaluation of such systems should also include intelligibility metrics or heuristics. Incorporating intelligibility as a criterion for ranking depends on accurately measuring intelligibility of languages. Methods developed by the computational linguistics community may be used to provide features of intelligibility in ranking (Gooskens and Swarte, 2017; Jens et al., 2007). Chapter 6 investigates intelligibility prediction and ranking using topical relevance features and intelligibility features.

Users may struggle to understand search results written in related languages as they are expected to read in unfamiliar languages. Therefore, we also explored the different issues associated with search difficulty and frustration, to understand user behaviour in the context of interacting with search results written in related languages. Previous work has associated search task difficulty with negative emotions (Arapakis, Jose, and Gray, 2008). We investigated what emotions are associated with interacting with search results in unfamiliar related languages. Our results show that users experience mixed emotions in response to this stimulus. Some of the participants were surprised that they could understand the contents of documents written in an unfamiliar language and had a positive search experience. Such participants reported to have experienced positive emotions. Other participants were frustrated and experienced negative emotions, which negatively affected their search experience – their experience was associated with negative emotions and struggled to complete the task. These opposite experiences made some of the tasks that were seemingly frustrating to not be affected by emotions. There are several interpretations for having these

results. Participants who reported positive emotions after intercomprehension may have struggled and experienced frustration while doing the task. Perhaps positive emotions may have been experienced only after the task was completed as a feeling of relief. Participants who reported negative motions and bad experience may have felt bad after not being able to complete the tasks. Therefore, the findings of this task may not be necessarily in contradiction with earlier findings related to task difficulty (Arapakis, Jose, and Gray, 2008) and writing style simplicity (Lopatovska and Mokros, 2008). Our approach to measure user behaviour through introspection may have contributed to this problem. Alternative methods such as using psycho-physiological methods that measure users' physiological responses could have been used in combination with self-reported methods to capture user emotions while conducting the study. Moreover, we did not investigate prior emotional state of participants before completing the tasks as previous work has reported that users with negative prior moods may perceive higher level of difficulty for a task (Xu, Zhou, and Gadiraju, 2019).

There are some limitations that may possibly have some effect on our results. The number of participants in the study was limited – there were few participants in each language group. We also explored a few languages with not much variation. The study did not include speakers of Luganda, Citonga and Cisena. It was also difficult to find participants with no knowledge of some of the languages used in the study. In particular, many of the Citumbuka speakers are multilingual and they speak Chichewa. It could be interesting to conduct a similar study involving speakers of many more closely related languages with a bigger sample size and to use log-analysis with metrics such as result skipping behaviour and click-through rate. Other interesting aspects to be studied would be exit points in terms of intelligibility, to explicitly determine what threshold of intelligibility or what languages would provide successful intercomprehension in retrieval.

Our findings on emotions experienced while interacting with search results in related languages have implications on how the use of intercomprehension should be incorporated in search engines. To avoid users being overwhelmed with search results written in related languages, and to assist users navigate such search results with less effort, enhanced features need to be added to search engine interfaces. We employed a User Centered Design (UCD) approach to find appropriate presentation layout of search results written in related languages, as well as additional features to be used when submitting queries. Our findings show that users prefer simple and familiar layouts that provide them with some flexibility in visualising the results, as well as additional language information to be added with the result. These findings are similar to observations made in previous work that used UCD to design user interfaces for CLIR (Petrelli et al., 2006a). There are several ways of explaining why users prefer familiar search results presentation layouts. Users may want to focus on completing their tasks with less navigation and effort. Familiar and simple interfaces help them achieve this with less cognitive load contributed by the search engine interface

itself. Similarly, participants' preference of language tools and associated features show that participants wanted to be in control of the search process and wanted to avoid surprises of languages by expecting the search engine to provide language information including search query and result language identification.

From a practical perspective, our findings suggest that information seekers may benefit from relevant content written in closely related languages. This would particularly be useful in countries or regions that have many closely related languages. For example, Chichewa is spoken widely in Malawi, and its dialect is spoken in Zambia. Relevant content generated in these countries may be used to meet information needs of users from the different countries. Providing users with retrieval results written in related languages does not ignore efforts to create resources and tools for all languages, but only offers a solution in the current resource constrained setting. Our research provides means for promoting users' language awareness but may appear to limit linguistic rights by offering results in related languages. Equally important, inexpensive methods to generate content and retrieval methods to make that information accessible equally need to be investigated.

5.4 Summary

The chapter presented user perspectives to understand how users would interact with search results written in related languages. We also discussed experimental results. We investigated the interaction behaviour of users who are presented with search results written in related languages of the query language. The results indicate that such results are useful to the user, and users are able to complete their search tasks, especially for simple tasks. Users preference of ranking was based on relevance as a primary criterion – intelligibility was a secondary criterion and, surprisingly, users were not interested in the ranking of results that were not relevant. Users struggled to complete search tasks when intelligibility was lower. Emotional experience of users moved from the positive to negative polarity as task difficulty increased and intelligibility decreased. These results have the implication that intelligibility should be introduced into search systems with caution. We further investigated what interface features should be added to such systems to assist users to interact with results effectively. We found that users prefer simple, flexible and familiar layouts, and features with some additional affordances that help them to reduce the cognitive load when interacting with the system – participants preferred features that would assist them to avoid being overwhelmed and surprised with the search results. Such affordances emerged to be explicit cues given with the results to provide some explanation for why the user is getting the result, as well as conventional layouts of search results presentation.

Chapter 6

Ranking for Relevance and Intelligibility

Ranking is one of the primary tasks in retrieval – documents are ranked in the order of usefulness to the user given their information needs and contexts. A better ranking algorithm orders documents in such a way that it helps users to find useful information effectively, i.e., more useful documents are ranked higher than those documents that are less useful. A retrieval system that presents users with documents written in related languages needs to present users with documents that are highly relevant and comprehensible to them. We propose re-ranking of search results based on traditional relevance features to estimate topical relevance and intelligibility features for intelligibility estimation.

In this chapter, we consider ranking of search results written in related languages using relevance and intelligibility features. We start by presenting an analysis of intelligibility features in Section 6.1: we report on feature selection from a set of intelligibility features and intelligibility prediction. We then provide our methodology in terms of our experimental design and implementation of the ranking models in Section 6.2. This is followed by a presentation of our experimental results obtained in the evaluation of the constructed ranking models in Section 6.3. Finally, we provide a discussion of the results, including implications and limitations of our approach in Section 6.4.

6.1 Intelligibility Feature Selection and Prediction

In this section, we report on two studies pertaining to intelligibility. We start with a presentation of work on selecting a subset of features to be used in our ranking task with relevance features. We then provide the study on intelligibility prediction.

6.1.1 Feature Selection

The use of intelligibility in ranking is a new approach and it is not yet clear in the linguistic community what the best features are to estimate intelligibility. In our work, several intelligibility features were extracted, including new features that have not been used in intelligibility studies before. We conducted feature selection to choose the optimal subset

of features that could give the best intelligibility prediction accuracy for five languages in group N cluster for Bantu languages investigated in our study. We combined the data from the linguistic intelligibility features with data from user intelligibility Web experiments discussed in Chapter 4. We then used four Random Forest (RF) trees based algorithms for the feature selection task.

The use of RF has become popular not only for prediction but also for feature selection because of their efficiency and robustness. We used the Boruta algorithm (Kursa and Rudnicki, 2010), permutation importance (Breiman et al., 1984a), gini importance (Breiman et al., 1984a) and conditional importance on unbiased conditional inference trees (Hothorn, Hornik, and Zeileis, 2006), to identify a set of relevant features to be used in our experiments.

Boruta algorithm: Boruta algorithm is a wrapper built around the Random Forest (RF) classification algorithm - selects features by iteratively removing features that are not statistically significant in a statistical test (Kursa and Rudnicki, 2010). The algorithm shuffles the given features to create shadow features and the best, average and minimum shadow features are selected. A random forest classifier is trained on a feature and its Z-score is compared with that of the best shadow feature. Features with higher Z scores are accepted as relevant features. Therefore, the output of the algorithm is a ranked list of all relevant features and non relevant features.

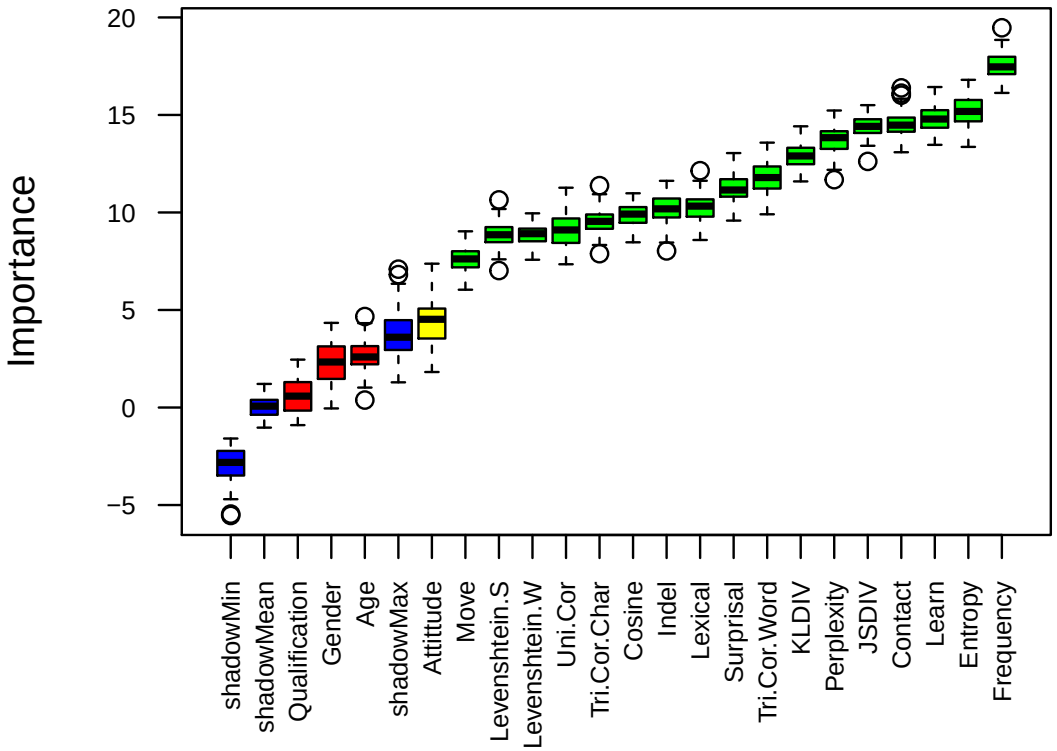


FIGURE 6.1: Boruta results plot for intelligibility features data. The green box plots represent relevant features while the red box plots are rejected features and the yellow boxplot represent tentative feature. The blue boxplots correspnd to the shadow features.

Boruta algorithm results plot (see Figure 6.1) shows that AGE, $QUALIFICATION$, and $GENDER$ are not relevant. This is not surprising as these features did not show any association with the target variable $SCORE$ in Pearson correlation (see Figure 4.9). Additionally, syntactic features ($INDEL$, MOV and TRI) were ranked lower by the algorithm and these features had low correlation with $SCORE$.

Gini Importance: We also evaluated the importance of features using gini importance (Breiman et al., 1984b) – total decrease in node impurity average over all trees in a forest. A higher gini mean decrease indicates that the feature has higher importance. Figure 6.2 shows the gini importance plot for our intelligibility dataset. The most important features to a classification model are highest in the plot and have higher mean decrease in gini scores. Gini importance produced results slightly different from the other two approaches – extra-linguistic features were ranked highly including, AGE and QUALIFICATION. This is not surprising because Gini importance is known to be biased towards variables with certain attributes, such as continuous variables or categorical variables with varying numbers of

choices (Strobl et al., 2007).

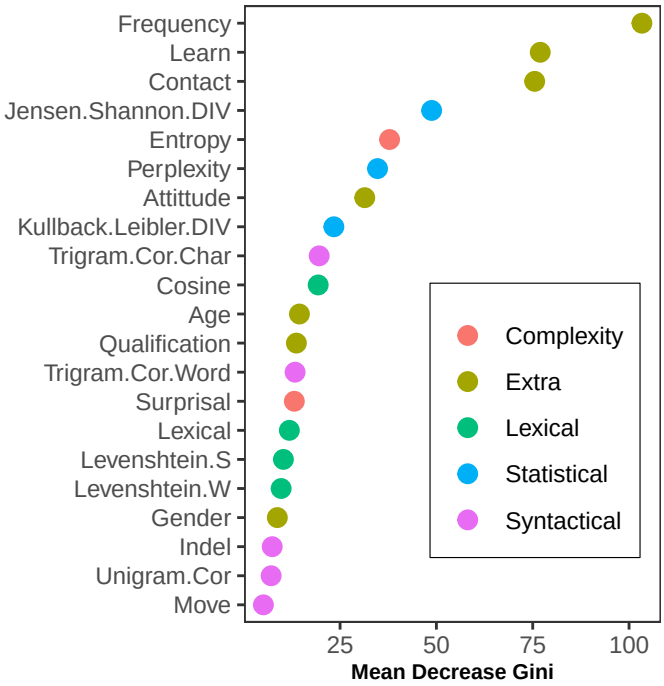


FIGURE 6.2: Feature importance as mean decrease gini. The colour of the dots represents the class of the feature.

Conditional Importance: Due to some of the features being of different types (binary, continuous and categorical) and highly correlated, we used other RF approaches to compare the results. We first used unbiased conditional inference trees (Hothorn, Hornik, and Zeileis, 2006; Strobl et al., 2008), a RF that uses feature conditional inference. The approach uses statistical significance testing to select variable and split points for a tree. At each decision point, the variables and the target are permuted, and a significant test is done. The variable with the lowest p-value is selected. The algorithm produces a ranked list of features with associated conditional importance scores. Figure 6.3 visualises the conditional importance of the features. The most important features have high scores.

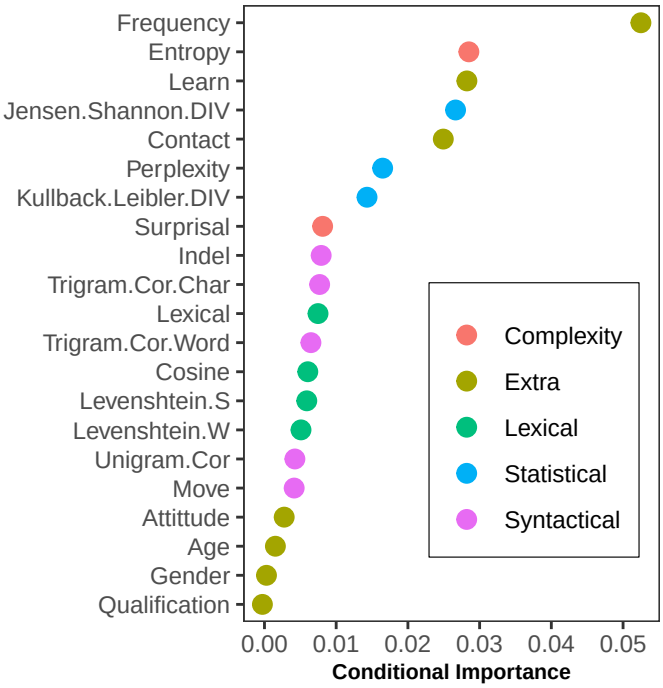


FIGURE 6.3: Plot of conditional feature importance for features in our dataset. The colour of the dots represents the class of the feature.

Similar to Boruta, unbiased RF algorithm ranked $GENDER$ and $QUALIFICATION$ at the lowest position. The rest of the features had different positions that were relatively similar to Boruta results.

Permutation Importance: We also measured feature importance using permutation importance. Permutation importance feature selection works by computing accuracy using the features and permuting each variable and measuring the accuracy. The variables with higher total mean decrease in accuracy are more important features. Figure 6.4 shows the plot of permutation importance for our intelligibility dataset.

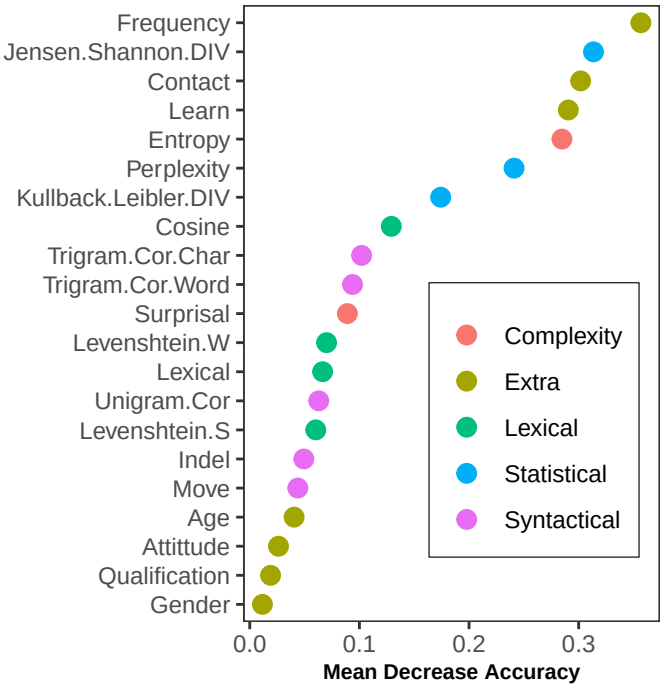


FIGURE 6.4: Plot of permutation feature importance for features in our dataset. The colour of the dots represent the class of the feature.

Permutation importance produced results similar to conditional importance and Boruta algorithm. Gini importance algorithm produced results slightly different from the rest of the approaches, i.e., gender, attitude and qualification were given the lowest ranks by the other algorithms while Gini importance ranked them slightly higher. The only consistent results are for the five top features, namely: frequency, learning, contact, entropy and Jensen-Shannon divergence. On average, gender, attitude and qualification were ranked lower and not correlated ($r_{AGE} = -0.02$, $r_{GEN} = -0.06$, $r_{QUA} = 0.03$) with the target feature (intelligibility score). Therefore, we proceeded by excluding age, qualification and gender features. The remaining features were explored further in intelligibility prediction using several classifiers in the proceeding section.

6.1.2 Intelligibility Prediction

Although the objective of our work is not prediction of intelligibility, we investigated the selected features for their prediction power to ensure that they had the ability to represent intelligibility in the ranking experiments. We formulated the prediction of intelligibility as a multi-class classification problem - a prediction model uses our intelligibility features to predict intelligibility classes; each class as the intelligibility score in the user text comprehension task (0,1,2,3,4) described in 4.5.3. The classification tasks were done using classifiers including Support Vector Machines (SVM), K-Nearest neighbour (KNN), Recurrent Neural

TABLE 6.1: Results for predicting intelligibility classes using Support Vector Machines (SVM), K-Nearest neighbour (KNN), Recurrent Neural Networks (RNN), Random Forest (RF), Naive Bayes (NB) and Logistic Regression (LR) on different subsets of the data: all the features, relevant features only, extra-linguistic features only and linguistic features only.

Classifier	**Feature Set**	**Accuracy**	**Kappa**
SVM	All	0.696	0.565
	Relevant	**0.716**	**0.595**
	Linguistic only	0.706	0.569
	Extra linguistic only	0.657	0.516
KNN	All	0.756	0.617
	Relevant	0.83	0.804
	Linguistic	0.73	**0.933**
	Extra linguistic	**0.853**	0.804
RNN	All	0.927	0.884
	Relevant	0.927	0.829
	Linguistic	0.829	0.756
	Extra linguistic	**0.927**	**0.893**
RF	All	0.706	0.58
	Relevant	0.716	0.593
	Linguistic	0.706	0.569
	Extra linguistic	**0.735**	**0.622**
LR	All	0.637	0.494
	Relevant	**0.716**	**0.596**
	Linguistic	0.706	0.568
	Extra linguistic	0.696	0.5667
NB	All	0.529	0.375
	Relevant	0.539	0.389
	Linguistic	**0.569**	**0.431**
	Extra linguistic	0.402	0.252

Networks (RNN), Random Forest (RF), Naive Bayes (NB) and Logistic Regression (LR). We used linguistic features only, extra-linguistic features, relevant features (relevant features produced by Boruta algorithm) and all features, to observe the effect of the different types of features. Extra-linguistic and linguistic features described in Section 4.5.3 are used to predict corresponding intelligibility scores (classes) from Web intelligibility experiments (the extra-linguistic features are also derived from these experiments). Our class prediction was evaluated using accuracy, precision, recall, F1 and Kappa. Table 6.1 shows the prediction results produced by different classifiers. The results show that the prediction using relevant features had better results than the other subsets of data across classifiers. ANN and KNN seem to give better classification results.

To examine the effect of the dataset size on performance, we investigated SVM and RF classifiers with different sizes of the dataset. Figure 6.5 shows the accuracy of the models

using different sizes of the dataset. The graph shows that performance of the classifiers increased with the increased size of the dataset. However, the performance was not consistent for the dataset size less than 60%.

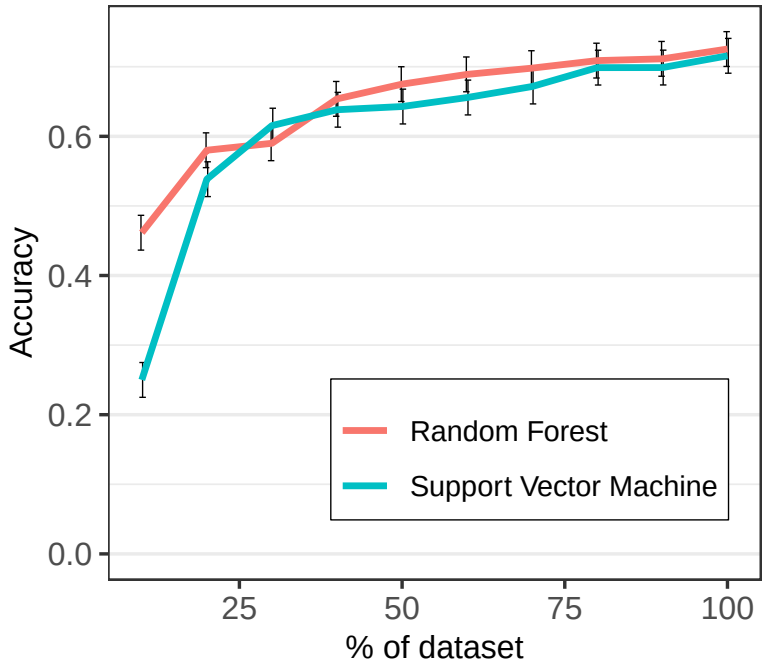


FIGURE 6.5: Accuracy of two classifiers, Support Vector Machine (SVM) and Random Forest (RF), with different sizes of the dataset. The error bars represent Standard Error of the Mean (SEM).

We found that there were major differences in performance for the specific classes. For example, class four had the best performance followed with class 3 while class 2 had the worst performance (see Table 6.2). The table shows that some of the classes had bad performance, below 0.5. To illustrate the extent of the differences in performance for the classes, we

TABLE 6.2: Results of RF classifier using relevant features only at intelligibility class level as well as at macro level.

Class	**Precision**	**Recall**	**F1**	**Class Error**
0	0.25	0.1667	0.2	0.588
1	0.667	0.625	0.645	0.25
2	0.333	0.375	0.353	0.92
3	0.769	0.667	0.714	0.273
4	0.8333	0.952	0.889	0.113
Overall	**Macro Precision**	**Macro Recall**	**Macro F1**	
	0.571	0.557	0.56	

performed an analysis of Receiver Operating Characteristic (ROC) curve. Figure 6.6 shows

the ROC curves for the classes based on RF classification. ROC analysis describe the performance of a classifier based on true and false positive rates and it is known to be independent of class distribution (Flach, 2003). The nearer a curve is to the upper left corner, the better the performance. The diagonal line divides the ROC space and the points in the line gives the classification performance of a random classifier. Points lying above the diagonal line show good results, i.e., performance better than a random classifier. The ROC curve shows that overall, our model has good performance although some classes have better performance.

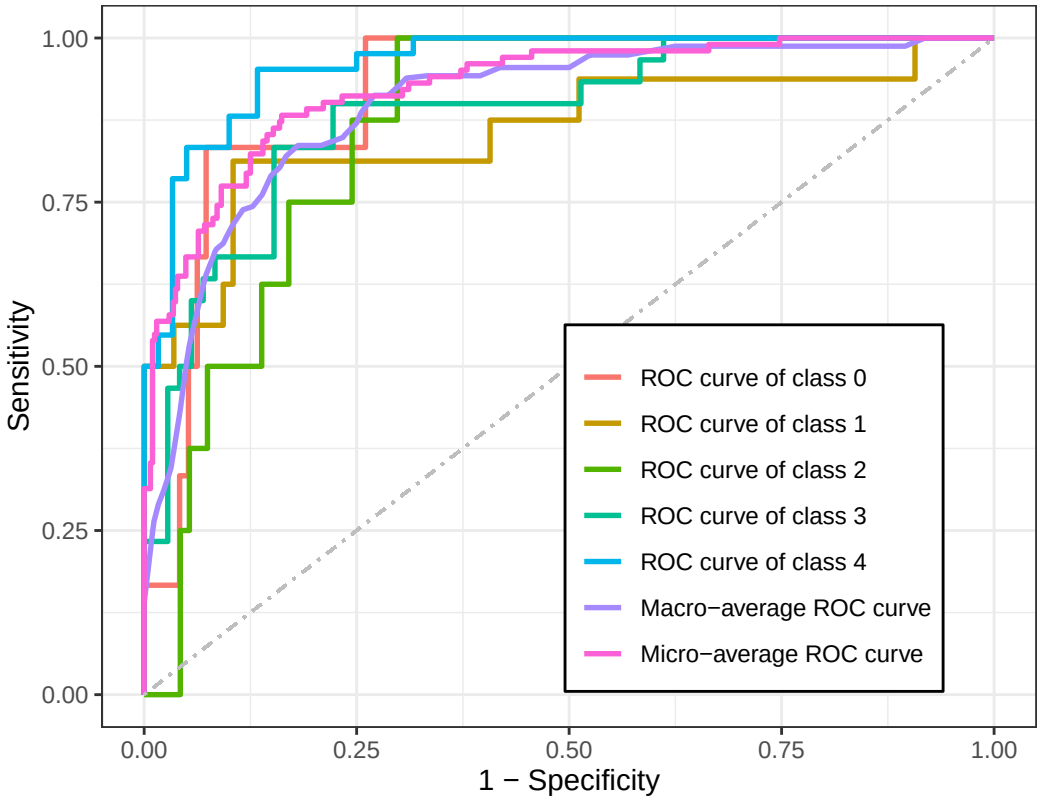


FIGURE 6.6: ROC curve evaluation for RF classifier with curves for each class, micro averaging and macro averaging.

Overall, using a subset of the features produced better results. To investigate further whether the size of the dataset affected the results, we experimented with increasing the data size in our dataset slowly to observe performance. Investigating the results per class showed that instances with intelligibility class of 4 produced better results, and the more the number of this value, the better the performance. We therefore concluded that the imbalances in class distribution in our data affected the results of the prediction.

The rest of the chapter reports on work on ranking using intelligibility features and relevance features. We first provide the approach for our study including the experimental design (Section 6.2). This is followed by results of our study in Section 6.3. We then provide a discussion of our results in Section 6.4.

6.2 Ranking Methodology

Our proposed re-ranking approach uses traditional topical relevance features and intelligibility features to improve retrieval quality of search results. The purpose of using additional features – intelligibility features – in our study is to ensure that relevant documents are ranked based on user preference of languages. Imagine a user, Tamandani, who is a retailer trader looking for information on tax on imported goods. She does not have a specific information need but would like to gather some information on custom duty before she decides to import goods. Hence, her information need is exploratory, and any information about tax on imported goods is relevant. If we assume that Tamandani is a monolingual speaker of Citumbuka, her preference would be Citumbuka documents over any other documents retrieved due to language similarities. The same behaviour would be expected if she was a monolingual Chichewa speaker. For example, Figure 6.7 shows two pages with five search results on 'tax on imported goods', i.e., A, B, C, D and E ranked differently for Citumbuka or Chichewa monolingual speakers. The left side illustrates the results for Citumbuka speakers and the right Chichewa monolingual speakers.

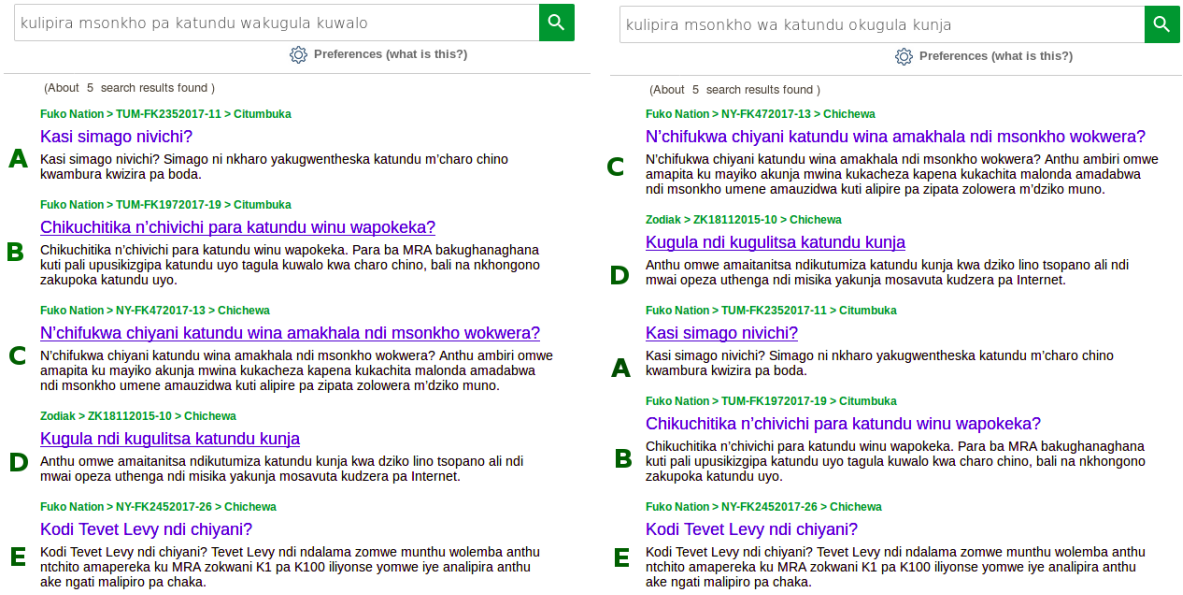


FIGURE 6.7: Ranking by relevance and intelligibility for a Citumbuka speaker on the left and Chichewa speaker on the right for the same search task

The results are ranked based on topical relevance and intelligibility. All the results are relevant except for document E, which cannot change its rank regardless of the user's first language. Although the other four documents are equally relevant, their ranks could change based on the language of the query. The assumption is that users would assign higher utility/relevance scores to useful or topically relevant documents written in their language.

Therefore, the gains for ranking topically relevant documents highly, which are more intelligible to the user are maximised especially.

Ranking documents retrieved on the basis of topical relevance and intelligibility needs to optimize both relevance and intelligibility. Our ranking problem is that of constructing a ranking model that finds the best combination of relevance and intelligibility features that matches with user ranking preferences, i.e., given lists or vectors of relevance and intercomprehension features of documents, our ranking function should map these features to ranks that matches the user ranking gold standard. This problem can be modelled as a supervised ranking problem, i.e., automatically constructing a ranking model using training data, i.e., examples of queries, associated document features and user preferred ranking, such that the model can rank documents for queries according to user preferences effectively. The ranking function attempts to rank documents effectively by minimising disagreements with user ranking preferences. In this regard, the proposed work attempts to answer the following research question:

RQ5 Does re-ranking of search results based on relevance and intelligibility improve retrieval effectiveness?

LTR is a supervised learning technique that requires training data. A typical training dataset consists of a set of queries, their associated documents and relevance grades or scores. Specifically, each query is associated with a set of features drawn from matching documents such as Term Frequency (TF), and how each document is related to the query in terms of the degree of relevance or ranking preference, or ranking order of documents with their associated features.

Suppose Q is the query set and D is the document set. Let each query $q_i \in Q$ be translated into a set of queries $q_i = q_{i1}, \ldots, q_{in}$. Suppose that these queries are submitted to several collections $C = \{c_1, c_2, c_3, \ldots, c_j, \ldots, c_n\}$ belonging to language set $L = \{l_1, l_2, l_3, \ldots, l_j, \ldots, l_n\}$, and a set of documents from collection j, $D_{ij} = d_{ij1}, \ldots, d_{ijn}$ are returned for query i on collection j where n is the number of languages or collections, and m is the number of returned documents. n and m are of fixed size, although the numbers may vary in reality. A ranking function needs to produce an optimal permutation π_i for the set of documents in D_i

$$D_i = \begin{bmatrix} d_{11} & d_{12} & d_{13} & \ldots & d_{1n} \\ d_{21} & d_{22} & d_{23} & \ldots & d_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & d_{m3} & \ldots & d_{mn} \end{bmatrix},$$

according to relevance and intercomprehension or intelligibility language features.

The learning task involves training a function to rank results using features estimating the probability of relevance of the document as well as how the involved languages relate

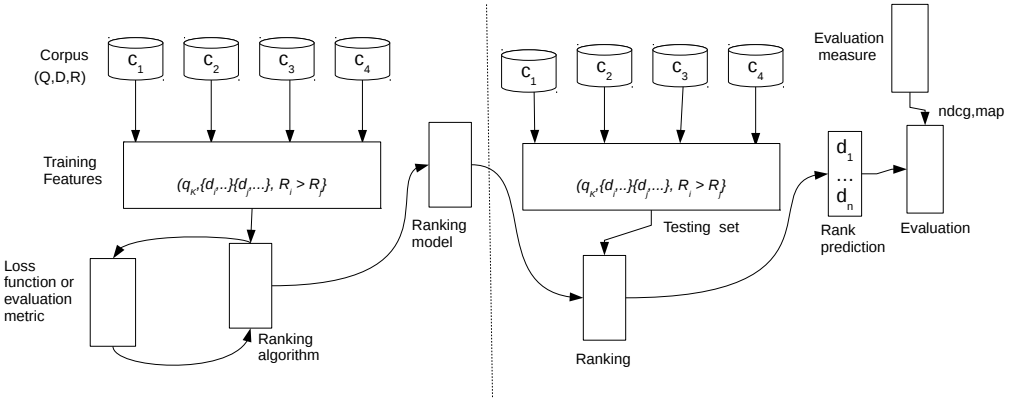


FIGURE 6.8: Proposed Learning Framework

in terms of intelligibility. Ranking is done by a scoring function F and produces an optimal permutation π_i for query q_i – ranking is based on permutations of the returned documents. $\prod_i$ denotes the set of all possible rankings on returned documents for query q_i. In essence, ranking is the task of selecting an optimal ranking $\pi_i \in \prod_i$.

6.2.1 Proposed LTR Approach

The aim of the LTR study is to learn a ranking function that improves quality of results by matching documents written in similar languages. The following sections describe how the proposed framework of learning in Figure 6.8 is implemented.

The ranking task aims to order documents based on relevance and intelligibility with respect to user ranking preferences. The training set for the experiments includes queries, documents, features and ranking preferences

- **Input Space:** The ranking function uses several features described in Chapter 4.
- **Output Space:** Ranking order prediction of documents.
- **Hypothesis Space:** LambdaMART algorithm is used to learn a ranking preference model and the algorithm uses regression trees for a learning model.
- **Loss Function:** LambdaMART uses gradient boosting to directly optimise a cost function such as Normalised Discounted Gain (NDCG).

LambdaMART is used because it has shown excellent performance in previous studies — an ensemble of LambdaMART won the Yahoo! Learning to Rank Challenge (Burges, 2010). Additionally, LambdaMART has been widely used and extended in other studies. For example, Papini and Diligenti (2012) used prior or expert knowledge in the form of First Order Logic

(FOL) knowledgebase with LambdaMART and obtained excellent results (Papini and Diligenti, 2012). Additionally, comparative studies on LTR algorithms have shown that LambdaMART is stable and produces consistent results on different benchmark datasets using different IR quality evaluation measures (Tax, Bockting, and Hiemstra, 2015). Moreover, LambdaMART has been extended to accommodate multiple objective function learning, which fits the proposed learning task (Svore, Volkovs, and Burges, 2011). LambdaMART uses an ensemble of regression trees with gradient boosting, i.e., MART is an instance of gradient boosting algorithm (Burges, 2010). LambdaMART algorithm combines the LambaRank and MART (Multiple Additive Regression Trees) algorithms (Burges, 2010). The algorithm uses boosted regression trees based on MART. IR quality evaluation measures are non-continuous and not easy to calculate gradients. LambdaMART is trained using Lambda – based on the idea that if d_i is more relevant than d_j, then d_i should be pushed upwards with lambda force λ_{ij} (Burges, 2010):

$$\lambda_{ij} = \frac{|\Delta Z_{ij}|}{1 + e^{s_i - s_j}}, \tag{6.1}$$

where s_i and s_j are ranking scores given by the current model for d_i, d_j respectively. $|\Delta Z_i j|$ is the difference in the quality evaluation measure after swapping rank positions of d_i and d_j, e.g, $\Delta NDCG$. Z is the chosen quality evaluation measure being optimised. Lambda calculations are based on Cross Entropy – a loss function used in RANKNET (Burges, 2010). The LambdaMART algorithm progresses as follows: the current tree calculates ranking scores and ranks documents using the scores. Lambda gradients are calculated based on the current tree scores and the quality measure being optimised. The principal idea is to find the Lambdas that will change the ranking scores to improve a quality evaluation measure. In the next iteration, the new rules are added to the tree. At the end, the trees are linearly combined to produce the final model. Figure 6.9 depicts how the LambdaMART algorithm works.

6.2.2 Experimental Design

The study investigated how intelligibility can be incorporated in matching and ranking documents written in several languages to improve the quality of results. LTR was used to train models that rank documents of related languages. Our experimental set-up used LTR as follows: i) LTR with relevance features only; ii) LTR with all relevance features and a single intelligibility feature; and iii) LTR with all proposed features for relevance and intelligibility. Models using relevance features only are used as the baselines for our approach – using relevance and additional intelligibility feature. All experiments used documents written in the five languages in Zone N, namely, Chichewa, Citumbuka, Chisena, Citonga and Cinyanja while Cinyanja, Citumbuka and Chichewa were used as query languages.

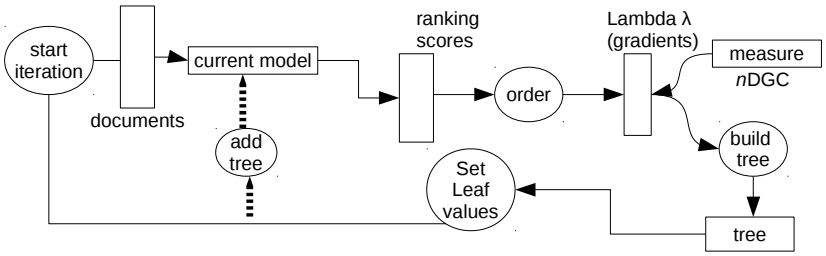


FIGURE 6.9: LambdaMART Ranking Algorithm (Burges, 2010)

Learning to Rank Experimental Set-Up

A pipeline of activities for LTR was done as follows: data preparation, feature selection, training and prediction.

- **Data preparation:** In the data preparation step, documents, queries and their associated features are prepared. Chapter 4 describes the process of gathering documents, creating queries, obtaining relevance judgements, feature extraction and feature engineering. Feature selection is done to select the final features to be used in the study.
- **Training:** RankLib's LambdaMART implementation was used in our experimentation. We used Five Fold Cross-Validation (CV) for training and testing. The dataset queries were randomly divided into five disjoint sets or folds: $F1$, $F2$, $F3$, $F4$ and $F5$. For each approach investigated, four folds were used for training and one fold for testing. Thus, we had five models for each approach. For example, the approach using all the features was firstly trained on data made up four folds $F2$, $F3$, $F4$ and $F5$ and tested on $F1$. This was done until all the folds were used as test data. Table 6.3 shows the test and training set partitioning procedure used in our experiments. Each time the model was tested, its performance was noted. We report the average performance for the model.
- **Prediction:** The trained model is used on a test-set where quality evaluation measures are calculated using tools provided by the RankLib.

TABLE 6.3: Five Fold Cross Validation procedure used in our experiments

Run	Test set	Training set
1	$F1$	$F2$, $F3$, $F4$ and $F5$
2	$F2$	$F1$, $F3$, $F4$ and $F5$
3	$F3$	$F1$, $F2$, $F4$ and $F5$
4	$F4$	$F1$, $F2$, $F3$ and $F5$
5	$F5$	$F2$, $F3$, $F4$ and $F1$

Weighted Sum Experimental Set-Up

We have proposed the ranking of search results written in related languages with the assumption of intercomprehension using relevance and intelligibility features. One of the strategies to combine multiple objective functions is to sum the product of the objective functions with selected weights. The weight is chosen in accordance to relative importance of each objective function. Our user study investigating preference of ranking search results in Chapter 5 showed that relevance was used as a primary feature and intelligibility was used as a secondary criterion. We explored using unsupervised methods for combining multiple objectives using weighted linear combination. Our experiments used the sum of normalised BM25 and Cosine similarity scores between the language of the query and documents. These were multiplied with selected weights, i.e., $f(x) = w_1x_1 + w_2x_2$. The weights w_1 and w_2 satisfied the condition $w_1 + w_2 = 1$. We used Rank Order Centroid (ROC) weight method to calculate weighting coefficients for combining the BM25 and cosine similarity scores. ROC is used when the rank order of the true weights is the only known information about the weights (Barron and Barrett, 1996a). ROC weights are known to be the most accurate weights among methods using rank based approaches (Barron and Barrett, 1996b). We calculated the weights using (Barron and Barrett, 1996b):

$$w_j(ROC) = \frac{1}{n}\sum_{k=j}^{n}\frac{1}{r_k}, \tag{6.2}$$

where n is the number of weights, r is the rank and j is the weight being calculated for the position j $j = 1, \ldots, n$. Thus, the obtained weights were $w_1 = 0.75$ for relevance (i.e., normalised BM25 score) and $w_2 = 0.25$ for intelligibility (i.e., cosine similarity score). Our new document scoring function was $f(x) = 0.75x_1 + 0.25x_2$, where x_1 is the normalised BM25 score using Zero-One Linear method and x_2 is the cosine similarity score of equivalent documents written in the language of the document and the query. The new obtained scores were used to rank documents.

Additional Baselines

Previous studies have proposed strategies for aggregating multilingual search results such as using raw and normalised relevance similarity scores like BM25. Score normalisation is done since the scores produced for different language collections are corpus dependent and therefore, not comparable. The scores are normalised into the same range of comparable values. The most common normalisation method is linear combine, i.e., Zero-One Linear Method. The normalised score of document d_{ij} is given as follows:

$$s_{ij} = \frac{r_{ij} - min_{ij}}{max_{ij} - min_{ij}}, \tag{6.3}$$

where s_{ij} is the normalised score for document d_{ij}, and min_{ij} and max_{ij} are minimum and maximum scores respectively for documents written in language L_i. BM25 normalised score ranking model is used as a baseline for the evaluation of the ranking models.

6.3 Ranking Experimental Results

Users' preference for ranking of search results written in related languages may be to rank them based on relevance and their intelligibility with the users' native language – our user study results support this observation. We used LTR to learn a ranking function of documents as would be preferred by a monolingual user looking at documents in related languages. Using several features, including topical relevance and linguistic intelligibility features described in Chapter 4, we trained and tested models and evaluated them using nDCG. We used several baselines to evaluate our proposed approach namely: normalised BM25 using linear combination and LTR model using relevance features only. We also used an unsupervised approach that combines relevance and intelligibility features to produce a single ranking function – using Rank Order Centroid (ROC) weight method – to calculate weighting coefficients for combining BM25 and cosine similarity scores, and we used the new score as function to rank documents.

We proceed to present experimental results obtained in two ways. Firstly, we present comparisons of results based on single hold-out method, including a query level analysis of the results. Secondly, we present average scores for nDCG for the ranking methods that were explored. The reported scores are derived from average scores from runs of each fold used as a test set. This allows the comparison of the different models using a single value.

6.3.1 Model Level Evaluation

We further analysed the results of our models at model level. Our evaluation is based on the best performing models – stable models trained using one of the five runs from five-fold cross validation. nDCG values at rank 10 (nDCG@10) were used to evaluate the models against a model using relevance features only. We used this baseline to investigate the differences in performance due to the differences in approaches, especially with respect to the use of additional intelligibility features. Our baseline is used to benchmark the performance of the supervised model and is the ultimate baseline for our research. Supervised methods allows several features to be combined using training data to produce a model that performs better than using any of the features separately. Unsurprisingly, the improvements depend on the quality of the features being used as well as the amount of data that is used to train the model. Therefore, a model using relevance features only makes a good baseline for our research – improvements after adding intelligibility features may signal that the approach

TABLE 6.4: Comparison of performance of nDCG@10. Scores in bold are significant for paired t-test at $p < 0.1$.

Type	System	Score	% Increment
Baseline	Relevance only	0.8466	
Unsupervised	BM25	**0.7315**	▽ 15.74%
	BM25 Normalised	**0.6938**	▽ 22.2%
	Weighted Sum	**0.7999**	▽ 5.83%
Lexical	Cosine	0.8342	▽ 1.47%
	Lexical	0.8537	△ 0.84%
	Levenshtein(s)	0.8629	△ 1.93%
	Levenshtein(w)	0.8485	△ 0.23%
Complexity	Perplexity	**0.8673**	△ 2.45%
	Surprisal	0.857	△ 1.24%
	SL Divergence	0.8507	△ 0.49%
	KL Divergence	0.8482	△ 0.19%
	Entropy	0.8566	△ 1.19%
Syntactical	Word gram	0.8461	▽ 0.05%
	Move	0.8326	▽ 1.65%
	Indel	0.8505	△ 0.46%
	Character gram	**0.8651**	△ 2.18%
	Word tri-gram	0.8512	△ 0.55%
Final model	all	**0.8676**	△ 2.49%

is beneficial to retrieval involving related languages documents. We used LambdaMart to train models using relevance features as well as intelligibility features.

We performed an ANOVA study to test the omnibus null hypothesis that the 19 models are the same or equivalent based on nDCG@10. This was accepted with $((F(18, 1382) = 0.044\ p < 0.05)$. The system effect size for the ANOVA analysis was $Fhat^2 = 0.0007$ and the analysis achieved a power of 0.0675, indicating a very small effect size. The analysis shows that 1498 queries will be required to achieve a power of 0.8. These results are promising, even though not significant due to the 19 models being evaluated together – multiple testing is known to be conservative, especially for small data sets (Boytsov, Belova, and Westfall, 2013) – some of the results are significant for pairwise t-test at $p < 0.1$ (see Table 6.4))

Table 6.4 shows evaluation results of all the investigated models against the supervised model that used the relevance features only. The results show that there were slight improvements between the relevance features only model and some of those models using additional features based on the intelligibility of the language of the query and document. Two of the models had lower NDCG scores compared to that of the baseline, indicating that the additional features harmed retrieval effectiveness in this scenario.

Figure 6.10 shows a box plot of average nDCG@10 values for four models: relevance features only, all the relevance features, weighted sum and normalised BM25. The plot

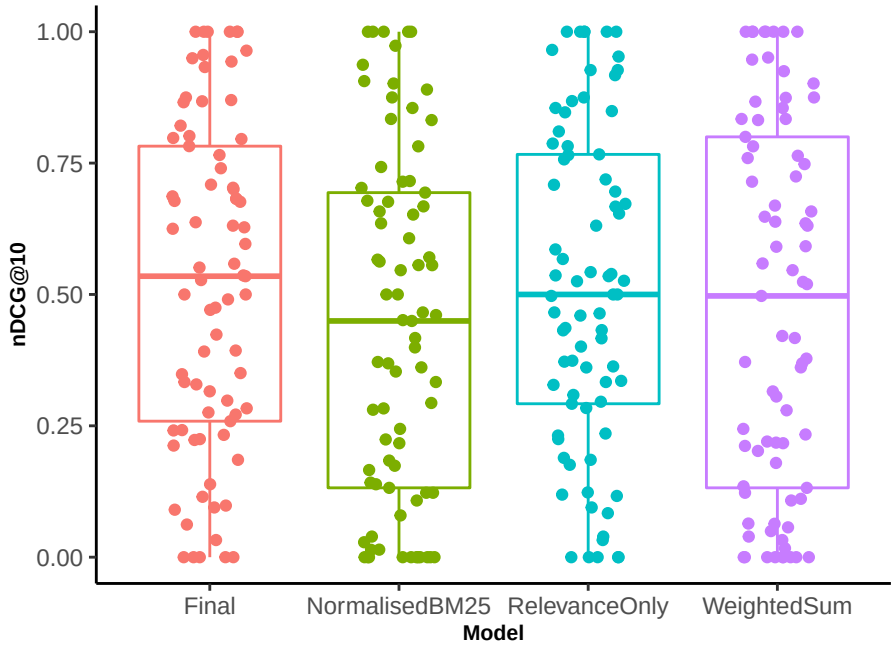


FIGURE 6.10: Boxplot of nDCG@10 scores for models using relevance features only, all the relevance features, weighted sum and normalised BM25.

shows that performance improved with additional features. The model using a weighted sum of normalised BM25 and cosine similarity scores (cosine similarity based on n-grams of equivalent documents of the language of the query and document) as a scoring function performed better than the one using normalised scores. The model using all the features performed slightly better than those using fewer features, i.e., using relevance features only.

Further, we investigated whether the observed differences were due to the additional features or out of chance, i.e., we tested if each of the models was identical to the baseline. The results show that the differences in performance between the baseline and most of the other models were not significant – the performance with the baseline was comparable. Only two models using additional intelligibility features – perplexity and character tri-gram correlation of frequencies had significant results ($\alpha = 0.05$). Similarly, the results of the model using all the features, i.e., relevance and all the investigated intelligibility features, and relevance only features were not significant, despite a slight improvement in NDCG scores ($+2.49\%$). The results suggest that the additional intelligibility features did not significantly improve the effectiveness of retrieval, but that it is likely that the two models have the same performance. This indicates that the model performance could be further improved by using more data for training and testing as well as using supervised approaches tailored to learn features with different preferences.

Overall, our findings suggest that adding intelligibility features to rank documents written in related languages has some potential to improve retrieval effectiveness. Using the weighted sum approach to combine cosine similarity (i.e, used as an intelligibility measure

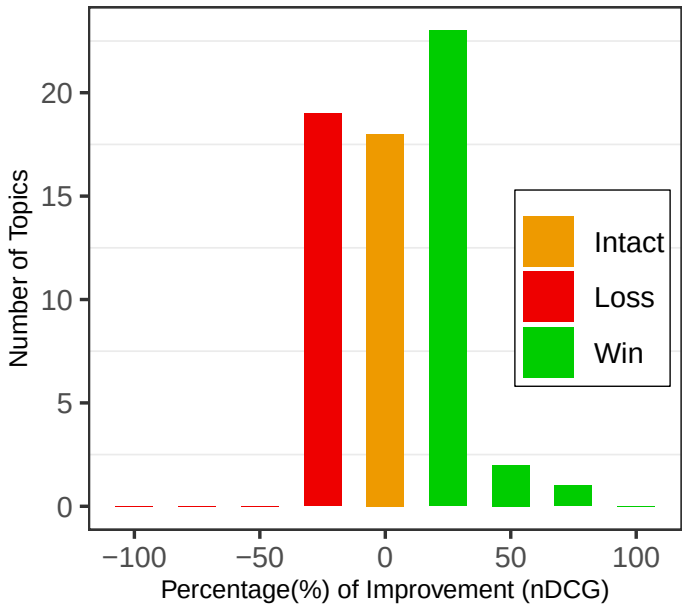


FIGURE 6.11: Plot of the number of queries that had their nDCG@10 improved, hurt or remained the same after adding intelligibility features to the ranking models.

between query and document language) and normalised BM25 scores outperformed the unsupervised baselines. Models using both intelligibility feature(s) and relevance features had slightly better performance than the baseline – model using relevance features only. However, most of the results were not significant for $\alpha = 0.05$.

We examined the differences between the performance of the the model using topical relevance features only and the model using all the additional intelligibility features at topic level to understand the interplay between topical relevance and intelligibility in our dataset. We found that from the test topics, the performance of 18 topics remained the same, 26 topics improved, and 19 were worsened. Figure 6.11 shows the distribution of topics in terms of their performance differences. Further investigation of the improved topics showed that improvements due to the consideration of intelligibility features were achieved when a more distant language had higher topical relevance score (e.g. BM25) but irrelevant and if a more closely related language had a lower topical relevance score. This shows that weighting topical relevance and intelligibility in this case improves the quality of results. Queries whose performance worsened were those with relevant documents in a related language but the topical relevance scores were very low, discounting the scores with intelligibility made the scores even lower and this promoted irrelevant documents. Using untranslated queries contributed to this and adding a method that improves matching across languages without translation may alleviate the problem.

6.3.2 Overall Ranking Performance Evaluation

Table 6.5 shows average nDCG results for retrieval using: relevance only and relevance and intelligibility features. The models include relevance features only, models using relevance features and intelligibility features and scoring function using BM25, normalised BM25 and weighted sum of normalised BM25 and cosine similarity scores between the language of the document and the query. The reported values are averages of models based on the five-fold cross validation training data and test data.

TABLE 6.5: Average nDCG scores at differen ranks for models using five fold cross validation

Type	**Model**	**1**	**3**	**5**	**10**	**50**
Unsupervised	BM25	0.3434	0.3659	0.3937	0.4618	0.6101
	Normalised BM25	0.376	0.412	0.438	0.487	0.6317
	Weighted Sum	0.453	0.4651	0.486	0.5359	0.6535
Baseline	Relevance Only	0.5511	0.5362	0.5601	0.6011	0.7755
Lexical	Cosine	0.5371	0.5591	0.5753	0.6095	0.7818
	Lexical Distance	0.5174	0.529	0.5725	0.6098	0.7817
	Levenshtein(s)	0.5481	0.5286	0.5763	0.6129	0.776
	Levenshtein(w)	0.5263	0.5362	**0.5838**	0.6071	0.7799
Complexity	Entropy	0.561	0.5435	0.5679	0.6159	0.7794
	KL Divergence	0.5611	0.5585	0.5755	0.6013	0.7837
	Perplexity	0.5525	0.5498	0.5649	0.6096	0.7825
	SL Divergence	**0.5991**	0.5549	0.5817	0.6016	0.7833
	Surprisal	0.5116	0.5442	0.5746	0.5996	0.7862
Syntactic	Indel	0.5136	0.5478	0.5638	0.617	0.7924
	Move	0.5506	**0.5599**	0.5781	0.6058	0.7849
	Charactergram	0.5635	0.5451	0.5766	0.6132	**0.7902**
	Wordgram	0.5541	0.5423	0.5679	**0.6206**	0.779
	Wordtrigram	0.5442	0.5726	0.5728	0.6046	0.7796
All	Final	0.5721	0.5571	0.5685	0.6116	0.7819

The trend in performance shows that using relevance and intelligibility features generally had a positive impact on nDCG at different ranks. The performance of the model using weighted sum is better than the model using normalised BM25 scores with an average difference in scores of 0.05. The supervised models follow the same trend. The model using topical relevance features and all intelligibility features performed better with an average difference of 0.01 across the ranks. The differences tend to decrease as the rank number increases, indicating that performance of the models converges as rank increases. The bigger

differences in performance early in the ranks can improve the search experience of users using RSLs as they may find useful information early in the search session and therefore reducing their frustration.

6.4 Discussion

Speakers of resource scarce languages may struggle to find information in their own languages. We have proposed to match, retrieve and present users with search results written in related languages to assist them to find relevant content they are looking for. Search engines that present users with documents written in related languages need to rank such results appropriately to help users locate relevant information that will be useful to them. We have proposed to use both topical relevance and intelligibility features to rank the search results. The use of intelligibility in ranking models assumes that it is possible to estimate intelligibility given a set of features. We formulate the problem of intelligibility prediction as a classification problem: we discuss our results on intelligibility feature selection and prediction. We then provide a discussion on our results on ranking using intelligibility and topical relevance features.

6.4.1 Feature Selection and Prediction

Predicting intelligibility is a challenging problem – several factors that may determine intelligibility have been proposed in literature, namely: linguistic and extra-linguistic features. We examined feature importance of our features using four Random Forest (RF) based feature selection algorithms. We have found that age, qualification and gender were irrelevant features for intelligibility in our feature set. Our results show that extra-linguistic features such as learning the language, contact with the language and frequency of use have high predictive power for intelligibility. Linguistic features such Jensen Shannon divergence and entropy were among the top five features in all the feature analysis methods used. Features based on measuring character distribution in words or corpus were ranked higher than any of the other classes of features. Lexical features performed fairly good. However, syntactical features had the worst performance. Our results on feature selection suggest that cognacy may be a strong predictor of intelligibility among closely related languages.

Previous work on Indo-European languages reported similar results. Kürschner, Gooskens, and Bezooijen (2009) used regression and logistic regression on linguistic features on data from Danish speakers presented with Swedish words and found that Levenshtein distance, word length and differences in number of syllables had higher importance in predicting intelligibility. Similarly, Gooskens and Swarte (2017) investigated the relative importance of linguistic and extra-linguistic predictors of mutual intelligibility using regression analysis

for five Germanic languages (Danish, Dutch, English, German and Swedish). The study found that extra-linguistic factors were strong predictors but attitude had less effect. Linguistic distances such as lexical, phonetic and orthographic distances were found to be stronger predictors than syntactic factors. These studies did not use the additional complexity or differences in character distribution based metrics explored in our study.

Our intelligibility classification results are promising, and we have shown that it is possible to predict intelligibility automatically from linguistic features. Using Random Forest classifiers provided stable results with good prediction accuracy. However, the imbalances in terms of intelligibility classes in the dataset affected prediction performance at class level. The results obtained in this study are limited to the languages explored. First, our dataset is very small and the used features were calculated based on a very small dataset. Our analysis on the effect of dataset size on performance suggests that with more data, it might be possible in future to obtain better improved results. Finally, intelligibility may be dependent on the specific document that the user is presented with. Our work has focused on linguistic features and general socio-linguistic relationship dimensions for the studied languages. For intelligibility to be useful in retrieval, it is necessary to explore the thresholds of intelligibility that would allow effective communication with users.

6.4.2 Ranking with Relevance and Intelligibility

We have proposed to improve the quality of search results for resource constrained languages by matching and retrieving search results written in related languages and re-ranking them using relevance and intelligibility criteria. To investigate our thesis, we extracted features from documents and queries to estimate the similarity relationship between the document language and that of the query, and to estimate topicality relevance between the query and the document. Our intelligibility features used in the study consisted only linguistic features as the use of extra-linguistic features may require personalization. We trained and tested LTR models using these features. We also used normalised BM25 scores and weighted cosine similarity and normalised BM25 scores.

Our evaluations of the models show slight performance improvements in terms of nDCG for the models using relevance and intelligibility features. The unsupervised weighted sum approach provided good performance when benchmarked with other unsupervised methods. However, significance tests against the best performing baseline shows that the registered improvements are only significant for $(t - test, \alpha = 0.1)$. There are several ways of interpreting these results. The direct consequence of the results can be that the registered improvements are due to random effects because of higher values of p ($p > 0.05$), there is some probability that the systems are similar to the baseline and hence no improvements due to

intelligibility features. However, we propose to interpret the results within its experimental context. Firstly, there is still some evidence that the null hypothesis could be false since $p < 0.1$. Secondly, our dataset for our study was very small for a supervised task that involved several features representing two objectives, relevance and intelligibility, which may be competing at times. The small improvements seen so far provide some evidence that integrating intelligibility in re-ranking of search results written in related languages can improve retrieval effectiveness. The weighted approach that does not require training data has shown significant improvement over other unsupervised approaches. There are opportunities to improve the results, which is a scope beyond this thesis – the study may be replicated using well resourced languages at a large scale as a confirmatory study to investigate further the validity of our results.

Our user study provided evidence that users prefer documents to be ranked primarily by relevance and secondarily intelligibility. Using system oriented evaluation based on a test collection has strengthened this observation. Additionally, these findings indicate that by using both intelligibility and relevance features, it is possible to mimic user search preferences evident in search behaviour of users interacting with search results written in related languages. The implications of our results is that search engine designers can help speakers of under-resourced languages to struggle less by presenting them with relevant search results that are more comprehensible to the users. This means avoiding irrelevant documents as well as incomprehensible documents that are matched only because of similarity at the lexical level – similar words or surface forms from closely related languages are likely to be translation equivalents than distant languages or languages that are not in contact.

The outcome of the study also provides opportunities for content developers. Content that is readily available in digital form should be made available online to alleviate the problem of the digital divide in terms of content availability. However, this is not a call to stop developing content for all under resourced languages. New innovative methods should be used to create content for resource constrained languages such as using gamification (Holy et al., 2017). Additionally, presenting users with information written in other related languages may have a negative side. Language forms an identity of a people, providing search results in other related languages may seem likely to take away their identity and these socio-linguistics factors may also affect user interaction behaviour with such search results. Our approach is only proposed to overcome the current problem of limited data by providing alternative relevant search results that are readily available.

Although our results are promising, there are some limitations in our approach. First, using relevance judgements as proxies for ranking preference may have affected the results. While using this approach has been effective in other studies (Palotti et al., 2016), document assessment using preference judgements could be more successful (Carterette et al., 2008). Also, the use of intelligibility classes may have affected the pairwise comparison of

the LAMBDAMART algorithm – continuous scores may have an advantage. Additionally, our evaluation focused on whether integrating intelligibility features improved relevance based ranking, and did not explore whether this also improves document intelligibility. Finally, the features for languages and languages to be used in retrieval are static and have to be known in advance. This may not be possible on Web scale for all languages.

6.5 Summary

In this chapter, we have investigated the ranking of search results written in related languages using relevance and intelligibility features. Firstly, we investigated the problem of intelligibility prediction and feature relevance and ranking. We investigated the importance of our intelligibility features using Boruta algorithm, Gini importance, conditional importance and permutation importance in Section 6.1.1. We found that age, qualification and gender were irrelevant features, and features based on character distribution differences were stronger predictors of intelligibility. We then explored the problem of intelligibility prediction as a classification problem in Section 6.1.2. Our results were promising, but were limited with the amount of data available. Secondly, we presented our approach for the ranking study including the experimental settings in Section 6.2. This was followed by a presentation of results on our ranking approach using relevance and intelligibility features in Section 6.3. Our results for ranking with relevance and intelligibility features show slight improvements for nDCG for the supervised methods and significant improvements for the unsupervised methods. Although our results on ranking are promising, the amount of data and the nature of the assessment for relevance for our data affected the effectiveness of our ranking models. Finally, we discussed our results on intelligibility feature selection and ranking in Section 6.4. Overall, re-ranking of search results using relevance and intelligibility can assist searchers of resource scarce language to find documents that are both relevant and comprehensible to them.

Chapter 7

Multilingual Stemming

Bantu languages are known to have complex morphological processes such as inflection, derivation and compounding. These features are widely known to affect retrieval effectiveness in languages that have rich morphology (Hollink et al., 2004). Stemming maps morphological variants to a common form such as a stem or root, and hence improves the likelihood of query terms matching with the documents' indexed terms, which improves recall (Krovetz, 1993; Frakes, 1992). Approaches proposed for stemming range from linguistically motivated to language independent techniques. The latter uses language specific knowledge to create rules for stemming while the former does not require any language knowledge. Using language specific rules to create stemmers for several languages is expensive and time consuming. We propose a stemming approach that is based on affixes learnt from the corpus to stem words of closely related languages using rules derived from common morphological features of the family. We use system oriented approach to evaluate the impact of using the proposed stemming approach on retrieval effectiveness for Chichewa and Citumbuka queries and corpora.

In this chapter, we present our stemming approach and its evaluation in different retrieval settings such as monolingual, cross-lingual and multilingual retrieval. We first provide the design of the stemmers in Section 7.1, including our supervised learning approach for learning affixes from corpora and the stemming algorithm. We then outline the experimental settings for the evaluation of the proposed approach in Section **??**. This is followed by experimental results in Section 7.2. Finally, we provide a discussion of the obtained search results in Section 7.3.

7.1 Stemming Approach

Stemming is a preprocessing step traditionally applied to document terms before they are indexed, and to query terms before matching with index terms in the retrieval step. Stemming is done to improve recall, as morphological variation of words may cause query terms and document terms to mismatch. Stemming reduces morphological variants to a common form, which makes matching and retrieval possible using morphologically related words

with equivalent meaning. This is very important for improvement of retrieval effectiveness for morphologically rich languages, i.e., languages that have morphologically productive processes, capable of generating hundreds of different forms of a root or stem.

Stemming may improve retrieval involving related languages where untranslated terms are used because reducing words to stems or common forms, may make terms from two related languages to be similar, and therefore, may make retrieval across languages possible. Several techniques have been proposed for stemming, from language specific approaches to language independent approaches. Language specific stemmers incorporate language specific morphological changes for words in a language. Morpho–phonological rules are used to capture how words change. Irregular forms of words that are not accounted for by the specified rules are given in a separate list to spare them from stemming. Stemming is widely known to affect retrieval performance for both linguistically motivated approaches as well as language independent approaches (Hollink et al., 2004; Mcnamee, 2008). We propose a stemming approach that combines both aspects of the two major approaches to overcome the problem of language resources and limited language knowledge. The approach uses two stages, namely: (i) in the first stage, we use a character similarity based approach to learn affixes from corpora; (ii) in the second stage, we perform stemming using rules derived from a morphological template designed for Bantu words. In this section, we present the proposed approach. Firstly, in Section 7.1.1, we detail our proposed affix learning approach. Secondly, we present the proposed stemming approach including the proposed stemming algorithm, in Section 7.1.2.

7.1.1 Affix Learning

Using rules to stem words requires a set of known affixes to be removed. For languages not well documented, such information is not readily available. We use an unsupervised method to extract possible affixes from the corpus. Our affix learning approach is done in four steps, namely: (i) grouping morphological variants from a corpus into clusters; (ii) estimating the stem in each cluster; (iii) segmenting words in each cluster; and (iv) choosing affixes. Cluster induction is done with the Agglomerative Hierarchical Clustering (AHC) algorithm, using a weighted similarity measure on characters based on a normal distribution. Next, we present details of the steps for learning affixes from the corpus as follows:

1. **Cluster induction:** We used a string similarity method to group morphological variants from a corpus into clusters of words with the same stem.

2. **Stem boundary estimation:** For each cluster, the boundary for the stem and affix material is estimated.

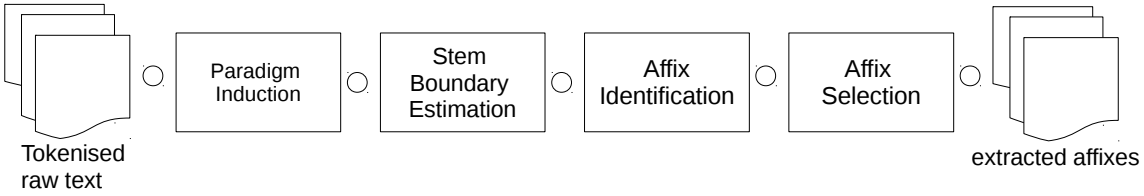


FIGURE 7.1: The proposed approach has four parts: (1) assigning words into morphological groups; (2) determining the stem of a word; (3) finding boundary the affixal segments; and (4) finding and selecting affixes of words

3. **Affix Identification:** The identified affixal section of each word is segmented into possible affixes.

4. **Affix selection:** Affixes are chosen from the pool of affixes.

Cluster Induction

Unsupervised methods have been widely used for languages that are not yet well documented (Hammarström and Borin, 2011). We use the string similarity method to group morphologically similar words into clusters. We propose a string similarity method that accommodates the common structure of Bantu words. The proposed general structural view of Bantu words consist of the prefix, stem and suffix. Using sequences of patterns, such as n-grams, as features for word similarity, the stem is always part of the internal morphemes. String similarity metrics that use character n-grams as features for similarity, such as dice coefficient, do not consider the position of the patterns in the compared strings. This may lead to high similarity values for words with similar affixes, and not similar stems. We propose a weighted string similarity measure that assigns higher weights to character sequences that are likely to be roots, with the assumption that roots provide the core meaning of a word. Tri-grams are used because the common Bantu root structure is *-CVC-* { *-fun-*} for verbs and have been shown to provide better results for Bantu words (Chavula and Suleman, 2016). We propose a weighting that is based on Normal distribution; the weights are assigned to similar character sequences based on the position of the pattern in a word and calculated using the normally distributed Ordered Weighted Aggregation (OWA) operator.

An OWA Operator is a collection of operators proposed by Yager (1988) for aggregating information, usually from multiple conflicting criteria. An OWA operator of dimension n is a mapping, OWA:$R_n \rightarrow R$, with an n vector w = $(w_1, w_2, ..., w_n)^T$ such that w$_i \in [0, 1]$ and $\sum_{i=1}^{n} w_i$= 1.
A collection of n aggregated arguments or objects $a_1, a_2, .., a_n$ takes a form of n preferences

provided by n different individuals , criteria or objects. The OWA averaging is performed as follows:

$$OWA_w(a_1, a_2, ..., a_n) = \sum_{j=1}^{n} w_j b_j \tag{7.1}$$

where b_j is the j_{th} largest element of the collection of the aggregate objects. The primary challenge to OWA aggregation is to determine the weights (see Yager, 1988).

Xu (2005) proposed a normal distribution-based method for calculating OWA weights. A normal distribution based OWA weighting assigns low weights to preferences or values away from the central value. This is analogous to calculating the similarity metrics of two words, where the stem or morphemes are highly likely to be internal morphemes. Therefore, a normal distribution-based OWA can be applied to determine similarity of character patterns found in any two terms t_1 and t_2. Both t_1 and t_2 can be divided into 3-grams to generate vectors of length $|t_1|$-2 and $|t_2|$-2. The longer vector is used to specify the value of n, which is the dimension of the OWA vector. The weights in the OWA vector are used to calculate the overall similarity metric. $a_1, a_2, .., a_n$ objects corresponds to tri-gram vector of the longest string. The value b_i is generated from a similarity vector consisting of 0's and 1's for positions with no matching patterns and similar patterns respectively.

The weights for character sequence patterns based on tri-grams is estimated using OWA weights based on normal distribution (see Xu, 2005 for the formulations and proofs). The formulation for calculating the weights are as follows (Xu, 2005):

$$w_i = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-[(i-\mu_n)^2/2\sigma^2 n]} i = 1, 2, ..., n \tag{7.2}$$

$$\mu_n = \frac{1+n}{2} \tag{7.3}$$

$$\sigma_n = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(i-\mu_n)^2} \tag{7.4}$$

The proposed weighted similarity measure is based on weighting tri-gram by their position in a word using a method for estimating weights for an OWA operator. The mean and variance of the used normal distribution depends on the length of words being compared, e.g., the mean is the centre of the word given by the position of the middle tri-gram.

Example for Calculating String Similarity Scores: Our proposed approach is compared with Dice Coefficient in terms of cluster formation. We provide here two examples for calculating the two types of scores. Suppose we want to compare two Chichewa words with the same root, namely: 'muyende' and 'timayendetsedwa'. Also, suppose we want to compare two other words with different stems but sharing affixes, namely: 'kutipitisa' and 'kutiphikitsa'. The steps for calculating scores using the proposed approach are as follows:

1. The first step is to generate tri-grams for the two pairs of words. Refer to Table 7.1 for the tri-grams.

2. The second step is to find the length ($long$) of the longest string – i.e., 13 for our first example and 10 for the second example.

3. The third step is to find the weights (owa) for n = 13 and n = 10. Refer to Table 7.1.

4. The last step is to find common (com) tri-grams and add together their weights, i.e., total score = $0.1006 + 0.112 + 0.1161$, which is 0.329 for the first example, and for the second example, 0.3358 ($0.0443 + 0.0719 + 0.1034 + 0.0443 + 0.0719$).

TABLE 7.1: List of tri-grams and their weights

Example 1													
muyende			muy	uye	yen	end	nde						
timayendetsedwa	tim	ima	may	aye	yen	end	nde	det	ets	tse	sed	edw	dwa
weights	0.0321	0.0475	0.0656	0.0842	0.1006	0.1120	0.1161	0.1120	0.1006	0.0842	0.0656	0.0475	0.0321
Example 2													
kutipitisa	kut	uti	tip	ipi	pit	iti	tit		its	tsa			
kutiphikitsa	kut	uti	tip	iph	phi	hik	iki	kit	its	tsa			
weights	0.0443	0.0719	0.1034	0.1317	0.1487	0.1487	0.1317	0.1034	0.0719	0.0443			

For the dice coefficient approach, the steps for calculating the score ($2C/|X| + |Y|$, where C is the number of common tri-grams and $|X|$ and $|Y|$ are the number of tri-grams for strings X and Y), is straightforward. We first count the number of common (com) tri-grams, namely: 3 and 5 for the first and second example, respectively. We then count the number of tri-grams for each of the two words and add them together (5 + 13) and (10 + 9). Finally, we multiply by two the number of common tri-grams and divide it with the sum found in the previous step , i.e., 6 / 18, which is 0.333 and 10/19 = 0.526. As shown the proposed approach gives more weights to tri-grams inside the word than those in the boundary. Our assumption is that common tri-grams on the boundary of the words are likely to be affixes and internal tri-grams are likely to be part of the root. For our two examples, we end up with similar scores for the first example using dice coefficient and the weighted OWA approach (0.33 and 0.329) (see Table 7.1 for details of the example). In the second example, where the words have different roots, we have very different scores, 0.526 and 0.3358. The dice

function CLUSTER INDUCTION($threshold$)
  while C is not empty **do**
    Select w from C
    C = C - w
    cl = $\{w\}$
    for each V in C **do**
      Find $long$, com, owa
      if $owa > threshold$ **then**
        cl = $cl \cup \{V\}$
        C = C - V
      end if
    end for
  end while
end function

ALGORITHM 1: Morphological Cluster Induction

coefficient approach erroneously portrays the two words in the second example as being more similar, which is not the case. For tasks that group words with the same stem together, the proposed approach is likely to provide better clusters. We provide the evaluation of clusters using the two methods in Section 7.2.1.

Grouping words into morphological clusters is done twice to obtain clusters with high purity, where purity is a measure of how similar members of the cluster are. We used two algorithms in the clustering step, namely: (i) a simple algorithm given in Algorithm 1 to create initial clusters, and (ii) an HAC algorithm to group words into smaller clusters. HAC is resource hungry and was used on pre-grouped smaller clusters generated in the initial step. HAC also requires ad-hoc selection of parameters and thresholds (Hammarström and Borin, 2011), and this worked better on a smaller number of words.

Simple Clustering: The initial task to assign words to morphological clusters is to prepare the raw text into words that can be processed. We first tokenise the corpus to obtain a list of words. Words with non-alphabetic characters such as numbers, as well as repeated words, are removed. Words in other languages such as English are also removed words. We then change the case of the words to lowercase. Clustering is considered an unsupervised learning approach since items are assigned to a class without prior knowledge of membership. For each word in the corpus, we first check if it has been clustered already. If the word is not assigned to a class, we compare it with every word that has not been clustered. The word is assigned to a particular cluster if its string similarity is above a certain threshold with a word that it was compared with. The algorithm for the clustering is given in Algorithm 1.

Hierarchical clustering: The second stage clustering uses Hierarchical clustering. Hierarchical clustering is achieved using two approaches, namely, top-down and bottom-up

(Jain, Murty, and Flynn, 1999). Bottom-up clustering is called Hierarchical Agglomerative Clustering (HAC). HAC starts with clusters of a single member and similar clusters are merged recursively until all the clusters are merged to a single cluster. The pair of clusters to merge is identified by a linkage method based on a dissimilarity measure or distance. Common linkage methods include Complete-linkage, average-linkage, wards method and a single link (Jain, Murty, and Flynn, 1999). Performing a similar task, Majumder et al. (2007) conducted analysis on Hindi data and found that the average-linkage was a better method for grouping morphologically similar words together. We chose the number of clusters using cophenetic correlation coefficient, i.e., a cluster quality measure, and the threshold with the best cophenetic correlation coefficient was chosen.

Stem boundary estimation

Morphological clusters are used to learn the affixes found in the corpus. Initially, the root of the words in a given cluster is determined by finding the common sub-strings for all word pairs in the cluster. The longest most frequent sub-string is chosen as a likely root in the cluster. An algorithm tailored for Bantu words is used to find affixes and morpheme boundaries of all words in the cluster. The chosen root is used to divide a word into three segments: prefix, stem and suffix.

Affix Identification

An algorithm that learns the affixes in both the prefix and suffix segments is used to segment each word into its possible morphemes. The algorithm uses the following basic knowledge for Bantu words: (1) prefixes are open syllable *(V,CV,CCV,CCCV,CCCCV)*; and (2) morphemes in the suffix segment are divided from the last vowel to the first character, usually removing the last vowel when it is either 'a' or 'e'. The affixes learnt from all the words are used to generate possible affixes of the language using the Greenberg Principle (Hammarström and Borin, 2011).

Each word is assumed to have at most nine slots regardless of its part of speech. This is based on the Bantu verbal structure, which has the most slots compared with the other word categories. Our assumption is based on Nurse (2001) proposal of a simplified structure with the following nine slots as follows:

PRE-SM+SM+NEG2+TA+OM+ROOT+EXTENSION+FV+POSTFV

The prefixal component has five slots and the suffixal segment has three slots (final vowel, verb extension and extra suffixal morphemes), and the sixth slot is for the root. Additionally, each slot is assumed to be a monosyllabic morpheme except the root. The process for affix identification is divided into the following steps:

1. Determine the root of the words in a given cluster
2. Divide every word in a cluster into three segments: prefix, root and suffix.
3. Create Greenberg square for the prefix and suffix segment. Keep segments that are only in these squares.
4. Generate a list of prefixal morphemes by dividing the prefix segment on syllables from the leftmost end.
5. Generate a list of suffixal morphemes by splitting the suffix segment from the rightmost end using length and type of end characters. For example, if the segment is just one character long and it is a vowel, then add it as a final vowel.

Affix Selection

Since errors can occur while finding the root, the segments to use for affix identification are selected using Greenberg square principle. The Greenberg square principle stipulates the minimum requirements for specifying sequences of phonemes that are segmentable – for any two given roots and suffixes or prefixes, both roots must appear with at least two similar suffixes or prefixes (Hammarström and Borin, 2011). For example, suppose that there are four words X, Y, Z and W, the square principle for segmenting sequences of phonemes can be applied if: $X = AC$, $V = BC$, $Z = AD$, $W = BD$. Given that C and D are stems, then both stems have A and B as prefixal material, satisfying the principle. Explained differently, the Greenberg Principle stipulates that if a character sequence appears after or before at least two roots or stems then it is likely to be an affix. For example, a square is given by four words with w_1 (ti-yend-a), w_2 (mu-yend-e), w_3 (ti-pit-a) and w_4 (mu-pit-e) with the form stem$_i$+suffix$_x$, stem$_i$+suffix$_y$, stem$_j$+suffix$_x$ and stem$_j$+suffix$_y$, i.e., suffix$_x$ is -a and suffix$_y$ is -e. Since the two suffixes, -e and -a are appearing on both sets of words, then, based on the Greenberg principle, these two suffixes (e and a) become affixes. This also applies to prefixes. This rule is used to select prefixal and suffixal segments that are used in the generation of affixes.

We apply the Greenberg square principle as follows. The prefixal and suffixal segments are matched with those appearing in other clusters. If a segment appears in multiple clusters, and in those other clusters exist a prefixal segment similar to the ones found in the cluster that the segment is drawn from, then that segment is kept, otherwise, it is discarded. The prefixal segment is divided into prefixes from left to right on syllables. The suffixal segment is divided into three sections – extensions, vowel and extra suffixes. Figure 7.2 illustrates the algorithm for finding the affixes. The collection of the identified affixes from the corpus is used with the stemming algorithm in the retrieval experiments.

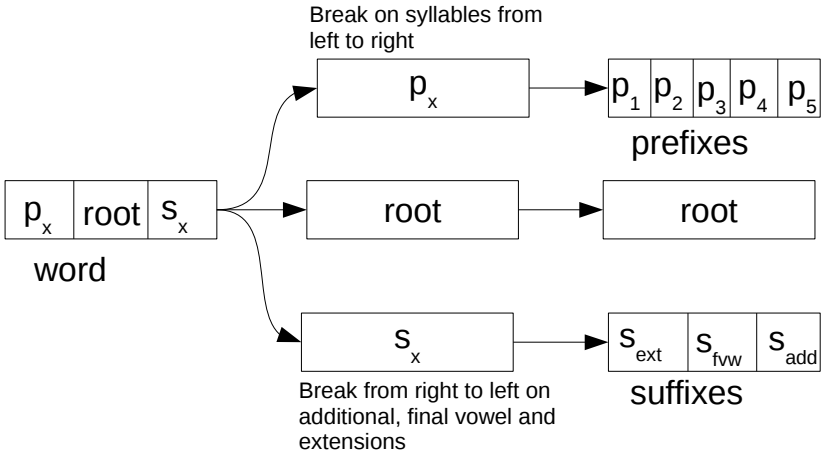


FIGURE 7.2: Finding Affixes

7.1.2 Stemmer Implementation

The stemming algorithm is divided into nine steps and at each step, a single affix is removed, except for the verbal extensions (affixes are removed recursively). The algorithm uses nine steps because we assume that a Bantu word has at-most nine slots including that of the root. The algorithm uses the number of syllables and affix stripping power to remove affixes. The number of syllables is based on the number of vowels in a word – thus – a shape of $[C]^*V$ is assumed for syllables. Stripping power is based on the number of syllables and the contagious verbal extension shapes at the end of the word together with final vowel. Therefore, stripping power is defined as follows: number of syllables - (number of verbal extensions + 1 (for final vowel) + 2 (for a minimum of the bisyllabic root of the form -CVC-). Formally:

$$p = s - ext - 3 \tag{7.5}$$

where $s = |v|$ in word w and $v \in a, e, i, o, u$, ext is the number of verbal affixes in w.

At each step the value of p is checked to see if it is greater or equal to one in order to take away an affix. If an affix is taken away, the value of p is reduced by 1.

The proposed stemming approach is implemented using Python2 for evaluation in an IR task. The evaluation task uses: i) language specific stemmers; ii) the proposed language family stemmer; iii)) a language independent stemming approach, based on character trigram; and v) no stemming.

The study adopts the Cranfield paradigm for IR evaluation – a test collection described in Chapter 4 is used as a Gold Standard in comparing the implemented stemmer with other stemming methods proposed in the literature. Evaluation is based on quality of retrieval – several quality evaluation metrics are calculated.

7.2 Evaluation of the Proposed Stemming Techniques

Our work investigates the use of a multilingual stemmer – a stemmer that works across related languages to strip affixes from multiple languages. The design of the stemmer is based on the word structure of Bantu languages, i.e., we assume that a Bantu word would have five affixal slots and the stemmer strips off those affixes while the number of syllables is greater and equal to two. The approach requires a list of possible affixes – affixes are both prefixes and suffixes. However, for many languages, this list of affixes is not readily available. We proposed an unsupervised method to learn affixes from text of a language using a statistical method that uses clustering to group morphological variants together.

We limited our study to two languages in group N languages, namely: Chichewa and Citumbuka, which have morphological structure similar to Bantu morphology with some minor variations. Therefore, the approach can be extended to other languages in Group N of Bantu languages. For our stemming experiments, we use affixes provided in literature for the two languages, i.e, (Mchombo, 2004; Chavula, 2016). We also use affixes learnt from the language using our approach for multilingual stemming described in Section 7.1.2.

7.2.1 Unsupervised Clustering Evaluation

We investigate how morphological clustering using two string similarity measures, i.e., dice similarity and OWA, would perform for the investigated languages. The two methods are evaluated in terms of the quality of clusters they would create.

Clustering uses observed patterns in data to group items together – the number of clusters is not usually known in advance. We use HAC, which deletes or merges together clusters above a threshold (a value set based on a certain metric and in our case dice or OWA) in order to obtain a certain number of clusters (Majumder et al., 2007). Thresholds affect the number of clusters and, therefore, the quality of the clusters. The number of thresholds can be empirically determined by plotting the number of clusters produced by a range of thresholds to obtain a region where the number of clusters tends to stabilise. Unfortunately, we were not able to observe this from our data because it is a small dataset. Figure 7.3 shows the relationship between threshold and number of clusters on our evaluation data. Therefore, our experiments used threshold values that produced clusters closer to the number of clusters created by human assessors.

We used purity to evaluate the quality of the clusters. Purity is an external metric that calculates the percent of objects or items that have been correctly classified and its values ranges 0 and 1. Purity is calculated as follows:

$$Purity = \frac{1}{N}\sum_{i=1}^{k} max_j \mid c_i \cap t_j \mid \tag{7.6}$$

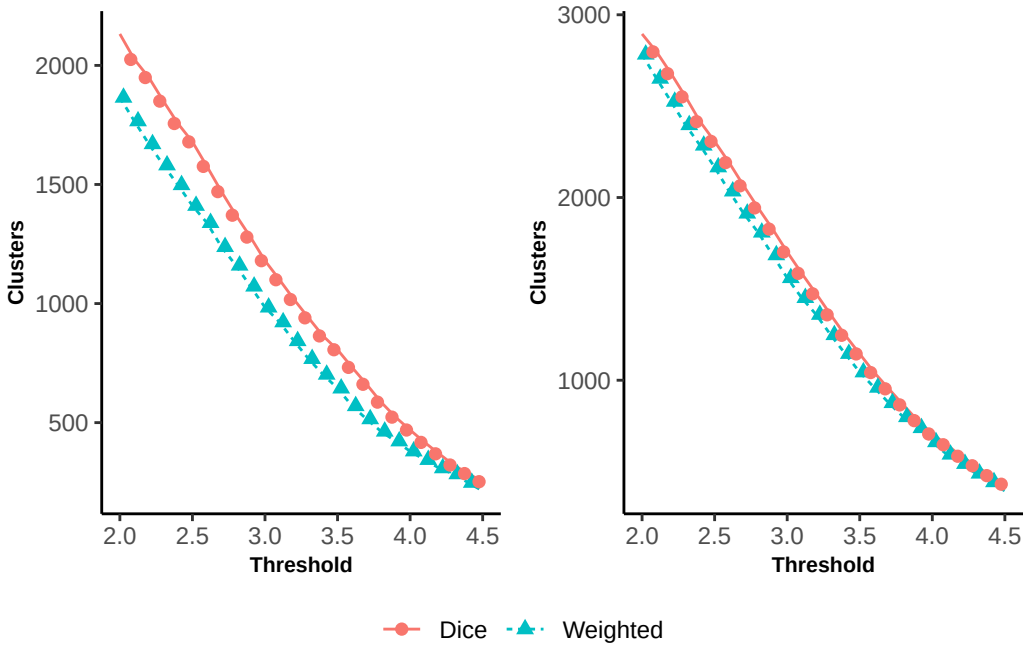


FIGURE 7.3: Plots of number of clusters against threshold for using the weighted approach and Dice coefficient for Citumbuka and Chichewa respectively

where N = number of items to be classified, k = number of clusters, c_i is a cluster, and t_j is the ground truth classification with the highest number of members in c_i.

We first created a dataset that was used to perform the evaluation. A multilingual corpus hosted by the Centre for Language Studies (CLS), Chancellor College Online Corpus was used to prepare a development data set. A seed list of 100 roots (40 nouns, 20 verbs, 15 adjectives and 15 adverbs) for Citumbuka and Chichewa was used to obtain morphological variants from the corpus. A total of 2104 terms were collected for Citumbuka and 2103 terms for Chichewa. The test dataset was created from a raw corpus from Fuko newspaper published by Nations Publications Malawi. Ten volumes of both languages were used to prepare a dataset for the experiments. The text was extracted from PDF documents and tokenised. 5, 210 tokens for Chichewa and 6,101 tokens in Citumbuka were realised. The data was analysed to remove any illegal words such as English words, named entities, and repeated words. In total 4,813 Chichewa tokens and 4,244 Citumbuka tokens were obtained. The tokens in the dataset were then clustered using Algorithm 1 and these clusters were given to two linguistics students who are also native speakers of the languages. The human assessors were asked to check the clusters and correct any mistakes done by the algorithm.

We calculate purity values for Chichewa and Citumbuka clusters based on OWA and Dice coefficient scores between words.Figure 7.4 shows the performance of the two clustering measures on our data. The data shows that the clusters created using OWA metric had more correctly classified terms than using Dice coefficient.

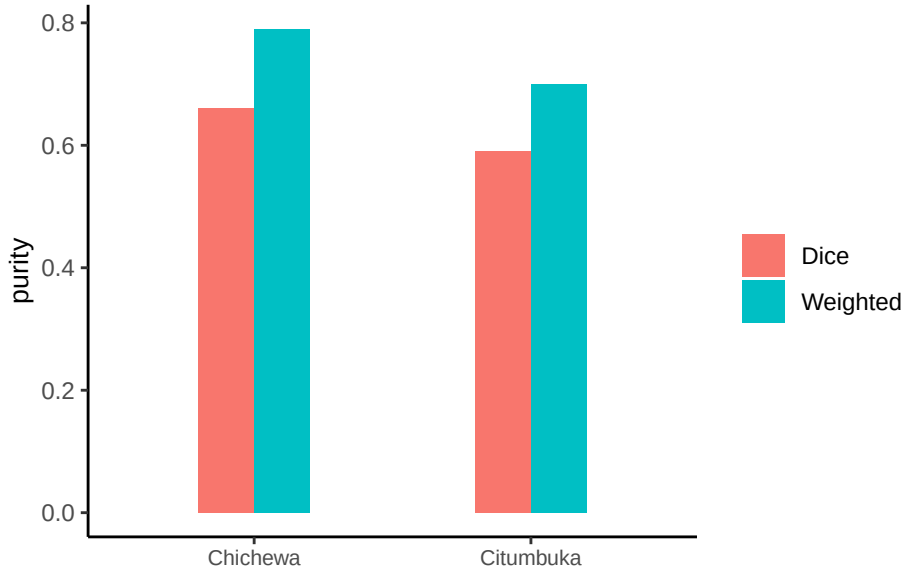


FIGURE 7.4: Cluster evaluation based on $Purity$ using the Weighted OWA and Dice on Citumbuka and Chichewa data respectively.

Using the formed OWA clusters, we performed an unsupervised segmentation – estimating the boundary between root and affixal material. Several affixes were obtained from the analysis of the prefixal and suffixal segments using the proposed algorithm. Each affix was generated based on the slot it can occupy in a word. Table 7.2 shows some of the generated correct affixes and their proposed slots.

TABLE 7.2: Example of affixes generated from the clusters.

Slot	Chichewa	Citumbuka
1	”zi’, ’ndi’, ’o’, ’li’, ’chi’	’mu’, ’vi’, ’vya’, ’gha’, ’ku’
2	’ngo’, ’chi’, ’dza’, ’sa’, ’ka’	’nga’, ’chi’, ’pa’, ’ma’, ’ku’
3	’yi’, ’zi’, ’wa’, ’ndi’, ’dzi’,	’ji’, ’ka’, ’ku’, ’chi’, ’ti’
Final vowel	’a’, ’e’, ’i’, ’o’ , ’u’	’a’, ’e’, ’i’, ’o’, ’u’
Extension	’ir’, ’ik’, ’its’, ’er’, ’ets’	’isk’, ’ir’, ’ik’, ’il’, ’esk’, ’er’
enclitic	’nso’, ’wo’, ’chi’, ’mo’, ’po’	’ko’, ’so’, ’po’

These affixes were extracted from a small dataset that was used for evaluating clustering using the Dice Coefficient and our proposed method. Affixes used in the stemming evaluation were extracted from the corpus used in the retrieval experimentation.

7.2.2 Stemming Evaluation Results and Analysis

Bantu languages have a rich morphology, and therefore experimenting with different stemming approaches is important to know how each approach performs. This work investigates

the problem of limited resources and tools, and also explores how a generic stemmer for related languages may perform. Our study aims to answer the following question:

RQ6 How do multilingual stemmers that use structural similarities of Chichewa and Citumbuka compare in terms of retrieval effectiveness with no stemming, language independent approach using character tri-grams and language specific stemmers?

We approached this question by using different language settings in the experiments including monolingual, cross language and multilingual retrieval on Citumbuka and Chichewa text. Queries were not translated for cross language and multilingual retrieval experiments. The following stemming or indexing strategies were used:

- Using space delimited tokens, i.e., words as they appeared in the corpus.
- Using a multilingual stemmer that removes suffixes and prefixes observed in the corpus.
- Using language specific rules for removing suffixes and prefixes.
- Using character tri-grams bounded by the tokens (no word-spanning trigrams).

We present MAP, P@K and nDCG@K results based on averages obtained from all topic runs. The total number of topics is 129 but the number of the topics used in the different evaluations varies due to some settings having no relevant documents. We also provide topic level analysis by providing visualisation of individual run performance for nDCG scores and statistical significance test results using Analysis of Variance (ANOVA) and Tukey's HSD (Honestly Significant Difference) test as a post hoc test (Sakai, 2014). All experiments were run in Apache Solr (using BM25 similarity measure and centralised indexing for MLIR settings) in batch mode using Python scripts.

Stemming in Within-language Retrieval

The four stemming approaches were used in monolingual retrieval runs involving Citumbuka and Chichewa corpora and queries. This was necessary to investigate how each approach will perform on documents and queries in the same language. We report results of each language separately and start with results for Chichewa and followed by Citumbuka results.

Monolingual Runs for Chichewa: In monolingual retrieval for Chichewa text, queries and document text written in Chichewa were used in the retrieval process. In total, one hundred and twenty-nine (129) queries were used in four runs based on the four stemming approaches being investigated, including the baseline – no stemming, using n-grams, using

TABLE 7.3: Evaluation scores for Chichewa monolingual runs using the baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	**0.4755**	0.4108	0.4077	0.3998
P@5	**0.4891**	0.4186	0.4186	0.4124
P@10	**0.3938**	0.3341	0.3403	0.3287
P@20	**0.2977**	0.2609	0.2593	0.2682
P@100	**0.1151**	0.1061	0.1047	0.105
P@500	**0.0284**	0.028	0.0269	0.0275
P@1000	0.0144	**0.0146**	0.0144	0.0143
ndcg@5	**0.5199**	0.4622	0.4501	0.4483
ndcg@10	**0.517**	0.4561	0.4577	0.4395
ndcg@20	**0.5307**	0.4775	0.4728	0.4654
ndcg@100	**0.5963**	0.5403	0.5394	0.5296
ndcg@500	**0.6358**	0.5898	0.5861	0.5764
ndcg@1000	**0.6389**	0.5971	0.5974	0.584
ndcg	**0.6389**	0.5971	0.5974	0.5847

a generic stemmer and a domain specific stemmer. The evaluation first investigates how each approach performed using several evaluation metrics such as MAP, nDCG and Precision. This is followed by topic specific analysis of the results. The results are presented in both tables and graphs.

Table 7.3 shows a summary of evaluation scores for MAP, Precision at a cut-off and nDCG at a cut-off, and overall average nDCG for the within-language experiments. The results indicate that using words as they appear in the documents for indexing and querying provides better results that the other three methods. The performance of the generic stemmer was also better than n-grams and a language specific stemmer.

Table 7.3 provides nDCG scores at different cut-off points. This shows the change in performance at different ranks, which is important for user oriented evaluation. The results indicate that using words without stemming for indexing and querying produced better results than the other three methods, which had small variation in performance.

The precision-recall curve shows the tradeoff between precision and recall for different thresholds. A high area under the curve represents both high recall and high precision. Figure 7.5 shows the precision-recall curve for the monolingual runs of Chichewa. The curve shows that using words as they appeared in the documents and queries had better performance. The remaining three methods had almost similar performance.

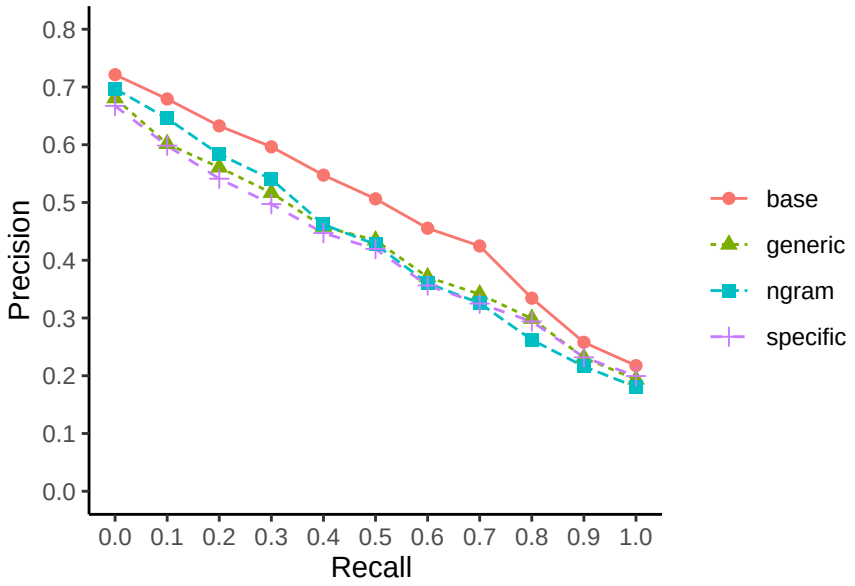


FIGURE 7.5: Recall-precision curves for Chichewa monolingual runs

Performance of a retrieval system is reported as averages over a set of topics in a test collection due to variability in topic performance. However, the variability in topic performance contributes to the overall retrieval system evaluation. We further analysed topic level performance using overall nDCG scores for each topic.

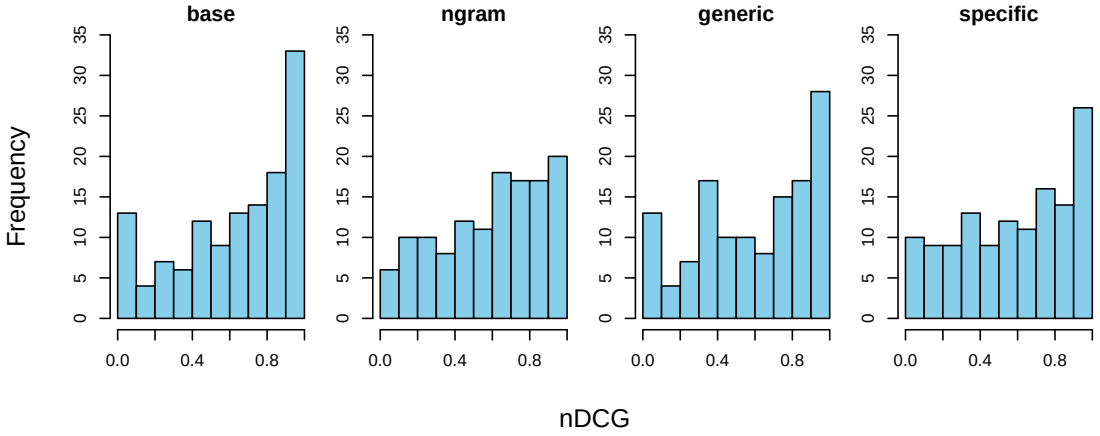


FIGURE 7.6: The distribution of nDCG scores for Chichewa monolingual runs.

Figure 7.6 shows the nDCG score distribution in the four runs for the stemming methods for all the queries. The results show that the baseline had the highest number of queries for the highest bin (0.9 to 1) (specific = 26 , generic = 28 , n-gram = 20 , baseline = 33).

The baseline run uses documents and queries that have not been processed in the context of morphology. The other three runs used different approaches namely, using statistical processing where words are divided into three character sequences, using a stemmer with general rules that cover the morphology of several related languages and using a Chichewa stemmer. It would also be important to know how each method performed in relation to

the baseline. Figure 7.7 shows the per topic differences in overall nDCG scores between the baseline and the other three stemming methods being explored.

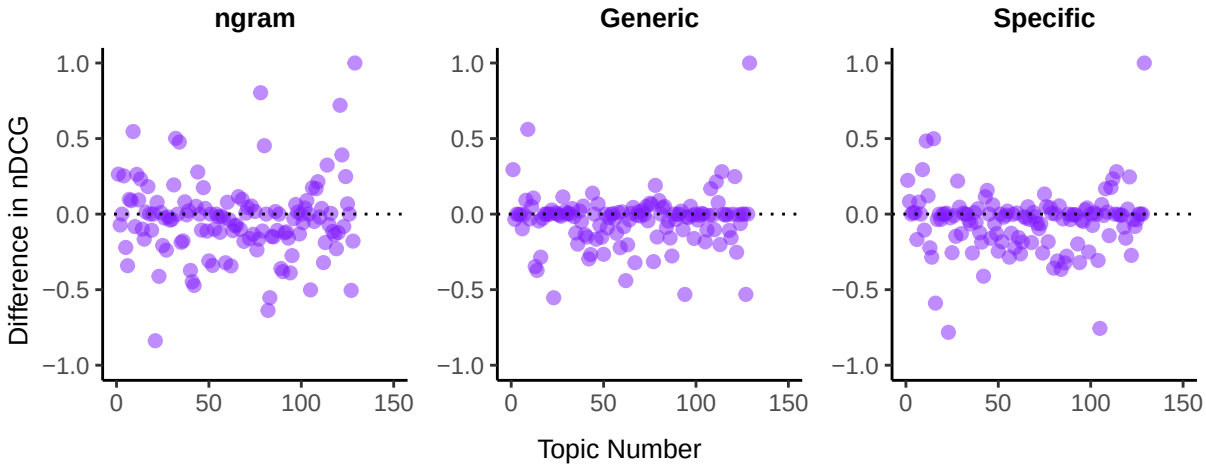


FIGURE 7.7: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches for Chichewa monolingual runs

We further used nDCG as a measure to investigate the differences in performance of the four stemming methods, i.e., whether the differences in performance observed were not due to chance. We conducted a two-way ANOVA without replication to test our null hypothesis H_0 = all the four stemming methods are the same or equivalent ($p = 0.05$).

TABLE 7.4: Two-way ANOVA (without replication) for within language retrieval for Chichewa.

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.175	0.058412	2.6268	0.05009
Between-topics	128	38.24	0.29875	13.4347	< 2e-16
Within	384	8.539	0.022237	–	–

Table 7.4 shows the results of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 129 topics. The stemming effect is not statistically significant ($F(128, 384) = 0.05009, p > 0.050$).

Monolingual Runs for Citumbuka: In monolingual retrieval for Citumbuka text, queries and text of Citumbuka were used for retrieval. In total, one hundred and twenty-nine (129) queries were used in four runs based on the four stemming approaches being investigated. Table 7.5 shows a summary of scores for the within-language experiments for Citumbuka. The results show that n-gram had the best performance followed by the baseline. The generic method and Citumbuka stemmer had scores within the same range.

TABLE 7.5: Evaluation scores for Citumbuka monolingual runs using the baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	0.2524	0.2259	**0.2863**	0.222
P@5	0.1897	0.1621	**0.2**	0.1586
P@10	0.1405	0.1276	**0.1431**	0.1198
P@20	0.0901	0.0841	**0.0922**	0.0789
P@100	0.0238	0.0246	**0.0262**	0.0247
P@500	0.0051	0.0057	**0.0061**	0.0056
P@1000	0.0027	0.0029	**0.0031**	0.0029
ndcg@5	0.2678	0.2312	**0.3084**	0.2329
ndcg@10	0.2936	0.2586	**0.3216**	0.2512
ndcg@20	0.311	0.279	**0.3382**	0.2723
ndcg@100	0.3278	0.3096	**0.3664**	0.3081
ndcg@500	0.3338	0.3201	**0.3792**	0.3183
ndcg@1000	0.3383	0.3228	**0.3805**	0.3211
ndcg	0.3383	0.3228	**0.3805**	0.3211

Figure 7.8 shows the precision-recall curve for the Citumbuka monolingual runs. n-gram performance tend to be the best followed by the baseline while the remaining two stemming methods have very similar performance.

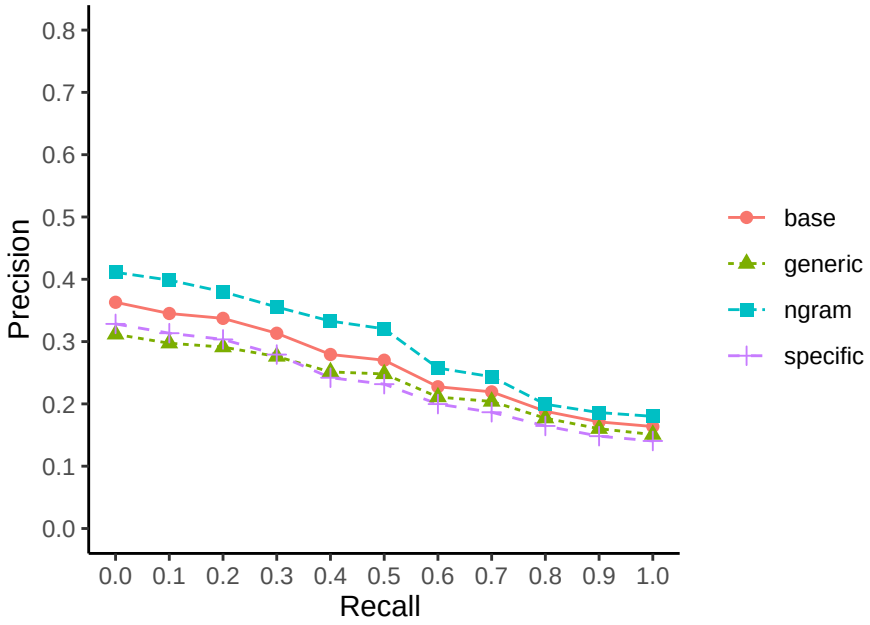


FIGURE 7.8: Recall-precision curves for Citumbuka monolingual runs

Figure 7.9 shows nDCG score distribution in the runs of the four stemming methods. The distribution for ngram shows relatively lower counts for the lower score bins and higher counts for higher score bins.

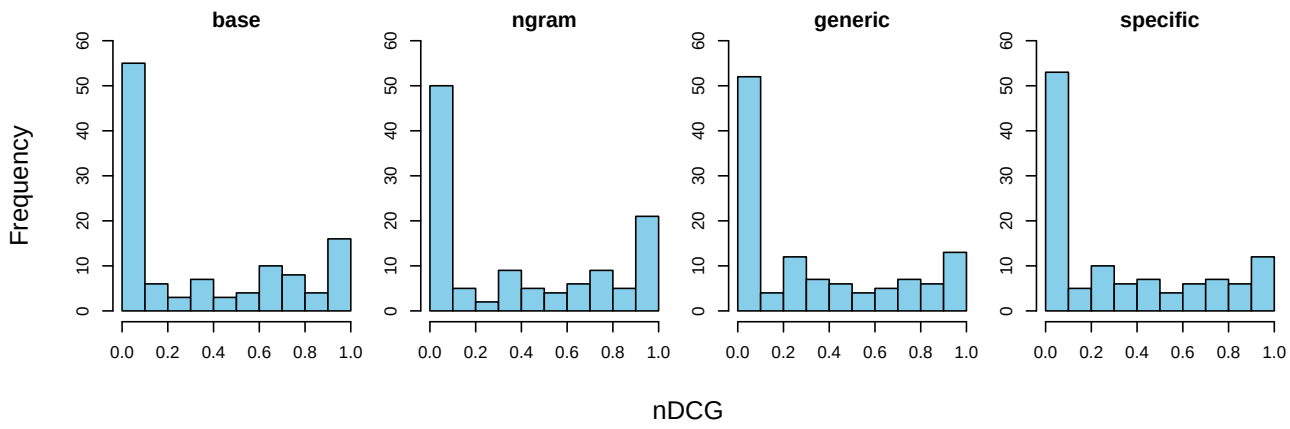


FIGURE 7.9: The distribution of nDCG scores for Citumbuka monolingual runs

Figure 7.10 shows the per topic differences between the baseline and the other stemming methods being investigated. The graph shows that many queries in the three methods had similar performance with that of the baseline. However, n-gram had more queries that had better performance than the the two other methods.

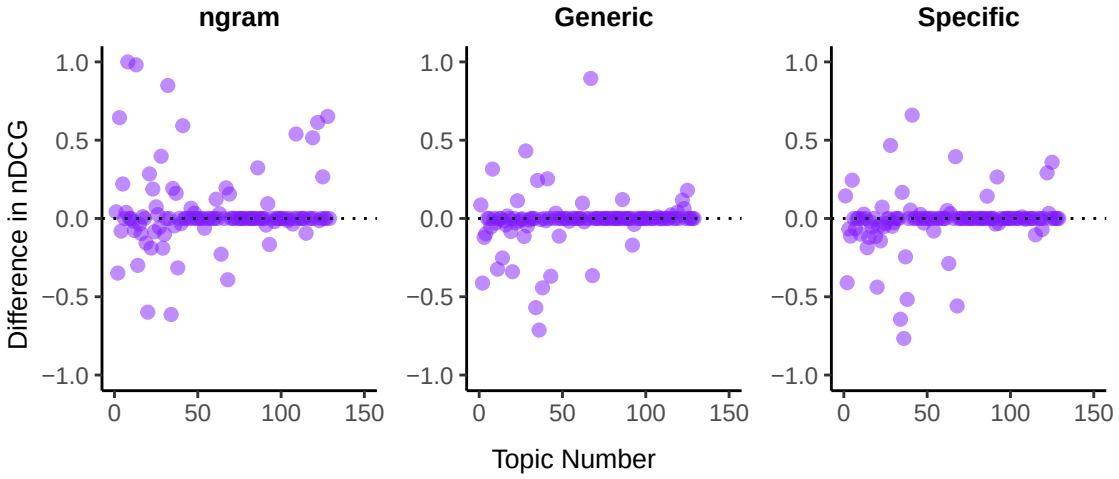


FIGURE 7.10: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches for Citumbuka monolingual runs.

We also investigated whether the differences in performance of the four stemming methods were due to chance. We conducted a two-way ANOVA without replication using nDCG scores to test our null hypothesis H_0 = all the four stemming methods are the same or equivalent ($p = 0.05$). The ANOVA test was significant and we conducted a post-hoc test using Tukey HSD to find the pairs of stemming methods that were significantly different.

TABLE 7.6: Two-way ANOVA (without replication) for within language retrieval for Citumbuka.

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.266	0.08883	4.817	0.002674
Between-topics	115	57.773	0.50237	27.242	< 2.2e-16
Within	345	6.362	0.01844	–	–

Table 7.6 shows the results of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 116 topics. The stemming effect is statistically significant $(F(115, 345) = 0.002674, p < 0.050)$. Tukey HSD test shows that the difference between stemming methods n-gram and generic is statistically significant (p = 0.0071893). Further, the difference in performance of stemming methods n-gram and Citumbuka stemmer (specific) is statistically significant (p = 0.0052401). However, the difference in performance of stemming methods of n-gram and the baseline is not statistically different.

Stemming in Cross-language Retrieval

Cross-language retrieval involves using queries in one language to retrieve documents written in another language. The standard approach is to translate queries to the language of the documents. In our experimentation, translation was not used, but we assumed that matching would be possible due to the similarity of vocabulary between the two investigated methods. We investigated the four stemming methods in two cross-lingual retrieval environments.

Cross-language Retrieval using Chichewa Corpus and Citumbuka Queries: Chichewa corpus was indexed using the four stemming methods, and Citumbuka queries were used to match and retrieval the Chichewa documents. Citumbuka queries were also given a similar treatment as the Citumbuka documents. For the language specific stemming, the Citumbuka queries used Citumbuka stemmer while the Chichewa stemmer was used to process the Chichewa queries.

Table 7.7 shows the results of cross language runs for Citumbuka queries on Chichewa corpus. The results show that the baseline had better performance than the other three stemming methods, which had similar performance.

The precision-recall plot in Figure 7.11 for cross lingual retrieval for Chichewa corpus using Citumbuka queries shows that the four stemming methods had similar performance, with the baseline having a slightly better performance and the language specific stemming performing worst.

TABLE 7.7: Evaluation scores for cross lingual retrieval for Chichewa corpus using Citumbuka queries runs for the baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	**0.2781**	0.2476	0.2283	0.1901
P@5	**0.2698**	0.2388	0.2264	0.1969
P@10	**0.2233**	0.2008	0.1876	0.1597
P@20	**0.1783**	0.1663	0.1539	0.1318
P@100	0.0743	**0.075**	0.0703	0.0587
P@500	0.0185	**0.0207**	0.02	0.0176
P@1000	0.0098	0.0113	**0.0115**	0.0098
ndcg@5	**0.2788**	0.2527	0.2414	0.1917
ndcg@10	**0.2861**	0.2606	0.2497	0.1973
ndcg@20	**0.3017**	0.2759	0.2633	0.2125
ndcg@100	**0.3511**	0.3338	0.3195	0.2679
ndcg@500	0.3776	**0.3795**	0.3674	0.3181
ndcg@1000	0.3835	**0.3932**	0.3862	0.3315
ndcg	0.3835	**0.3932**	0.3862	0.3315

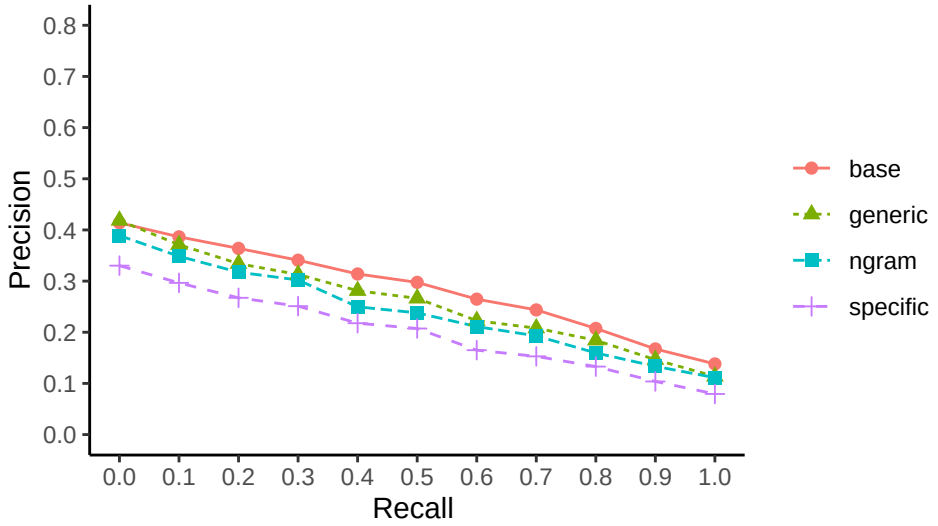


FIGURE 7.11: Recall-precision curves for cross language runs for Citumbuka queries on Chichewa corpus using the four investigated stemming methods

nDCG score distribution in the four runs for all the queries is given in Figure 7.12. The graph shows the baseline to have the highest number of counts in the lowest score bin (0 to 0.1). N-gram run scores shows a relatively balanced distribution with a relatively lower count in the bin for the highest score (0.9 to 1).

Figure 7.13 shows the differences between individual topic nDCG scores of three methods being explored, and the baseline. Generic stemming results show small differences with the baseline. The use of n-grams shows a bigger variation in differences both for positive and negative differences. The use of language specific stemming shows bigger differences

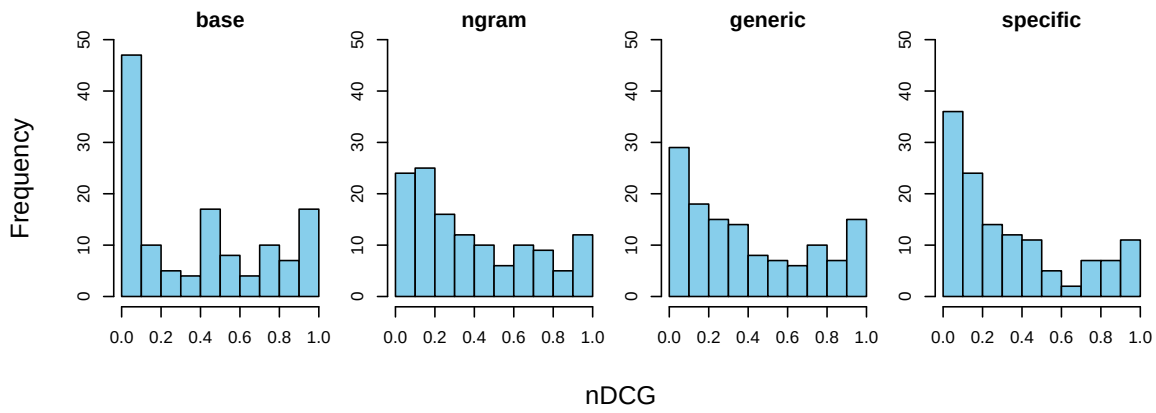


FIGURE 7.12: The distribution of nDCG scores for the baseline, ngram, generic and language specific stemming for Citumbuka queries on Chichewa corpus.

only in the negatives, indicating worse performance than the baseline.

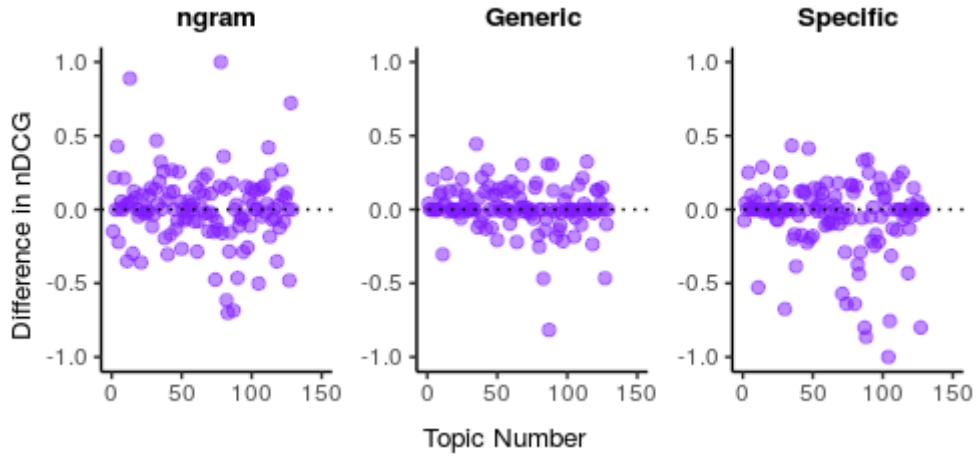


FIGURE 7.13: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches for cross lingual retrieval for Chichewa corpus using Citumbuka queries

We used nDCG to investigate the differences in performance of the four systems. We conducted a two-way ANOVA without replication to test our null hypothesis H_0 = all the four methods are the same or equivalent.

Table 7.8 shows the result of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 129 topics. The stemming effect is statistically significant ($F(115,345) = 0.005824$, $p < 0.050$). Tukey HSD test shows that the differences between specific and the other three are statistically significant ($p = 0.03964$ for the baseline and specific, $p = 0.00933$ for generic and specific and $p = 0.02668$ for specific and n-gram). The results show that the language specific stemmer has significantly different performance with the other methods.

TABLE 7.8: Two-way ANOVA (without replication) for cross language retrieval for Citumbuka queries on Chichewa corpus.

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.312	0.10389	4.2332	0.005824
Between-topics	128	45.137	0.35263	14.3681	< 2.2e-16
Within	384	9.424	0.02454	–	–

TABLE 7.9: Evaluation scores for cross language retrieval for Chichewa queries on Citumbuka corpus for baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	**0.1898**	0.1713	0.1879	0.1778
P@5	**0.1414**	0.1345	**0.1414**	0.1207
P@10	**0.1009**	**0.1009**	**0.1009**	0.0957
P@20	0.0647	0.0629	0.0616	**0.0655**
P@100	0.0178	0.0185	0.0192	0.0187
P@500	0.004	0.0046	**0.0051**	0.0045
P@1000	0.0021	0.0025	**0.0029**	0.0024
ndcg@5	**0.2053**	0.1772	0.1951	0.182
ndcg@10	**0.2122**	0.1883	0.2054	0.197
ndcg@20	**0.2244**	0.2026	0.2118	0.2163
ndcg@100	**0.2415**	0.2247	0.238	0.2369
ndcg@500	0.2507	0.2383	**0.2585**	0.249
ndcg@1000	0.255	0.2455	**0.2715**	0.2558
ndcg	0.255	0.2455	**0.2715**	0.2558

Cross Language Retrieval for Chichewa Queries on Citumbuka Corpus: Cross language retrieval for Citumbuka queries on Chichewa corpus were run based on the four stemming methods. Words in the Citumbuka corpus were stemmed using the Citumbuka stemmer and words in the Chichewa queries were stemmed using the Chichewa stemmer.

Table 7.9 shows the results for cross language retrieval for Chichewa queries on Citumbuka corpus for the four stemming methods. The results show that n-gram and the baseline runs had similar results while the remaining two had similar results. N-gram run performance started at lower scores but the scores increased surpassing the baseline scores in the lower ranks.

The precision-recall curve in Figure 7.14 shows that the four methods had similar performance. The n-gram run performed slightly better than the other stemming methods, including the baseline.

Figure 7.15 shows nDCG score distribution for each of the runs, i.e, the four stemming methods. N-gram had lower number of counts in the lowest score bin than the rest of the

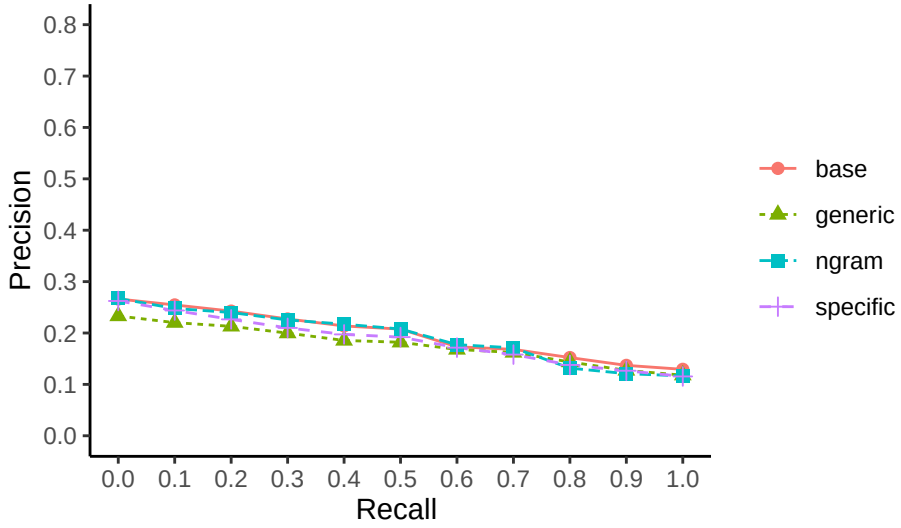


FIGURE 7.14: Recall-precision curves for cross language runs for Chichewa queries on Citumbuka corpus on the four investigated methods

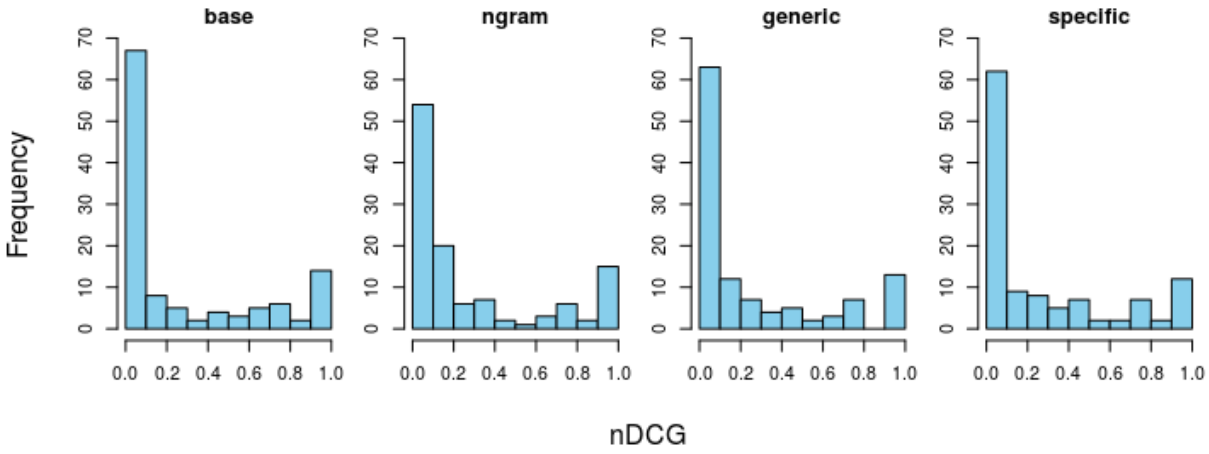


FIGURE 7.15: The distribution of nDCG scores for the baseline, ngram, generic and language specific stemming.

methods and relatively better performance than the other methods.

Figure 7.16 shows the per topic differences between the four methods being explored. The plot shows that the three stemming methods had similar performance with the baseline.

We wanted to explore whether the differences seen in the performance of the four methods were random and we used nDCG as a criterion for our investigation. We conducted two-way ANOVA without replication to test our null hypothesis H_0 = all the four methods are the same or equivalent ($p = 0.05$).

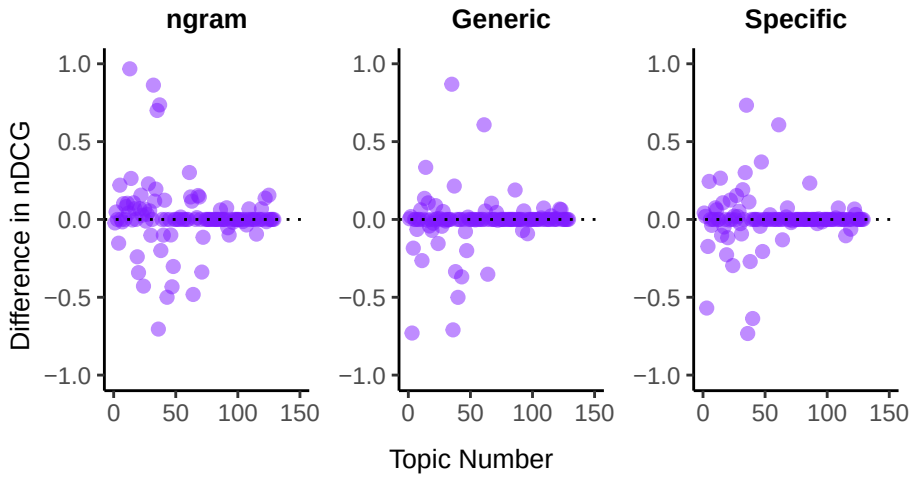


FIGURE 7.16: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches for the cross language retrieval of Chichewa queries on Citumbuka corpus.

TABLE 7.10: Two-way ANOVA (without replication) for cross language retrieval of Chichewa queries on Citumbuka corpus.

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.040	0.01347	0.9259	0.4284
Between-topics	115	50.919	0.44277	30.4297	<2e-16
Within	345	5.020	0.01455	–	–

Table 7.10 shows the result of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 116 topics. The stemming effect is not statistically significant $(F(115, 345) = 0.9259, p = 0.4284, p < 0.050)$.

Stemming in Multilingual Retrieval

In multilingual retrieval, the Chichewa and Citumbuka corpora were combined and indexed together using a centralised architecture approach. Each of the runs for our experiments used a single approach for stemming the index terms and query terms. Query terms were not translated and were used to match and retrieve documents written in two languages.

Multilingual Retrieval using Chichewa Queries: Monolingual queries written in Chichewa were run on the multilingual corpus consisting of Citumbuka and Chichewa text. Each of the run used a single method for pre-processing index terms as well as query terms.

Table 7.11 shows the performance of the runs for multilingual retrieval using Chichewa corpus. The results indicate better performance for the baseline. However, the remaining methods show very similar performance.

TABLE 7.11: Evaluation scores for multilingual retrieval for Chichewa queries runs for the baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	**0.4748**	0.4067	0.4119	0.3859
P@5	**0.493**	0.4155	0.4341	0.414
P@10	**0.3946**	0.3419	0.3535	0.3326
P@20	**0.3023**	0.2671	0.2671	0.2682
P@100	**0.1178**	0.1086	0.1071	0.1067
P@500	**0.0292**	0.0285	0.0279	0.0283
P@1000	**0.0149**	**0.0149**	**0.0149**	0.0148
ndcg@5	**0.5229**	0.4555	0.4596	0.4362
ndcg@10	**0.518**	0.4551	0.464	0.4288
ndcg@20	**0.5334**	0.4739	0.4746	0.4489
ndcg@100	**0.5992**	0.5369	0.5417	0.5182
ndcg@500	**0.6397**	0.5867	0.5899	0.5673
ndcg@1000	**0.6434**	0.5953	0.602	0.5759
ndcg	**0.6434**	0.5953	0.602	0.5759

Figure 7.17 shows a precision-recall curve for the runs using multilingual corpus and Chichewa queries. The results indicate that the baseline had better performance while the remaining stemming methods had similar performance.

Figure 7.18 shows the distribution of scores for individual queries for each of the four runs. The baseline shows better performance than the other three methods with most scores in the bins for higher nDCG scores. The n-gram run had almost uniform distribution of scores.

Figure 7.19 shows the per topic differences between the baseline and the three stemming methods being explored. The plot shows that the baseline had more similar performance

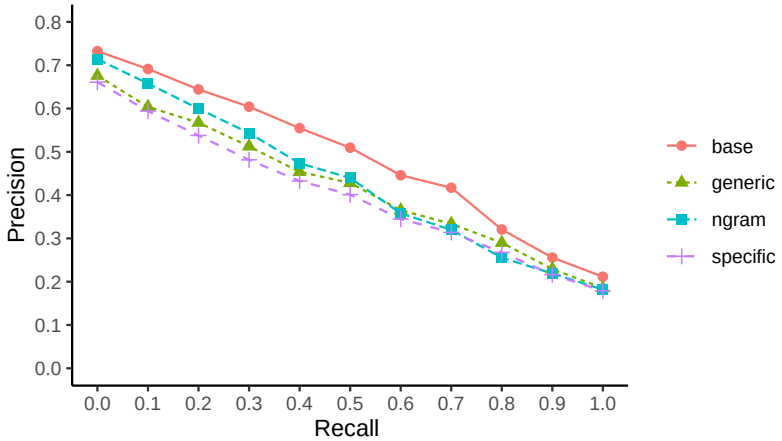


FIGURE 7.17: Recall-precision curves for multilingual retrieval for Chichewa queries runs

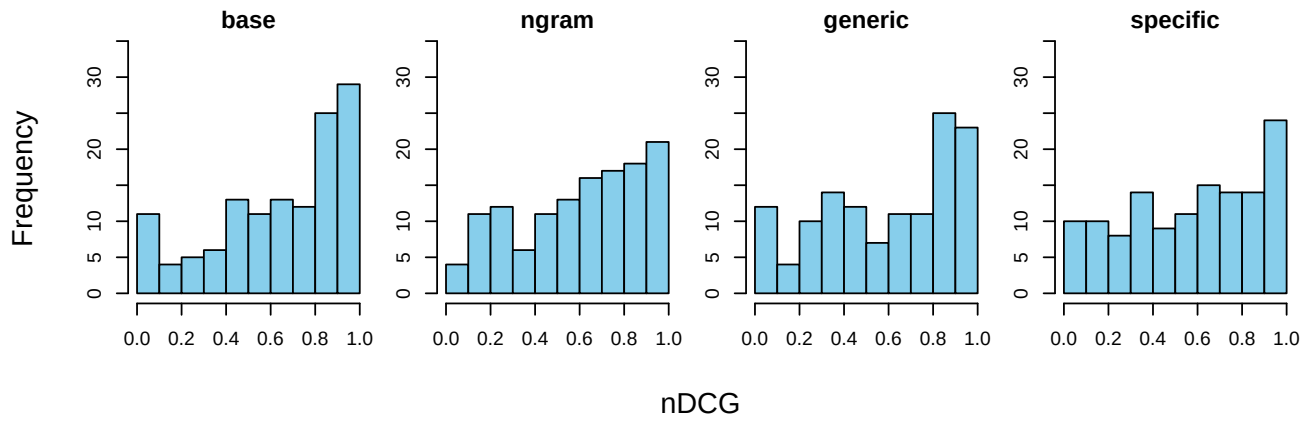


FIGURE 7.18: The distribution of nDCG scores for multilingual retrieval for Chichewa queries.

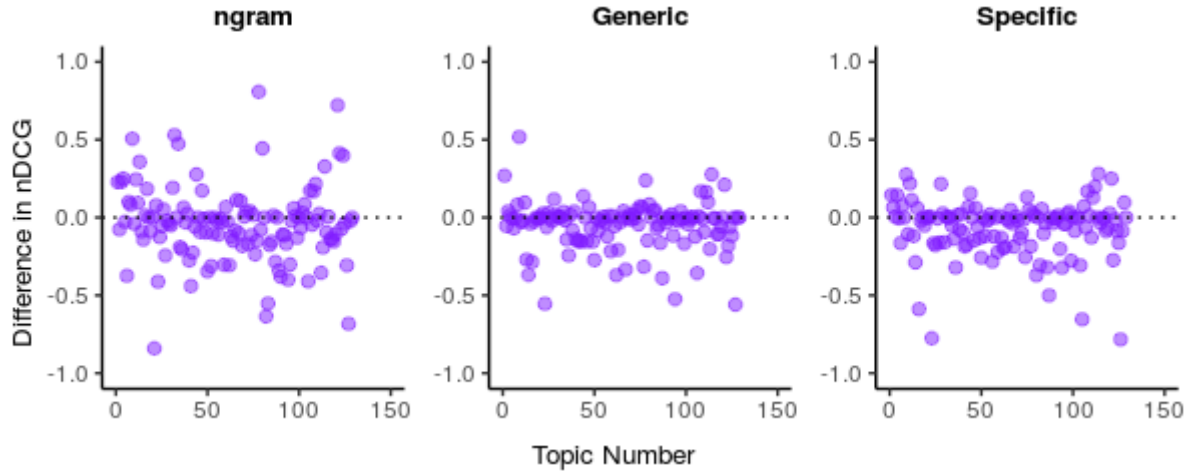


FIGURE 7.19: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches multilingual retrieval for Chichewa queries

to the multilingual stemmer and language specific stemming. N-gram stemming had more positive differences.

We also investigated the differences in performance of the four systems using overall nDCG scores. We conducted a two-way ANOVA without replication to test our null hypothesis H_0 = all the four methods are the same or equivalent $(p = 0.05)$.

Table 7.12 shows the results of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 129 topics. The stemming effect is statistically significant (F(128,384) = 5.1358 , p = 0.001712, p < 0.050). Tukey HSD test shows that the differences between baseline and generic and specific are statistically significant (p = <0.001 for the baseline and specific, p = 0.0353 for generic and the baseline).

TABLE 7.12: Two-way ANOVA (without replication) for multilingual retrieval for Chichewa queries

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.312	0.104081	5.1358	0.001712
Between-topics	128	36.950	0.288671	14.2442	<2.2e-16
Within	384	7.782	0.020266	–	–

Multilingual Retrieval using Citumbuka Queries: Citumbuka and Chichewa documents were used to create a multilingual corpus. The corpus was indexed using the four stemming methods, a separate index for each method. Each index was queried using Citumbuka queries processed as the index terms. The section reports on the results on multilingual retrieval using Citumbuka queries.

TABLE 7.13: Evaluation scores for multilingual retrieval for Citumbuka queries run for the baseline – words with no processing, generic stemming, n-grams and using language specific rules.

Metric	Baseline	Generic	n-gram	Specific
MAP	0.2576	0.2565	**0.2791**	0.2289
P@5	0.3238	**0.3286**	0.3095	0.2873
P@10	**0.2841**	0.2778	0.2746	0.2421
P@20	**0.2183**	**0.2183**	0.2067	0.1873
P@100	0.0888	**0.0889**	0.0865	0.0758
P@500	0.0231	**0.0244**	0.0232	0.0216
P@1000	0.0121	**0.0138**	0.0134	0.0123
ndcg@5	0.3222	0.3053	**0.3262**	0.2797
ndcg@10	0.3206	0.3036	**0.333**	0.278
ndcg@20	0.3208	0.311	**0.3357**	0.2821
ndcg@100	0.3603	0.3546	**0.3783**	0.3197
ndcg@500	0.3937	0.4007	**0.4193**	0.3706
ndcg@1000	0.4001	0.4198	**0.439**	0.3874
ndcg	0.4001	0.4198	**0.439**	0.3874

Table 7.13 shows results for multilingual retrieval for Citumbuka queries on Citumbuka and Chichewa corpora. The results show slight differences between the baseline and the other stemming methods. The n-gram based run has slightly higher nDCG scores than the other methods while the baseline has slightly higher scores for precision. Figure 7.20 shows the precision-recall curves for the four stemming methods. The curve for n-gram run indicates better performance than the baseline and the multilingual stemmer – these two seems to have similar curves.

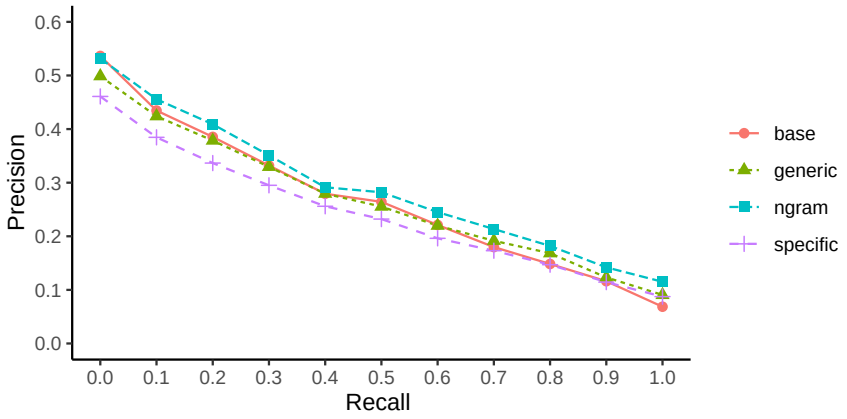


FIGURE 7.20: Recall-precision curves for for Citumbuka queries runs

Figure 7.21 shows the nDCG scores distribution for the runs of the four stemming methods using Chichewa queries on a corpus of Chichewa and Citumbuka documents. Figure

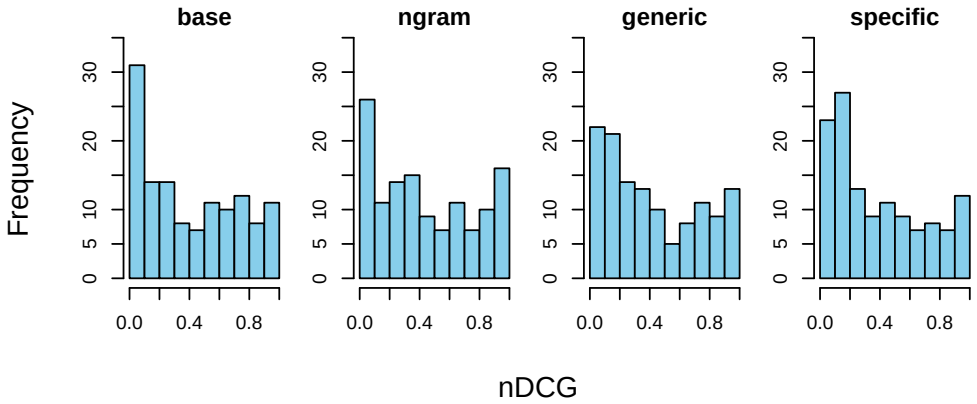


FIGURE 7.21: The distribution of nDCG scores for the baseline, ngram, generic and language specific stemming.

7.22 shows the differences in nDCG of individual queries between the baseline and the other three stemming methods. The n-gram run shows more differences especially positive differences indicating better performance than the baseline.

We also investigated the differences in performance of the four systems. We conducted a two-way ANOVA without replication to test the following null hypothesis: H_0 = all the four methods are the same or equivalent $p = 0.05$.

TABLE 7.14: Two-way ANOVA (without replication) for for Citumbuka queries runs.

	Degrees of freedom	Sum of squares	Mean squares	F_0	Pr(>F_0)
Between-stemming	3	0.194	0.06461	2.3199	0.07499
Between-topics	125	40.041	0.32032	11.5023	< 2e-16
Within	375	10.443	0.02785	–	–

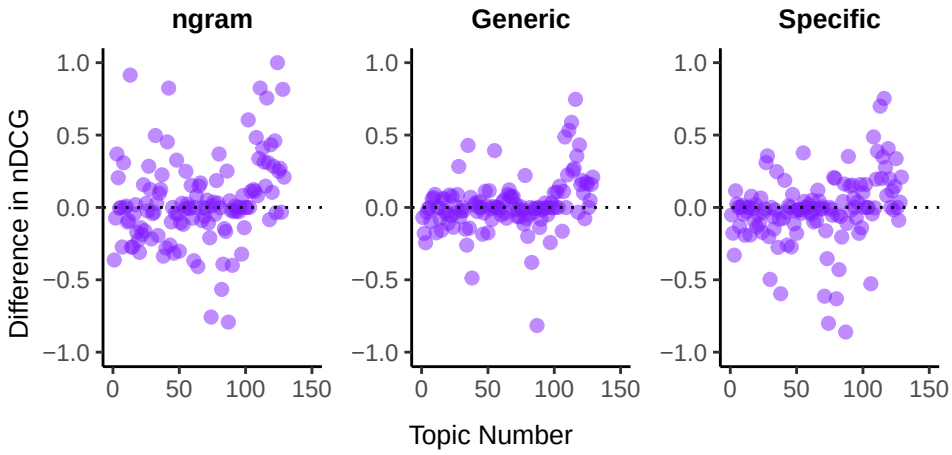


FIGURE 7.22: The per topic differences of nDCG between the Baseline and each of the three other stemming approaches for the multilingual runs for Citumbuka queries

Table 7.14 shows the result of a two-way ANOVA (without replication) applied to the comparison of m = 4 stemming methods with n = 126 topics. The stemming effect is not statistically significant ($F(125,375) = 2.3199$, $p = 0.07499$, $p > 0.050$).

In the remainder of the section (7.2.2), we provide an analysis of the results at topic level. In particular, we investigate queries with the best and worst performance for the baseline and how stemming impacted them. We then examine the results of queries using named entities and general terms, and how these queries were affected by stemming. Finally, we provide a summary of all results in terms of nDCG@10.

Topic Level Analysis

Our evaluation uses a set of 129 topics – the topics were translated to Chichewa and Citumbuka by native speakers of the languages. Most of the corpora is made up of Chichewa documents, and therefore the relevant set of documents is mostly comprised of Chichewa documents. Using words as they appear in the documents for indexing and queries for querying, i.e., the baseline, performed relatively better on Chichewa corpus and queries than the other methods. This is not surprising as the number of relevant documents were mostly those written in Chichewa. However, the baseline retrieved a lower number of documents than the other methods. Table 7.15 shows the number of documents retrieved by each method. N-grams performed well with Citumbuka queries – this is expected as most of the matching of query terms and document terms is approximate matching; no translation was used and matching was based on similar vocabulary. Possible matching using vocabulary similarity depends on whether the terms are equivalent across languages and proper names are a big candidate for these queries. In addition, using n-grams would match more terms that are cognates especially words with a small variation. The generic method performed

better than the language specific stemming and retrieved the most relevant documents for Citumbuka multilingual retrieval. The approach also performed relatively similar to n-gram and the baseline on Citumbuka queries on Chichewa corpus (similar to baseline). Language specific stemming performed the worst than any of the methods; this may have been due to over stripping, which made matching of terms impossible.

TABLE 7.15: Number of retrieved relevant documents by each stemming method

Query	Type	Baseline	Generic	n-gram	Specific
Chichewa	Within	1848	1878	1856	1846
Citumbuka	Within	307	336	358	334
Chichewa	Cross	245	288	339	277
Citumbuka	Cross	1265	1456	1487	1258
Chichewa	Multilingual	1923	1926	1923	1909
Citumbuka	Multilingual	1523	1737	1686	1544

Our observation on the performance of the methods showed that performance was skewed towards Chichewa corpus and topics. To understand the differences in the topics between the languages, we investigated what queries were hard and easy. We first investigated what queries were easy using the multilingual corpus. We then investigated the worst queries for both language topics and a single language.

Best Performing Queries: Our first topic level investigation looked at the best performing topics. We restricted our attention to the best ten performing topics across languages in multilingual retrieval (we selected topics with the highest scores but with the least difference in nDCG scores). Our analysis focused on the impact of stemming in these high performing topics (topics were selected using the baseline scores). The selected queries show that stemming harms performance. We wanted to investigate the type of queries involved and found that every query had at least one proper name. This is not surprising as proper names would not change significantly across languages. Our best performing query (Topic #45) across language on 'diseases treated by Aloe vera' had the same performance across languages. The second best performing query was on *Barack Obama's visits to Africa*, Topic #53. The performance of these two queries did not improve with stemming, as stemming changed the matching terms for both named entities and common words. The queries did not have the same advantage of exact matching that the baseline had, and which had allowed for matching with relevant documents. Similar to studies on non-English corpus (Hollink et al., 2004), these results provide us with insight that proper names have high discriminating power in estimating relevance, and their use in queries can help to retrieve relevant documents, i.e., improve retrieval performance.

Large Performance Difference Queries: We investigated queries that performed very well when using one language as the topic but their translation equivalents performed very

TABLE 7.16: Average nDCG for the best and worst performing queries

	Best	**Worst**	**Difference**
Chichewa			
Baseline	0.904	0.0941	0.859
Generic	0.884	0.189	0.788
ngram	0.87	0.345	0.723
specific	0.854	0.158	0.647
Citumbuka			
Baseline	0.921	0.0402	0.122
Generic	0.893	0.177	0.134
ngram	0.77	0.243	0.135
specific	0.795	0.118	0.126

badly. Our analysis focused on top ten queries of this type in multilingual retrieval and we analysed their nDCG values. Nine out of ten of these topics were Chichewa queries. The performance of these Chichewa queries did not improve with stemming. However, performance of the Citumbuka queries slightly improved with stemming. The topics in this set did not have proper names except for one query, but the naming did not match with that in the documents (these queries consisted of general terms). The results improved slightly with stemming for the Citumbuka topics but were largely the worst performing of this set. There were two reasons for the differences in performance. Firstly, cognates between the translations of the queries between the languages were almost non-existent – the translations looked very different. For example, Topic 95 about the benefits of breastfeeding newborn babies was translated to Citumbuka as *uweme bakuwonkhesa wana wakubabika sono* while the Chichewa translation was *ubwino woyamwitsa ana obadwa kumene*. The only matching term was *wana* and *ana* but could only be matched after stemming, hence very little improvements as these were also high frequent terms in our corpus. Secondly, the relevant documents for these topic were only available in Chichewa. Retrieving relevant documents needed matching terms across language and having little or no cognates made it almost impossible to match with relevant documents. Our further analysis showed that queries that improved performance had a single cognate. This cognate allowed very few relevant documents to match as they were general terms. The rest of the queries had no relevant documents matching because of lack of links between the languages. Therefore, performance of retrieval using related languages with no translation depends on the vocabulary similarity between the two languages. The more similar the languages, the better the retrieval performance. More importantly, stemming can improve performance for such hard queries.

Worst Performing Queries: We also investigated the worst performing queries. Our treatment only focused on the ten worst performing queries for both languages in multilingual retrieval. The topics in this set consisted of three topics about named entities and seven topics with queries containing general terms only. The three named entity queries

used a spelling different from that used in the corpus, i.e., Amerika in the query and America in the corpus. Queries with general terms in this set had either terms not used in the relevant documents or terms that required some normalisation to match the relevant documents. Therefore, stemming improved performance of these hard queries. The proposed method performed well on queries with general terms. However, using n-grams had better advantage for queries with spelling variations in their terms. Table 7.16 provides the average nDCG of the top ten topics, ten worst topics and ten topics with huge performance difference.

Query Classification: The analysis of the best performing queries showed that queries with named entities had the best performance across languages. We classified our queries into two categories, namely: queries with proper names and queries with general terms. From the 129 queries, 46 queries had at least one named entity and 83 queries consisted of general terms only. Table 7.17 provides the average nDCG queries with proper names and queries with general terms for our experiments.

TABLE 7.17: Average nDCG for queries with named entities and general terms

		baseline	**generic**	**ngram**	**specific**
Chichewa					
monolingual	common	**0.6052**	0.54	0.5747	0.5357
	named	**0.6831**	0.6741	0.6181	0.6522
multilingual	common	**0.6177**	0.5527	0.5848	0.5351
	named	**0.6724**	0.6527	0.6226	0.6300
Cross-language	named	0.4398	0.4313	**0.4612**	0.4528
	common	0.1656	0.1584	**0.1760**	0.1720
Citumbuka					
monolingual	common	0.3045	0.2834	**0.3534**	0.2817
	named	0.4599	0.4514	**0.49761**	0.4494
multilingual	named	0.5758	0.6170	**0.6186**	0.5076
	common	0.2989	0.3088	**0.3390**	0.3171
Cross-language	named	**0.6364**	0.6106	0.5919	0.4845
	common	0.2436	0.2701	**0.2772**	0.2614

Results of our query analysis shows that performance was mostly skewed towards languages. Stemming did not improve performance for Chichewa corpus. Improvements were only registered on Citumbuka corpus, with n-gram having the most increment, followed by the generic stemmer. Using n-grams improved retrieval for Citumbuka queries in multilingual and monolingual retrieval. However, when Chichewa corpus was used with Citumbuka queries, stemming did not improve performance of named entity queries, but better results were obtained with queries involving general terms only. These results indicate that stemming can improve retrieval and that our proposed method has competitive performance. Since relevant documents are skewed towards Chichewa documents, it is not possible to see a consistent pattern across languages. These results shows that stemming

is important for languages with little content especially when multilingual retrieval is involved with better resourced languages.

We wanted to understand the types of queries for which the proposed approach worked better. Topic 12 with title "*Is praying is useful?*" performed better for Chichewa queries in both multilingual and monolingual query. The query for Chichewa was "kodi Kupemphera ndikothandizadi?". Interestingly, the relevant document had phrases such as *Kodi kupemphera n'kothandizadi? ... kupemphera n'kothandiza...kungothandiza*. The query sheds more light on how the topics were formulated and other linguistic issues. Firstly, this shows that apart form using named entities, words as they appear in the corpus were used to formulate topics. This gave an advantage to the baseline. Secondly, the use of contractions in Chichewa has effect on retrieval. Stemming and handling contractions, i.e., normalising the text before querying and indexing may help retrieval. Thirdly, the assumption of a word having both suffixal and prefixal segments may have been working well only on words of this structure, i.e., ndi-ko-thandiz-a-di.

TABLE 7.18: Summary results : Average nDCG@10 Results for all queries.

	baseline	**generic**	**ngram**	**specific**
Chichewa				
monolingual	**0.517**	0.4561	0.4577	0.4395
Cross-language	**0.2861**	0.2606	0.2497	0.1973
multilingual	**0.518**	0.4551	0.464	0.4288
Citumbuka				
monolingual	0.2936	0.2586	**0.3216**	0.2512
Cross-language	**0.2122**	0.1883	0.2054	0.197
multilingual	0.3208	0.3036	**0.333**	0.278

Summary of Results: Table 7.18 provides a summary of results based on nDCG@10 scores for all the experiments. These results show that applying no stemming for Chichewa produced better results that the other approaches. The Citumbuka experiments provides a different trend – the use of n-grams produced better results (although the table shows that words for cross-lingual retrieval has better performance, n-gram nDCG@1000 has better results than the no baseline.).

7.3 Discussion

Stemming is a widely used approach for text normalisation to improve retrieval performance, especially recall. We have proposed a multilingual stemmer that uses common structural features of Bantu closely related languages. We conducted retrieval experiments involving two languages: Citumbuka and Chichewa. We compared our results with other approaches, namely: tri-grams, language specific stemmers, and using words as they appear in the corpus as the baseline. Evaluation of these stemmers against the baseline shows

mixed outcome; the baseline performed better in runs involving Chichewa corpus. However, improvements were realised with Citumbuka queries on corpus with Chichewa and Citumbuka documents and Citumbuka documents only. The observed trend of performance is not new for stemming. Previous work has found mixed results with regards to stemming in retrieval. Stemming in retrieval has been found to improve retrieval effectiveness for morphologically rich languages (e.g., for cases of Finnish and German)(Hollink et al., 2004). However, some studies have shown that stemming improves recall while at the same time hurting precision, or improving some queries while hurting other queries (Hollink et al., 2004; Krovetz, 1993; Harman, 1991). The latter has been observed in our experimentation; word-based runs were slightly better or comparable to runs with some form of stemming but more documents were retrieved in runs that involved some stemming; and performance improved as more documents were being retrieved.

Overall, our results were skewed towards corpus differences. It is worthy noting that differences in performance were seen in corpora of different languages using equivalent translated queries. We noted that the baseline performed better in situations where the query and document terms perfectly match, especially with named entities. This could be one of the factors that contributed to the stemmed approaches having little or no improvement. Using character tri-grams performed slightly better than the other two methods. This is not surprising as this is more similar to the baseline – parts of the words are not thrown away and likelihood of matching is higher with spelling variations. Our Chichewa topics may have ended up with the given choice of words due to the procedure used in query formulation. In the query formulation step, queries were selected if there were at least five relevant documents and this may have caused the assessors to formulate queries with named entities or words exactly matching the document words. In addition, the corpus is made up of several sets of documents from different domains such as health, religion and news bulletins. Such domains contain content that is specific, such as people, events, diseases and medical treatment. The Citumbuka corpus had fewer documents and was composed of religious and Wikipedia text. Many of the queries were originally formulated in Chichewa. This meant that many of the relevant documents were not seen in the query translation step. Translations may have introduced some variation in the queries especially for queries with no named entities. For instance, the worst performing queries in Citumbuka performed better in Chichewa. The proposed approach performed well on terms with both the prefixal and suffixal segments (additional suffixes and the final vowel). This may not have an advantage on named entities. Therefore, we recommend our method on retrieval that is not domain specific but where people will look for information for general to specific topics.

Language independent methods, such as statistical methods, provide opportunities to improve retrieval performance without implementing language specific rules. Previous work using statistical methods have shown different results dependent on the language.

For example, little improvement was registered when statistically derived rules were used in stemming for Italian (Bacchin, Ferro, and Melucci, 2005). The use of n-grams investigated on eight European languages showed slight as well as significant improvements for within language retrieval when compared with word-based baseline (Hollink et al., 2004) while worse performance was seen in some languages. We have observed the same trend in our results – n-grams provided some improvements in the within language retrieval but performance went down in multilingual retrieval. Overall, results based on the use of n-grams in several studies have shown to improve retrieval effectiveness, especially for languages with simple morphology (McNamee, Nicholas, and Mayfield, 2009; Hollink et al., 2004; Snajder and Basic, 2009). Our results are similar to these previous studies; the n-gram approach improved performance for queries with proper nouns and general terms, and retrieved the most number of relevant documents.

Our assumption on the word structure may have some effects on the results. Therefore, performance of stemming in terms of quality of language specific rules may have affected our results. In addition, the affixes used in the experiments were those extracted from the corpus using our proposed affix learning approach. These affixes were only 66% accurate. The language specific stemmers did not have any advantage in our experiments as well because the affixes used were not those seen in the corpus but those generally known to be affixes from syntax studies of the languages (Mchombo, 2004). Therefore, there are some aspects in our approach that may be explored further, although there is no direct approach to investigate why the stemming approach is giving certain results in retrieval other than retrieval evaluation itself. Possible approaches would be to investigate language specific stemming where only inflectional morphology is considered for Chichewa (inflectional morphology is largely prefixal in Chichewa).

7.4 Summary

In this chapter, we have investigated the impact of using a multilingual stemming method based on morphology similarities of the languages on retrieval effectiveness. We first presented our approach of stemming and its design. We followed this with results of evaluation of the proposed method in retrieval involving Chichewa and Citumbuka. The results for the proposed stemming approach is competitive with n-gramming. However, using tokens as they appeared in the corpora had better performance for most of the experiments using Chichewa. Overall, the proposed approach had performance between n-gram and the word-based approach. In most of the cases, the results were not different from those obtained using words as they appeared in the corpus. This is attributed to the presence of named entities in the queries as well as using words in the queries that directly appear in the corpus.

Chapter 8

Conclusion

Speakers of RSLs struggle to find information written in their own languages due to the limited amounts of published content in those languages, as well as lack of tools and methods tailored for resource constrained contexts. The goal of this thesis was to investigate whether providing search results written in related languages might help RSL speakers to complete their search tasks and improve their search experience. To better understand the problem and to evaluate the impact of our proposed solution, the study was conducted in three stages. Firstly, we investigated user preferences and behaviour when interacting with search results written in related languages. This was done to understand the usefulness of results presented in related languages, as well as user ranking preferences and emotional episodes when interacting with such kind of search results. In the second stage, we investigated the ranking of related languages search results based on relevance and intelligibility criteria, using both supervised and unsupervised methods to improve quality of search results. Thirdly, we investigated the use of multilingual stemming that uses morphological similarities of related languages to overcome challenges of limited or no readily available morphological normalisation tools. User perspectives in interacting with search results written in related languages, as well as stemming results, contributed to the formulation and design of ranking approaches used in our investigation and the overall thesis – whether results written in related languages improve retrieval effectiveness and search experience of users searching using RSLs.

The chapter provides a summary of our findings reported in Chapters 5 , 6 and 7. Particularly, we highlight the major findings of our work in terms of user behaviour when interacting with related languages search results in Section 8.1.1, re-ranking of search results using relevance and intelligibility features in Section 8.1.2 and multilingual stemming in Section 8.1.3. Further, we outline the contributions the thesis has made in Section 8.2 and describe possible future directions of this research in Section 8.3.

8.1 Summary of Results

8.1.1 User Perspectives

In this thesis, we explored user interaction behaviour in a retrieval scenario where intercomprehension is expected for the user to complete a search task using search results written in related languages. In particular, we conducted a user study in the form of controlled laboratory experiments with the following aims: i) to understand whether users would find search results written in related languages useful – whether users would be able to complete search tasks using content in related languages; ii) to understand the user preferences for ranking search results written in related languages; and iii) to explore what emotions users would experience in a search session involving search results written in related languages. We also investigated interface features that are appropriate for such search system interaction using a User Centered Design (UCD) approach. Our research on user interaction with search results in related languages answered the following questions:

RQ1 Are search results written in closely related languages useful to the user?

RQ2 What are the ranking preferences of users for search results written in related languages with varying intelligibility? Does intelligibility matter in the rank preference of such results?

RQ3 What types of emotions do users experience when interacting with search results that require intercomprehension?

RQ4 What is the appropriate presentation style for related languages search results.

Concerning usefulness of search results written in related languages, our findings from the user study in Section 5.1.8 suggest that search results written in closely related languages are useful to the user when relevant content is limited. Participants completed search tasks using their own languages. In cases where relevant search results were written in related languages, participants used documents written in a language most closely related to their L_1 language. However, when the distance between the language of the relevant document and the participant's L_1 increased, success rate of task completion decreased. The direct implication of these results is that search systems may present search results written in related languages, but there is a threshold of intelligibility for users to successfully complete their tasks.

Our results on user preferences for ranking documents written in related languages (presented in Section 5.1.8) indicate that participants prefer documents that are relevant and more comprehensible to them to be ranked highly. In search tasks where multiple documents were relevant, participants ranked highly documents that were more comprehensible

to them. Participants ranked highly relevant documents, and not irrelevant documents written in their own languages. Participants ranked documents written in other languages that were relevant highly than documents that were not relevant. Therefore, our observation on these findings is that users' ranking preference of such search results is based primarily on relevance, and that intelligibility is used as a secondary criterion.

With respect to the type of emotional episodes experienced during search sessions involving related languages, our results in Section 5.1.8 indicate that users experience positive emotions when they are able to complete tasks without struggling, and negative emotions surface when intelligibility of the relevant documents decreases. Participants reported positive emotions in tasks that were simple and relevant documents were written in languages more intelligible to them. As the intelligibility of the relevant documents decreased, more participants experienced negative emotions. Participants who experienced positive emotions while interacting with less intelligible documents indicated that they were happy and at the same time surprised that they could understand content written in another language. Our interpretation of this type of behaviour is that all participants struggled to complete tasks using less intelligible relevant documents, but participants who reported positive emotions were relieved after completing the tasks and their frustration feelings went away.

Our UCD approach to design appropriate user interfaces with associated features involved users throughout the design process (Section 5.2). Participants were involved in the various steps of the process from interface requirements specification, initial mock-up interfaces to the final interface design and its evaluation. The final search result presentation layout interface consisted of a combination of tabbed and mixed merged results layout, which provided the user with the possibility to switch depending on the retrieved results. Our understanding of the proposed interaction style for results written in related languages were guided by human-computer-interaction principles such as simplicity, flexibility and familiarity.

8.1.2 Ranking

The thesis proposed to provide users with results written in related languages and our user study in this regard provided insights on user preference for ranking in this context (i..e, using relevance as a primary criterion, and intelligibility as a secondary criterion). This observation strengthened our proposal to investigate the following research question:

RQ5 Does re-ranking of search results based on relevance and intelligibility improve retrieval effectiveness?

Our ranking problem was to find the best combination of relevance and intelligibility features that matches with user ranking preferences, i.e., given a list or a vector of relevance

and intercomprehension features for documents, a ranking function should map these features to ranks. As described in Section 2.1.3, we used both supervised and unsupervised approaches to answer this question. The supervised approach is modelled as an LTR problem, i.e., automatically constructing a ranking model using training data, i.e., examples of queries, associated document features and user preferred ranking, such that the model can rank documents for queries according to user preferences effectively. Our unsupervised approach used weighted sum to combine a relevance and an intelligibility feature, i.e., normalised BM25 and Cosine similarity.

Overall, our results, presented in Section 6.3, showed improvements when intelligibility features are used with relevance features to rank search results written in related languages. Significant improvements ($\alpha = 0.05$) from the unsupervised baselines (BM25 and normalised BM25) were achieved using the weighted approach. While improvements in performance were noted with the supervised model, significant levels of improved performance ($\alpha = 0.05$) was only achieved by models using relevance features and additional intelligibility features, namely, perplexity and character tri-gram correlation score. The improvements achieved by the model using all features of relevance and intelligibility were only significant for ($\alpha = 0.1$). The limitations contributing to these results are discussed in Section 6.4.

8.1.3 Stemming

Closely related languages within the Bantu Zone N family share some similarities. Our thesis investigated whether using these similarities, by capturing common similarities as rules, could be used to build a stemmer based on affixes derived from the language corpus. We evaluated this proposed approach to answer the following research question:

RQ6 How do multilingual stemmers that use structural similarities of Chichewa and Citumbuka compare in terms of retrieval effectiveness with no stemming, language independent stemming using character tri-grams, and language specific stemming?

Our results for evaluation of multilingual stemming with the other three stemming methods namely, word-based, language-specific, and tri-gramming, show that the performance is language and corpus dependent (Section 7.3). Using words as they appear in the corpus performed slightly higher than the other three methods in monolingual retrieval for Chichewa, but the differences were not significant. The use of n-gram had better performance for the monolingual retrieval for Citumbuka. However, the results were significant with the language specific stemmer and generic stemmer. Word based retrieval performed better than the other three approaches in both runs of cross language retrieval, but the differences were not significant. For multilingual retrieval involving Chichewa queries, the word-based retrieval had the best performance and the results were significant for the generic and

language specific stemmers. The n-gram run for multilingual retrieval using Citumbuka queries had the best performance but the differences were not significant. Word-based approach performed better on Chichewa corpus while the n-gram performed better on Citumbuka queries and corpus. Investigating how the multilingual generic stemmer performed shows that it was the second best in performance, twice following the word-based approach and once following the n-gram approach. In the other cases, the approach was third twice and last once.

In summary, performance of the multilingual generic approach tends to be between the performance of the word-based approach and n-gram approach. Query level analysis showed that the stemmer behaviour was influenced by many queries having named entities, which did not favour stemming – methods that took away some of the word material were disadvantaged although they were able to retrieve more relevant documents than the word-based approach. The generic approach therefore was the middle ground between the word-based approach and the other two methods as it worked as a weak stemmer that strips away only a few affixes from the words. The next section discusses the specific contributions our thesis has made.

8.2 Contributions

The major contribution of this thesis is in providing evidence that using language similarities in retrieval improves retrieval effectiveness for search in related languages. In particular, our research work contributes the following to the advancement of knowledge in the area of IR:

1. A dataset for conducting LTR experiments in less studied languages. We have developed and made available language datasets in Chichewa, Citumbuka, Cinyanja, Cisena and Citonga languages. This dataset comprises 13,627 files in XML format, as well as traditional relevance features and intelligibility features. The dataset can be used as a teaching tool as well as a resource for further research. Developing language datasets, especially for resource scarce languages, is labour intensive and expensive, and we consider our dataset an important contribution to the research community in IR.

2. User perspectives on struggling search where related languages search results are presented to the user. Our work provides the following insights in this context:

   (a) Search results written in related languages are useful when relevant results are limited in resource constrained settings. This finding provides an important consideration for implementation of inclusive and progressive search systems, as this would aid efforts to reduce constraints to information access.

(b) Users prefer relevant results to be highly ranked and intelligibility to be used as a secondary criterion when relevant resource consists of documents in multiple unfamiliar languages including related languages. Again, this finding would provide a useful reference point for designing and implementing ranking algorithms for multilingual search systems, especially for RSLs.

(c) Users struggle to complete tasks as intelligibility of their native language and the document language goes down. This finding is a useful guide particularly for service providers and policy makers, especially where dissemination of critical information is concerned: information in related languages will only be useful to an extent, and efforts to provide critical information in all languages should be enhanced.

(d) As intelligibility between language of the query and document goes down, more users experience negative emotions. This finding is important for effective design of search systems, especially for building system features that can enable users to more easily find and comprehend information that is presented to them in different but related languages.

3. Interface design with appropriate features for interacting with research results written in related languages. Our insights based on evaluation of a Web interface for presentation of related language results will provide guidance for improvement of search systems that aim to support speakers of RSLs.

4. We have shown that intelligibility prediction is possible using linguistics features only and fair accuracy is achievable. With respect to intelligibility, we also used new features that have not been used before for intelligibility, such as n-gram cosine similarity, Kullback-Leibler divergence and Jensen-Shannon divergence. These features demonstrated high prediction power for intelligibility.

5. Our study on re-ranking of search results written in related languages has shown that adding intelligibility features to relevance features both in LTR and unsupervised models improves retrieval effectiveness. This finding is important for improvement of ranking algorithms to improve retrieval effectiveness of search results for RSL speakers.

6. We have developed a string similarity algorithm that improves clustering quality of morphologically related languages. This algorithm is particularly useful for improving tasks of creating morphological paradigms that can be used in morphological word segmentation and stemming tools for under-resourced languages.

7. We have developed and evaluated a stemming approach based on shared word structure of Citumbuka and Chichewa. Our approach is the first step in building multilingual tools for related languages, which can cut costs for developing tools for every language.

Overall, we have shown in this thesis that presenting users with results written in related languages can improve retrieval quality and assist users to complete their search tasks. We have also provided insights on user behaviour and perspectives when interacting with search results written in related languages. Further, we have shown that ranking models that use relevance and intelligibility features for related languages retrieval can improve retrieval quality. Finally, we have proposed and evaluated morphological processing approaches, including a multilingual stemming algorithm that uses rules derived from common morphological features across the Bantu language family. Intelligibility of languages is a difficult attribute to model effectively – both linguistic (e.g., vocabulary, morphology, phonology and phonetics, and syntax) and extra-linguistic (e.g., a speaker's prior language knowledge and experience, and perceptions) factors affect intelligibility. We have shown that it is possible to automatically classify intelligibility with a good accuracy. Reading results written in another language, other than the languages one is most familiar with, may be unpleasant, and some users may be frustrated with such results. Although presenting users with search results written in related languages has some challenges for both users and system designers, there are still some opportunities to explore further, which will assist speakers of RSLs to find relevant information more effectively. The remainder of the chapter presents future directions (Section 8.3) of the work presented in this thesis.

8.3 Future Research Directions

In this section, we provide new avenues for improving and extending our work:

8.3.1 Multi-Objective Optimisation

The thesis has proposed to integrate intelligibility when ranking documents written in related languages. However, user preference of intelligibility and relevance are different, with the latter having higher precedence. The weighted sum approach used to combine BM25 and cosine similarity scores used this evidence to calculate the weights assigned to each objective. Our supervised approach used relevance and intelligibility features uniformly without any weighting on the features (i.e., relevance and intelligibility features could have the same level of prediction power depending on the model that has been learnt). This was a challenge as the dataset was too small to allow a model to learn such a complex relationship. The deployment of more sophisticated multi-objective optimisation techniques

that maximise both relevance and intelligibility may lead to improved retrieval effectiveness similar to (Gollapudi and Sharma, 2009; Dai, Shokouhi, and Davison, 2011; Dong et al., 2010; Wang, Lin, and Metzler, 2010). Unfortunately, large scale data sets are required for such multi-objective methods to be effective.

More work remains to be done to understand how people apply intercomprehension and what factors can genuinely predict intelligibility between languages. Therefore, another approach would be to conduct more large scale user experiments to learn user behaviour, especially to establish thresholds for intelligibility, as well as to identify tasks that could easily be completed using intercomprehension. Such knowledge could then be used as constraints in the re-ranking process after optimising for relevance. Similar work by Papini and Diligenti (2012) may provide a starting point to explore this approach in the context of intelligibility. In this approach, the multi-objective problem is solved by maximising relevance subject to additional constraints specified as prior domain knowledge.

8.3.2 Search Results Personalization

In this thesis, we have proposed to use relevance and intelligibility to rank documents written in related languages. Our approach assumed uniform intelligibility across monolingual individuals speaking the same language as L_1 – we used language features to estimate how intelligible two languages would be without speakers having any prior knowledge of the other language. However, studies in linguistics have shown that intelligibility between languages may depend on other extra-linguistic factors such as language contact, learning and other social factors like attitude towards a language, which may be subjective experiences and traits. Therefore, interaction behaviour may be different across individuals and may depend on their prior language experiences and other personal factors. Thus, learning the individual behaviour of users in interacting with these results may assist to determine whether a specific user may find a document written in a particular related language useful. The results presented to the user in this context could then depend on their historical behaviour towards search results written in related languages, e.g., user click behaviour towards search results written in related languages. Using varying search behaviour across individuals to tailor search results may improve overall user experience.

8.3.3 Cost and Benefits based Analysis

The thesis investigated the problem of introducing intelligibility in ranking by focusing on methods that can combine relevance and intelligibility features, such as LTR. Unfortunately, the approach does not provide a theoretical model that can be used to quantify the interplay between relevance and intelligibility. Using cost and benefits analysis may provide a

theoretical model to account for the gain and cost of interacting with results of varying intelligibility and relevance. For each item in the result-set, an intercomprehension cost and information gain could be assigned, and these could be used to rank documents. Intercomprehension cost in this sense is the amount of effort a user may need to comprehend a document in a related language, while information gain is the usefulness of a document. The work of Azzopardi et al. (2019) may provide guidance on how such an approach may be applied in this context.

8.3.4 Real Time Intelligibility

Our approach relied on a pre-defined relationship between languages – we extracted features of the involved languages to estimate how intelligible the languages were. Unfortunately, the approach is static and cannot work on other languages that have not been seen. A more dynamic approach would be to retrieve documents based on intelligibility calculated dynamically without prior knowledge of the languages involved. The relevance of documents may be estimated from patterns of exact matches and the similarity of languages may be based on available online corpora of the languages using simple metrics such as cosine similarity.

8.3.5 Equivalent queries

Our approach for estimating relevance is the use of exact matching of character patterns (e.g., using n-grams and words as they appear in the corpus of documents and queries) and using traditional IR approaches such as BM25 and language models. Since untranslated queries were used, the similarity scores of highly relevant documents would be very low. Another approach of identifying relevant documents is to use interaction information of users of related languages. The search system could keep track of documents that are deemed to be relevant. Query variations could also be used to study what documents are relevant by learning from user clicks from other languages. The retrieval system could return results independent of the person's specific query but incorporating intelligibility when ranking the search results. However, this is only possible if either the query intent is known or if there is a way of identifying equivalent queries across languages. Similar work on syntactic query variation within languages can be a point of departure for this approach (Bailey et al., 2017). In this way, search engines would be consistent in returning relevant documents especially for languages with limited resources. Additionally, language resources from other languages may help to provide triangulation based translation. Also, user behaviour of other native languages could help in understanding how intelligibility plays in document preference for search results written in related languages.

8.3.6 Collaborative Search

Our thesis is motivated by lack of resources and we have investigated the use of language similarities to provide users with more results. Perhaps, the use of collaborative search can be used to find information in other closely related languages and local users can help with the translation of the information. This would be necessary especially for critical information such as health related information, which would require full understanding of the information.

8.4 Final Remarks

Most of the Web is published in languages not accessible to many potential users, and consequentially, speakers of RSLs struggle to find information on the Web. As more and more data is being published everyday especially in already dominant languages, the gap continues to grow. The challenge of resources for RSLs especially for Bantu languages has implications on the type of research that can be done for these languages. The limitation of data means that RSLs research work is done on unfair ground – big versus no data. Several research directions may provide progress for retrieval of RSLs. First, new creative methods need to be used to develop content for RSLs. For example, local languages are widely used for communication on social media and these avenues may be explored to encourage speakers to contribute or assess content generated automatically using other tools. Second, investigating of topics requiring more data such as neural IR may use resources developed for well-resourced languages resources. In this case transfer learning may be explored to benefit under-resourced languages. Research focusing on specific aspects of RSLs may require new metrics to accommodate the challenges of data scarcity. Moreover, little retrieval improvements in RSLs context is important as long as it improves user satisfaction. Finally, an important aspect that needs to be considered in IR evaluation for RSLs is whether the approaches improve user satisfaction or assist users to complete their tasks. Evaluation metrics are proxies and improving these metrics may not always translate to improve user satisfaction. Our gold standard should be user satisfaction.

Bibliography

Adamson, George W. and Jillian Boreham (1974). "The use of an association measure based on character structure to identify semantically related pairs of words and document titles." In: *Information Storage and Retrieval* 10.7-8, pp. 253–260. URL: http://dblp.uni-trier.de/db/journals/ipm/ipm10.html\#AdamsonB74.

Adomavicius, Gediminas, Nikos Manouselis, and YoungOk Kwon (2011). "Multi-Criteria Recommender Systems". In: *Recommender Systems Handbook*. Ed. by Francesco Ricci et al. Boston, MA: Springer US, pp. 769–803. ISBN: 978-0-387-85820-3. DOI: 10.1007/978-0-387-85820-3_24. URL: https://doi.org/10.1007/978-0-387-85820-3_24.

Amati, Gianni and Cornelis Joost Van Rijsbergen (Oct. 2002). "Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness". In: *ACM Trans. Inf. Syst.* 20.4, pp. 357–389. ISSN: 1046-8188. DOI: 10.1145/582415.582416. URL: http://doi.acm.org/10.1145/582415.582416.

Arapakis, Ioannis, Joemon M. Jose, and Philip D. Gray (2008). "Affective Feedback: An Investigation into the Role of Emotions in the Information Seeking Process". In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '08. Singapore, Singapore: ACM, pp. 395–402. ISBN: 978-1-60558-164-4. DOI: 10.1145/1390334.1390403. URL: http://doi.acm.org/10.1145/1390334.1390403.

Aslam, Javed A. and Mark Montague (2001). "Models for Metasearch". In: *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '01. New Orleans, Louisiana, USA: ACM, pp. 276–284. ISBN: 1-58113-331-6. DOI: 10.1145/383952.384007. URL: http://doi.acm.org/10.1145/383952.384007.

Azzopardi, Leif et al. (2019). "Building Economic Models and Measures of Search". In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*. Ed. by Benjamin Piwowarski et al. ACM, pp. 1401–1402. DOI: 10.1145/3331184.3331379. URL: https://doi.org/10.1145/3331184.3331379.

Bacchin, Michela, Nicola Ferro, and Massimo Melucci (2002). "The Effectiveness of a Graph-Based Algorithm for Stemming". In: *Proceedings of the 5th International Conference on Asian Digital Libraries: Digital Libraries: People, Knowledge, and Technology*. ICADL '02. Berlin,

Heidelberg: Springer-Verlag, pp. 117–128. ISBN: 3-540-00261-8. URL: http://dl.acm.org/citation.cfm?id=646228.681543.

Bacchin, Michela, Nicola Ferro, and Massimo Melucci (Jan. 2005). "A Probabilistic Model for Stemmer Generation". In: *Inf. Process. Manage.* 41.1, pp. 121–137. ISSN: 0306-4573. DOI: 10.1016/j.ipm.2004.04.006. URL: http://dx.doi.org/10.1016/j.ipm.2004.04.006.

Bailey, Peter et al. (2017). "Retrieval Consistency in the Presence of Query Variations". In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '17. Shinjuku, Tokyo, Japan: Association for Computing Machinery, 395–404. ISBN: 9781450350228. DOI: 10.1145/3077136.3080839. URL: https://doi.org/10.1145/3077136.3080839.

Bangor, Aaron, Philip T. Kortum, and James T. Miller (2008). "An Empirical Evaluation of the System Usability Scale". In: *International Journal of Human–Computer Interaction* 24.6, pp. 574–594. DOI: 10.1080/10447310802205776. eprint: https://doi.org/10.1080/10447310802205776. URL: https://doi.org/10.1080/10447310802205776.

Baroni, Marco, Johannes Matiasek, and Harald Trost (2002). "Unsupervised Discovery of Morphologically Related Words Based on Orthographic and Semantic Similarity". In: *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning - Volume 6*. MPL '02. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 48–57. DOI: 10.3115/1118647.1118653. URL: https://doi.org/10.3115/1118647.1118653.

Barron, F.Hutton and Bruce E. Barrett (1996a). "The efficacy of SMARTER — Simple Multi-Attribute Rating Technique Extended to Ranking". In: *Acta Psychologica* 93.1. Contributions to Decision Making II, pp. 23 –36. ISSN: 0001-6918. DOI: https://doi.org/10.1016/0001-6918(96)00010-8. URL: http://www.sciencedirect.com/science/article/pii/0001691896000108.

Barron, Hutton and Bruce Barrett (1996b). "Decision Quality Using Ranked Attribute Weights". In: *Management Science* 42.11, pp. 1515–1523.

Bartell, Brian Theodore (1994). "Optimizing Ranking Functions: A Connectionist Approach to Adaptive Information Retrieval". UMI Order No. GAX94-14751. PhD thesis. La Jolla, CA, USA.

Bayley, Robert et al. (Jan. 2013). *Experimental Methods for Measuring Intelligibility of Closely Related Language Varieties*. URL: http://www.oxfordhandbooks.com/view/10.1093/oxfordhb/9780199744084.001.0001/oxfordhb-9780199744084-e-10.

Bhat, Suma (2013). "Statistical stemming for Kannada". In: *The 4th Workshop on South and Southeast Asian NLP (WSSANLP), WSSANLP-2013*, pp. 25–33. DOI: doi=10.1.1.401.2223.

Boytsov, Leonid, Anna Belova, and Peter Westfall (2013). "Deciding on an Adjustment for Multiplicity in IR Experiments". In: *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '13. Dublin, Ireland: Association for Computing Machinery, 403–412. ISBN: 9781450320344. DOI: 10.1145/2484028.2484034. URL: https://doi.org/10.1145/2484028.2484034.

Breiman, L. et al. (1984a). *Classification and Regression Trees*. new edition **cart93**? Monterey, CA: Wadsworth and Brooks.

Breiman, L. et al. (1984b). *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks.

Brooke, John (1996). "SUS: a "quick and dirty'usability". In: *Usability evaluation in industry*, p. 189.

Buckley, Chris et al. (2000). "Using clustering and superconcepts within SMART: TREC 6". In: *Information Processing & Management* 36.1, pp. 109–131.

Burges, Chris et al. (2005). "Learning to Rank Using Gradient Descent". In: *Proceedings of the 22Nd International Conference on Machine Learning*. ICML '05. Bonn, Germany: ACM, pp. 89–96. ISBN: 1-59593-180-5. DOI: 10.1145/1102351.1102363. URL: http://doi.acm.org/10.1145/1102351.1102363.

Burges, Christopher J. C. (2010). *From RankNet to LambdaRank to LambdaMART: An Overview*. Tech. rep. Microsoft Research. URL: http://research.microsoft.com/en-us/um/people/cburges/tech_reports/MSR-TR-2010-82.pdf.

Burges, Christopher J. C., Robert Ragno, and Quoc Viet Le (2006). "Learning to Rank with Nonsmooth Cost Functions". In: *Proceedings of the 19th International Conference on Neural Information Processing Systems*. NIPS'06. Canada: MIT Press, pp. 193–200. URL: http://dl.acm.org/citation.cfm?id=2976456.2976481.

Callan, James P., Zhihong Lu, and W. Bruce Croft (1995). "Searching Distributed Collections with Inference Networks". In: *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '95. Seattle, Washington, USA: ACM, pp. 21–28. ISBN: 0-89791-714-6. DOI: 10.1145/215206.215328. URL: http://doi.acm.org/10.1145/215206.215328.

Callan, Jamie (2000). "Distributed Information Retrieval". In: *Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval*. Ed. by W. Bruce Croft. Boston, MA: Springer US, pp. 127–150. ISBN: 978-0-306-47019-6. DOI: 10.1007/0-306-47019-5_5. URL: https://doi.org/10.1007/0-306-47019-5_5.

Can, Burcu and Suresh Manandhar (2010). "Clustering Morphological Paradigms Using Syntactic Categories". In: *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*. Ed. by Carol Peters et al. Berlin,

Heidelberg: Springer Berlin Heidelberg, pp. 641–648. ISBN: 978-3-642-15754-7. DOI: 10.1007/978-3-642-15754-7_77.

Cao, Yunbo et al. (2006). "Adapting Ranking SVM to Document Retrieval". In: ACM, pp. 186–193. URL: https://www.microsoft.com/en-us/research/publication/adapting-ranking-svm-document-retrieval/.

Cao, Zhe et al. (2007). "Learning to Rank: From Pairwise Approach to Listwise Approach". In: *Proceedings of the 24th International Conference on Machine Learning*. ICML '07. Corvalis, Oregon, USA: ACM, pp. 129–136. ISBN: 978-1-59593-793-3. DOI: 10.1145/1273496.1273513. URL: http://doi.acm.org/10.1145/1273496.1273513.

Capstick, Joanne et al. (1998). "MULINEX: Multilingual Web Search and Navigation". In: *Proceedings of the 14th Twente Workshop on Language Technology (TWLT 14). Language Technology in Multimedia Information Retrieval*. o.A.

Capstick, Joanne et al. (Jan. 2000). "A System for Supporting Cross-lingual Information Retrieval". In: *Inf. Process. Manage.* 36.2, pp. 275–289. ISSN: 0306-4573. DOI: 10.1016/S0306-4573(99)00058-8. URL: http://dx.doi.org/10.1016/S0306-4573(99)00058-8.

Carterette, Ben (2011). "System Effectiveness, User Models, and User Utility: A Conceptual Framework for Investigation". In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '11. Beijing, China: ACM, pp. 903–912. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2010037. URL: http://doi.acm.org/10.1145/2009916.2010037.

Carterette, Ben et al. (2008). "Here or There". In: *Advances in Information Retrieval*. Ed. by Craig Macdonald et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 16–27. ISBN: 978-3-540-78646-7.

Chavula, Catherine and Hussein Suleman (2016). "Assessing the Impact of Vocabulary Similarity on Multilingual Information Retrieval for Bantu Languages". In: *Proceedings of the 8th Annual Meeting of the Forum on Information Retrieval Evaluation*. FIRE '16. Kolkata, India: ACM, pp. 16–23. ISBN: 978-1-4503-4838-6. DOI: 10.1145/3015157.3015160. URL: http://doi.acm.org/10.1145/3015157.3015160.

Chavula, Jean Josephine (2016). "Verbal Derivation and Valency in Citumbuka". PhD thesis. Centre for Linguistics, Leiden University.

Chen, Aitao and Fredric C. Gey (2004). "Combining Query Translation and Document Translation in Cross-Language Retrieval". In: *Comparative Evaluation of Multilingual Information Access Systems*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 108–121. ISBN: 978-3-540-30222-3.

Chew, Peter A. and Ahmed Abdelali (2008). "The Effects of Language Relatedness on Multilingual Information Retrieval: A Case Study With Indo-European and Semitic Languages." In: *IJCNLP*, pp. 1–9.

Chu, Peng and Anita Komlodi (2017). "TranSearch: A Multilingual Search User Interface Accommodating User Interaction and Preference". In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '17. Denver, Colorado, USA: ACM, pp. 2466–2472. ISBN: 978-1-4503-4656-6. DOI: 10.1145/3027063.3053262. URL: http://doi.acm.org/10.1145/3027063.3053262.

Cleverdon, Cyril W. (1991). "The Significance of the Cranfield Tests on Index Languages". In: *Proceedings of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '91. Chicago, Illinois, USA: Association for Computing Machinery, 3–12. ISBN: 0897914481. DOI: 10.1145/122860.122861. URL: https://doi.org/10.1145/122860.122861.

Collins-Thompson, Kevyn et al. (2011). "Personalizing Web Search Results by Reading Level". In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. CIKM '11. Glasgow, Scotland, UK: Association for Computing Machinery, 403–412. ISBN: 9781450307178. DOI: 10.1145/2063576.2063639. URL: https://doi.org/10.1145/2063576.2063639.

Cosijn, Erica and Peter Ingwersen (July 2000). "Dimensions of Relevance". In: *Inf. Process. Manage.* 36.4, pp. 533–550. ISSN: 0306-4573. DOI: 10.1016/S0306-4573(99)00072-2. URL: http://dx.doi.org/10.1016/S0306-4573(99)00072-2.

Crammer, Koby and Yoram Singer (2001). "Pranking with Ranking". In: *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*. NIPS'01. Vancouver, British Columbia, Canada: MIT Press, pp. 641–647. URL: http://dl.acm.org/citation.cfm?id=2980539.2980623.

Dai, Na, Milad Shokouhi, and Brian D. Davison (2011). "Learning to Rank for Freshness and Relevance". In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '11. Beijing, China: ACM, pp. 95–104. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2009933. URL: http://doi.acm.org/10.1145/2009916.2009933.

Dolamic, Ljiljana and Jacques Savoy (Sept. 2010). "Comparative Study of Indexing and Search Strategies for the Hindi, Marathi, and Bengali Languages". In: 9.3, 11:1–11:24. ISSN: 1530-0226. DOI: 10.1145/1838745.1838748. URL: http://doi.acm.org/10.1145/1838745.1838748.

Dong, Anlei et al. (2010). "Towards Recency Ranking in Web Search". In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. WSDM '10. New York, New York, USA: ACM, pp. 11–20. ISBN: 978-1-60558-889-6. DOI: 10.1145/1718487.1718490. URL: http://doi.acm.org/10.1145/1718487.1718490.

Fischer, Andrea K., Jilles Vreeken, and Dietrich Klakow (2017). "Beyond Pairwise Similarity: Quantifying and Characterizing Linguistic Similarity between Groups of Languages by

MDL". In: *Computación y Sistemas* 21.4. URL: http://www.cys.cic.ipn.mx/ojs/index.php/CyS/article/view/2865.

Flach, Peter A. (2003). "The Geometry of ROC Space: Understanding Machine Learning Metrics through ROC Isometrics". In: *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*. ICML'03. Washington, DC, USA: AAAI Press, 194–201. ISBN: 1577351894.

Ford, Nigel (2015). "Frontmatter". In: *Introduction to Information Behaviour*. Facet, pp. i–iv.

Frakes, William B (1992). *Stemming Algorithms.*

Freund, Yoav et al. (Dec. 2003). "An Efficient Boosting Algorithm for Combining Preferences". In: *J. Mach. Learn. Res.* 4, pp. 933–969. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=945365.964285.

Funnell, Barry John (2004). "A contrastive analysis of two standardised varieties of Sena". PhD thesis. PhD thesis.

Gamallo, Pablo, José Ramom Pichel, and Iñaki Alegria (2017). "From language identification to language distance". In: *Physica A: Statistical Mechanics and its Applications* 484, pp. 152 –162. ISSN: 0378-4371. DOI: https://doi.org/10.1016/j.physa.2017.05.011. URL: http://www.sciencedirect.com/science/article/pii/S0378437117305137.

Gao, Jianfeng et al. (2005). "Linear Discriminant Model for Information Retrieval". In: *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '05. Salvador, Brazil: ACM, pp. 290–297. ISBN: 1-59593-034-5. DOI: 10.1145/1076034.1076085. URL: http://doi.acm.org/10.1145/1076034.1076085.

Gao, Wei et al. (2009). "Joint Ranking for Multilingual Web Search". In: *Advances in Information Retrieval*. Ed. by Mohand Boughanem et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 114–125.

Gaussier, Eric (1999). "Unsupervised Learning of Derivational Morphology from Inflectional Lexicons". In: *Workshop on Supervised Learning in Natural Langauge Processing at the 37th Annual Meeting of the Association for Compuational Linguistics*. Philadephia, PA, pp. 24–30.

Gey, Fredric (2007). "Search Between Chinese and Japanese Text Collections". In: *Proceedings of NTCIR-6 Workshop Meeting*. UC Data Archive and Technical Assistance University of California, Berkeley.

Goldsmith, John (June 2001). "Unsupervised Learning of the Morphology of a Natural Language". In: *Comput. Linguist.* 27.2, 153–198. ISSN: 0891-2017. DOI: 10.1162/089120101750300490 URL: https://doi.org/10.1162/089120101750300490.

Golebiewski, Michael and danah Boyd (2018). *Data Voids: Where Missing Data Can Easily Be Exploited*. Tech. rep. Data & Society, pp. 1–51.

Gollapudi, Sreenivas and Aneesh Sharma (2009). "An Axiomatic Approach for Result Diversification". In: *Proceedings of the 18th International Conference on World Wide Web*. WWW '09. Madrid, Spain: ACM, pp. 381–390. ISBN: 978-1-60558-487-4. DOI: 10.1145/1526709.1526761. URL: http://doi.acm.org/10.1145/1526709.1526761.

Goodkind, Adam and Klinton Bicknell (2018). "Predictive power of word surprisal for reading times is a linear function of language model quality". In: *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics, CMCL 2018, Salt Lake City, Utah, USA, January 7, 2018*, pp. 10–18. URL: https://aclanthology.info/papers/W18-0102/w18-0102.

Gooskens, Charlotte (2013). "Methods for measuring intelligibility of closely related language varieties". English. In: *The Oxford Handbook of Sociolinguistics*. Ed. by R. Bayley, R. Cameron, and C. Lucas. Oxford University Press, pp. 195–213. ISBN: 9780199744084. DOI: 10.1093/oxfordhb/9780199744084.013.0010.

— (2018). "Dialect Intelligibility". In: *The Handbook of Dialectology*. John Wiley & Sons, Ltd. Chap. 11, pp. 204–218. ISBN: 9781118827628. DOI: 10.1002/9781118827628.ch11. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118827628.ch11.

Gooskens, Charlotte and Femke Swarte (2017). "Linguistic and extra-linguistic predictors of mutual intelligibility between Germanic languages". In: *Nordic Journal of Linguistics* 40.2, pp. 123–147. DOI: 10.1017/S0332586517000099.

Greenberg, Joseph H. (1957). "Order of Affixing: a Study in General Linguistics". In: *Essays in Linguistics*. Ed. by Joseph H. Greenberg. Chicago: University of Chicago Press, pp. 86–94.

Guthrie, Malcolm (1967). *Comparative Bantu : an introduction to the comparative linguistics and prehistory of the Bantu languages*. Farnborough : Gregg. ISBN: 0576110000.

Hafer, Margaret A. and Stephen F. Weiss (1974). "Word segmentation by letter successor varieties". In: *Information Storage and Retrieval* 10.11, pp. 371 –385. ISSN: 0020-0271. DOI: https://doi.org/10.1016/0020-0271(74)90044-8. URL: http://www.sciencedirect.com/science/article/pii/0020027174900448.

Hale, John (2001). "A Probabilistic Earley Parser As a Psycholinguistic Model". In: *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*. NAACL '01. Pittsburgh, Pennsylvania: Association for Computational Linguistics, pp. 1–8. DOI: 10.3115/1073336.1073357. URL: https://doi.org/10.3115/1073336.1073357.

Hammarström, Harald and Lars Borin (2011). "Unsupervised learning of morphology". In: *Computational Linguistics* 37.2, pp. 309–350.

Hans, Dua (2006). "Hindustani". In: *Encyclopedia of Language Linguistics (Second Edition)*. Ed. by Keith Brown. Second Edition. Oxford: Elsevier, pp. 309 –312. ISBN: 978-0-08-044854-1.

DOI: https://doi.org/10.1016/B0-08-044854-2/02220-3. URL: http://www.sciencedirect.com/science/article/pii/B0080448542022203.

Harman, Donna (1991). "How effective is suffixing?" In: *Journal of the American Society for Information Science* 42.1, p. 7.

Haspelmath, Martin and Uri Tadmor (2009). *Loanwords in the world's languages a comparative handbook / edited by Martin Haspelmath, Uri Tadmor.* eng. New York, NY: Mouton de Gruyter.

Heeringa, Wilbert et al. (2013). "Lexical and orthographic distances between Germanic, Romance and Slavic languages and their relationship to geographic distance". English. In: *Phonetics in Europe*. Ed. by C. Gooskens and R. van Bezooijen. P.I.E. - Peter Lang, pp. 99–137.

Hollink, Vera et al. (2004). "Monolingual Document Retrieval for European Languages". In: *Information Retrieval* 7.1, pp. 33 –52. DOI: 10.1023/B:INRT.0000009439.19151.4c. URL: https://doi.org/10.1023/B:INRT.0000009439.19151.4c.

Holy, Andreas von et al. (2017). "Bantuweb: A Digital Library for Resource Scarce South African Languages". In: *Proceedings of the South African Institute of Computer Scientists and Information Technologists*. SAICSIT '17. Thaba 'Nchu, South Africa: ACM, 36:1–36:10. ISBN: 978-1-4503-5250-5. DOI: 10.1145/3129416.3129446. URL: http://doi.acm.org/10.1145/3129416.3129446.

Hothorn, Torsten, Kurt Hornik, and Achim Zeileis (2006). "Unbiased Recursive Partitioning: A Conditional Inference Framework". In: *Journal of Computational and Graphical Statistics* 15.3, pp. 651–674.

Hu, Yu et al. (2005). "Refining the SED Heuristic for Morpheme Discovery: Another Look at Swahili". In: *Proceedings of the Workshop on Psychocomputational Models of Human Language Acquisition*. Ann Arbor, Michigan: Association for Computational Linguistics, pp. 28–35. URL: http://www.aclweb.org/anthology/W05-0504.

Hull, David A. (Jan. 1996). "Stemming Algorithms: A Case Study for Detailed Evaluation". In: *J. Am. Soc. Inf. Sci.* 47.1, 70–84. ISSN: 0002-8231.

Hyman, Larry M. (2003). "Suffix ordering in Bantu: a morphocentric approach". In: *Yearbook of Morphology 2002*. Ed. by Geert Booij and Jaap van Marle. Dordrecht: Springer Netherlands, pp. 245–281. DOI: 10.1007/0-306-48223-1_8. URL: http://dx.doi.org/10.1007/0-306-48223-1_8.

Jacquemin, Christian (1997). "Guessing Morphology from Terms and Corpora". In: *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '97. Philadelphia, Pennsylvania, USA: ACM, pp. 156–165. ISBN: 0-89791-836-3. DOI: 10.1145/258525.258557. URL: http://doi.acm.org/10.1145/258525.258557.

Jain, A. K., M. N. Murty, and P. J. Flynn (Sept. 1999). "Data Clustering: A Review". In: *ACM Comput. Surv.* 31.3, pp. 264–323. ISSN: 0360-0300. DOI: 10.1145/331499.331504. URL: http://doi.acm.org/10.1145/331499.331504.

Jann, Ben (Jan. 2008). *Multinomial goodness-of-fit: large sample tests with survey design correction and exact tests for small samples*. ETH Zurich Sociology Working Papers 2. ETH Zurich, Chair of Sociology. URL: https://ideas.repec.org/p/ets/wpaper/2.html.

Järvelin, Anni et al. (2006). "Dictionary-independent translation in CLIR between closely related languages". In: *Proceedings of the 6th Dutch-Belgian Infromation Retrieval Workshop (DIR 2006)*.

Järvelin, Kalervo and Jaana Kekäläinen (2000). "IR Evaluation Methods for Retrieving Highly Relevant Documents". In: *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '00. Athens, Greece: ACM, pp. 41–48. ISBN: 1-58113-226-3. DOI: 10.1145/345508.345545. URL: http://doi.acm.org/10.1145/345508.345545.

Jens, Moberg et al. (2007). "Conditional Entropy Measures Intelligibility among Related Languages". In: 7, pp. 51–66.

Käki, Mika (2005). "Findex: Search Result Categories Help Users When Document Ranking Fails". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. Portland, Oregon, USA: Association for Computing Machinery, 131–140. ISBN: 1581139985. DOI: 10.1145/1054972.1054991. URL: https://doi.org/10.1145/1054972.1054991.

Kang, Changsung et al. (2012). "Learning to Rank with Multi-aspect Relevance for Vertical Search". In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. WSDM '12. Seattle, Washington, USA: ACM, pp. 453–462. ISBN: 978-1-4503-0747-5. DOI: 10.1145/2124295.2124350. URL: http://doi.acm.org/10.1145/2124295.2124350.

Katamba, Francis X. (2003). "Bantu nominal morphology." In: *The Bantu Languages*. Routledge. ISBN: 0700711341.

Kelly, Diane (Jan. 2009). "Methods for Evaluating Interactive Information Retrieval Systems with Users". In: *Found. Trends Inf. Retr.* 3.1—2, pp. 1–224. ISSN: 1554-0669. DOI: 10.1561/1500000012. URL: http://dx.doi.org/10.1561/1500000012.

Kiso, Andrea (2012). "Tense and aspect in Chichewa, Citumbuka and Cisena: A description and comparison of the tense-aspect systems in three southeastern Bantu languages". PhD thesis. Department of Linguistics, Stockholm University.

Kosch, I.M. (2006). *Topics in Morphology in the African Language Context*. Unisa Press. ISBN: 9781868883691. URL: https://books.google.co.za/books?id=e26AoEvgttcC.

Krovetz, Robert (1993). "Viewing Morphology As an Inference Process". In: *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information*

Retrieval. SIGIR '93. Pittsburgh, Pennsylvania, USA: ACM, pp. 191–202. ISBN: 0-89791-605-0. DOI: 10.1145/160688.160718. URL: http://doi.acm.org/10.1145/160688.160718.

Krovetz, Robert and W. Bruce Croft (Apr. 1992). "Lexical Ambiguity and Information Retrieval". In: *ACM Trans. Inf. Syst.* 10.2, pp. 115–141. ISSN: 1046-8188. DOI: 10.1145/146802.146810. URL: http://doi.acm.org/10.1145/146802.146810.

Kullback, Solomon (1959). *Information Theory and Statistics*. John Wiley and Sons, New York.

Kursa, Miron B. and Witold R. Rudnicki (2010). "Feature Selection with the Boruta Package". In: *Journal of Statistical Software* 36.11, pp. 1–13. URL: http://www.jstatsoft.org/v36/i11/.

Kürschner, Sebastian, Charlotte Gooskens, and Renée van Bezooijen (2009). "Linguistic Determinants of the Intelligibility of Swedish Words among Danes". In: *International Journal of Humanities and Arts Computing* 2.1–2, pp. 83–100. DOI: 10.3366/e1753854809000329.

Lebanon, Guy and John D. Lafferty (2002). "Cranking: Combining Rankings Using Conditional Probability Models on Permutations". In: *Proceedings of the Nineteenth International Conference on Machine Learning*. ICML '02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 363–370. ISBN: 1-55860-873-7. URL: http://dl.acm.org/citation.cfm?id=645531.655830.

Lee, Chia-Jung et al. (2015). "An Optimization Framework for Merging Multiple Result Lists". In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. CIKM '15. Melbourne, Australia: ACM, pp. 303–312. ISBN: 978-1-4503-3794-6. DOI: 10.1145/2806416.2806489. URL: http://doi.acm.org/10.1145/2806416.2806489.

Lee, Joon Ho (1997). "Analyses of Multiple Evidence Combination". In: *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '97. Philadelphia, Pennsylvania, USA: ACM, pp. 267–276. ISBN: 0-89791-836-3. DOI: 10.1145/258525.258587. URL: http://doi.acm.org/10.1145/258525.258587.

Levenshtein, V. I. (Feb. 1966). "Binary Codes Capable of Correcting Deletions, Insertions and Reversals". In: *Soviet Physics Doklady* 10, p. 707.

Li, Hang (2011). *Learning to Rank for Information Retrieval and Natural Language Processing*. Morgan & Claypool Publishers. ISBN: 1608457079, 9781608457076.

Li, Ping, Christopher J. C. Burges, and Qiang Wu (2007). "McRank: Learning to Rank Using Multiple Classification and Gradient Boosting". In: *Proceedings of the 20th International Conference on Neural Information Processing Systems*. NIPS'07. Vancouver, British Columbia, Canada: Curran Associates Inc., pp. 897–904. ISBN: 978-1-60560-352-0. URL: http://dl.acm.org/citation.cfm?id=2981562.2981675.

Lillis, David et al. (2006). "ProbFuse: A Probabilistic Approach to Data Fusion". In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '06. Seattle, Washington, USA: ACM, pp. 139–146. ISBN: 1-59593-369-7. DOI: 10.1145/1148170.1148197. URL: http://doi.acm.org/10.1145/1148170.1148197.

Lin, J. (Sept. 2006). "Divergence Measures Based on the Shannon Entropy". In: *IEEE Trans. Inf. Theor.* 37.1, 145–151. ISSN: 0018-9448. DOI: 10.1109/18.61115. URL: https://doi.org/10.1109/18.61115.

Lin, Wen-Cheng and Hsin-Hsi Chen (2003a). "Merging Mechanisms in Multilingual Information Retrieval". In: *Advances in Cross-Language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002 Rome, Italy, September 19–20, 2002 Revised Papers*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 175–186.

Lin, Wen-Cheng and Hsin-Hsi Chen (2003b). "Merging Mechanisms in Multilingual Information Retrieval". In: *Advances in Cross-Language Information Retrieval*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 175–186. ISBN: 978-3-540-45237-9.

Ling, Chenjun, Ben Steichen, and Alexander G. Choulos (2018). "A Comparative User Study of Interactive Multilingual Search Interfaces". In: *Proceedings of the 2018 Conference on Human Information Interaction&Retrieval, CHIIR 2018, New Brunswick, NJ, USA, March 11-15, 2018*, pp. 211–220. DOI: 10.1145/3176349.3176383. URL: http://doi.acm.org/10.1145/3176349.3176383.

Liu, Yu-Ting et al. (2007). "Supervised Rank Aggregation". In: *Proceedings of the 16th International Conference on World Wide Web*. WWW '07. Banff, Alberta, Canada: ACM, pp. 481–490. ISBN: 978-1-59593-654-7. DOI: 10.1145/1242572.1242638. URL: http://doi.acm.org/10.1145/1242572.1242638.

Lopatovska, Irene and Ioannis Arapakis (July 2011). "Theories, Methods and Current Research on Emotions in Library and Information Science, Information Retrieval and Human-computer Interaction". In: *Inf. Process. Manage.* 47.4, pp. 575–592. ISSN: 0306-4573. DOI: 10.1016/j.ipm.2010.09.001. URL: http://dx.doi.org/10.1016/j.ipm.2010.09.001.

Lopatovska, Irene and Hartmut B. Mokros (Jan. 2008). "Willingness to Pay and Experienced Utility As Measures of Affective Value of Information Objects: Users' Accounts". In: *Inf. Process. Manage.* 44.1, pp. 92–104. ISSN: 0306-4573. DOI: 10.1016/j.ipm.2007.01.020. URL: http://dx.doi.org/10.1016/j.ipm.2007.01.020.

Lovins, Julie (1968). *Development of a stemming algorithm*. Vol. 11. 2. Electronic Systems Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Lowe, Ryan and Ben Steichen (2017). "Multilingual Search User Behaviors – Exploring Multilingual Querying and Result Selection Through Crowdsourcing". In: *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*. UMAP '17. Bratislava, Slovakia: ACM, pp. 303–307. ISBN: 978-1-4503-4635-1. DOI: 10.1145/3079628.3079702. URL: http://doi.acm.org/10.1145/3079628.3079702.

Maho, Jouni (2006). *A Classification of the Bantu Languages: An Update of Guthrie's Referential System*. Ed. by D. Nurse and G. Philippson. Routledge Language Family Series. Taylor & Francis. ISBN: 9781135796839.

Maho, Jouni Filip (2007). *The Linear Ordering of TAM/NEG Markers in the Bantu languages*. URL: https://www.soas.ac.uk/linguistics/research/workingpapers/volume-15/file37810.pdf.

Majumder, Prasenjit, Mandar Mitra, and Dipasree Pal (2008). "Bulgarian, Hungarian and Czech Stemming Using YASS". In: *Advances in Multilingual and Multimodal Information Retrieval*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 49–56. ISBN: 978-3-540-85760-0.

Majumder, Prasenjit et al. (Oct. 2007). "YASS: Yet Another Suffix Stripper". In: *ACM Trans. Inf. Syst.* 25.4. ISSN: 1046-8188. DOI: 10.1145/1281485.1281489. URL: http://doi.acm.org/10.1145/1281485.1281489.

Malumba, Nkosana, Katlego Moukangwe, and Hussein Suleman (2015). "Digital Libraries: Providing Quality Information: 17th International Conference on Asia-Pacific Digital Libraries, ICADL 2015, Seoul, Korea, December 9-12, 2015. Proceedings". In: ed. by B. Robert Allen, Jane Hunter, and L. Marcia Zeng. Cham: Springer International Publishing. Chap. AfriWeb: A Web Search Engine for a Marginalized Language, pp. 180–189.

Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to Information Retrieval*. USA: Cambridge University Press. ISBN: 0521865719.

Mao, Jiaxin et al. (2016). "When Does Relevance Mean Usefulness and User Satisfaction in Web Search?" In: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '16. Pisa, Italy: ACM, pp. 463–472. ISBN: 978-1-4503-4069-4. DOI: 10.1145/2911451.2911507. URL: http://doi.acm.org/10.1145/2911451.2911507.

Markov, Ilya, Avi Arampatzis, and Fabio Crestani (2012). "Unsupervised Linear Score Normalization Revisited". In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '12. Portland, Oregon, USA: ACM, pp. 1161–1162. ISBN: 978-1-4503-1472-5. DOI: 10.1145/2348283.2348519. URL: http://doi.acm.org/10.1145/2348283.2348519.

— (2013). "On CORI Results Merging". In: *Proceedings of the 35th European Conference on Advances in Information Retrieval*. ECIR'13. Moscow, Russia: Springer-Verlag, pp. 752–755.

ISBN: 978-3-642-36972-8. DOI: 10.1007/978-3-642-36973-5_76. URL: http://dx.doi.org/10.1007/978-3-642-36973-5_76.

Martínez-Santiago, Fernando, L. Alfonso Ureña-López, and Maite Martín-Valdivia (2006). "A merging strategy proposal: The 2-step retrieval status value method". In: *Information Retrieval* 9.1, pp. 71–93. ISSN: 1573-7659. DOI: 10.1007/s10791-005-5722-4. URL: https://doi.org/10.1007/s10791-005-5722-4.

Matiki, Alfred (2000). "A functional categoriality of adjectives in Chichewa and Chiyao". In: *Journal of Humanities* 14 (1), pp. 48–62.

Matiki, Alfred J (2016). "Patterns Of Lexical Borrowing In Chichewa". In: *Journal of the Linguistics Association of Southern African Development Community Universities* 4.4, pp. 79–93. ISSN: 0306-4573. DOI: 10.1016/S0306-4573(99)00047-3. URL: http://dx.doi.org/10.1016/S0306-4573(99)00047-3.

Mayfield, James and Paul McNamee (2003). "Single N-gram Stemming". In: *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. SIGIR '03. Toronto, Canada: ACM, pp. 415–416. ISBN: 1-58113-646-3. DOI: 10.1145/860435.860528. URL: http://doi.acm.org/10.1145/860435.860528.

Mchombo, S. (2004). *The Syntax of Chichewa*. Cambridge Syntax Guides. Cambridge University Press. ISBN: 9780521573788. URL: https://books.google.co.za/books?id=SRCFoDp88oUC.

Mcnamee, Paul (2008). "N-gram tokenization for Indian language text retrieval". In: *In Working Notes of the Forum for Information Retrieval Evaluation 2008*.

McNamee, Paul, Charles Nicholas, and James Mayfield (2009). "Addressing Morphological Variation in Alphabetic Languages". In: *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '09. Boston, MA, USA: ACM, pp. 75–82. ISBN: 978-1-60558-483-6. DOI: 10.1145/1571941.1571957. URL: http://doi.acm.org/10.1145/1571941.1571957.

Meeussen, Achille Emile (1967). "Bantu grammatical reconstructions". eng. In: ISSN: 2033-8732. DOI: 10.3406/aflin.1967.873. URL: https://www.persee.fr/doc/aflin_2033-8732_1967_num_3_1_873.

Melucci, Massimo (June 2007). "On Rank Correlation in Information Retrieval Evaluation". In: *SIGIR Forum* 41.1, pp. 18–33. ISSN: 0163-5840. DOI: 10.1145/1273221.1273223. URL: http://doi.acm.org/10.1145/1273221.1273223.

Mizzaro, Stefano (1997). "Relevance: The whole history". In: *Journal of the American Society for Information Science* 48.9, pp. 810–832. DOI: 10.1002/(SICI)1097-4571(199709)48:9<810::AID-ASI6>3.0.CO;2-U. eprint: https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291097-4571%28199709%2948%3A9%3C810%3A%3AAID-ASI6%3E3.0.CO%3B2-U. URL: https://asistdl.

onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-4571%28199709%2948%3A9%3C810%3A%3AAID-ASI6%3E3.0.CO%3B2-U.

Mosbach, M et al. (2019). "incom.py - A Toolbox for Calculating Linguistic Distances and Asymmetries between Related Languages". In: *Proceedings of Recent Advances in Natural Language Processing, RANLP 2019, Varna, Bulgaria, 2-4 September 2019*, pp. 811–819.

Mustafa, Mohammed and Hussein Suleman (2011). "Multilingual Querying". In: *Proceedings of the Arabic Language Technology International Conference (ALTIC), Alexandria, Egypt*.

Nakov, Preslav et al., eds. (2017). *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects, VarDial 2017, Valencia, Spain, April 3, 2017*. Association for Computational Linguistics. ISBN: 978-1-945626-43-2. URL: https://aclanthology.info/volumes/proceedings-of-the-fourth-workshop-on-nlp-for-similar-languages-varieties-and-dialects-vardial.

Ng, Kwong Bor (1998). "An Investigation of the Conditions for Effective Data Fusion in Information Retrieval". PhD thesis. Rutgers University: School of Communication, Information, and Library Studies.

Nie, Jian-Yun and Fuman Jin (2003). "A Multilingual Approach to Multilingual Information Retrieval". In: *Advances in Cross-Language Information Retrieval*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 101–110. ISBN: 978-3-540-45237-9.

Nurse, D. (2008). *Tense and Aspect in Bantu*. Oxford linguistics. OUP Oxford. ISBN: 9780199239290. URL: https://books.google.co.za/books?id=lqIUDAAAQBAJ.

Nurse, D. and G. Philippson (2006). *The Bantu Languages*. Routledge Language Family Series. Taylor & Francis. ISBN: 9781135796839.

Nurse, Derek (2001). "A Survey Report for the Bantu Languages". In: *SLI*.

Oard, Douglas W., Gina-Anne Levow, and Clara I. Cabezas (2000). "CLEF Experiments at the University of Maryland: Statistical Stemming and Back-off Translation Strategies". In: *Working Notes for CLEF 2000 Workshop co-located with the 4th European Conference on Digital Libraries (ECDL 2000), Lisbon, Portugal, September 21-22, 2000*. URL: http://ceur-ws.org/Vol-1166/CLEF2000wn-adhoc-OardEt2000.pdf.

Palotti, Joao et al. (2016). "Ranking Health Web Pages with Relevance and Understandability". In: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '16. Pisa, Italy: ACM, pp. 965–968. ISBN: 978-1-4503-4069-4. DOI: 10.1145/2911451.2914741. URL: http://doi.acm.org/10.1145/2911451.2914741.

Papini, Tiziano and Michelangelo Diligenti (2012). "Learning-to-rank with Prior Knowledge as Global Constraints". In: *Workshop on Combining Constraint solving with Mining and Learning (CoCoMiLe)*.

Peter, doye (2005). *Intercomprehension : Reference Study*.

Peters, Carol, Martin Braschler, and Paul D. Clough (2012). *Multilingual Information Retrieval - From Research To Practice*. Springer. ISBN: 978-3-642-23007-3. DOI: 10.1007/978-3-642-23008-0. URL: http://dx.doi.org/10.1007/978-3-642-23008-0.

Petrelli, Daniela (2008). "On the role of user-centred evaluation in the advancement of interactive information retrieval". In: *Information Processing & Management* 44.1. Evaluation of Interactive Information Retrieval Systems, pp. 22–38. ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2007.01.024. URL: http://www.sciencedirect.com/science/article/pii/S0306457307000489.

Petrelli, Daniela et al. (2006a). "Observing users, designing clarity: A case study on the user-centered design of a cross-language information retrieval system". In: *Journal of the American Society for Information Science and Technology* 55.10, pp. 923–934. DOI: 10.1002/asi.20036. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asi.20036. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.20036.

Petrelli, Daniela et al. (Mar. 2006b). "Which User Interaction for Cross-language Information Retrieval? Design Issues and Reflections". In: *J. Am. Soc. Inf. Sci. Technol.* 57.5, pp. 709–722. ISSN: 1532-2882. DOI: 10.1002/asi.v57:5. URL: https://doi.org/10.1002/asi.v57:5.

Plutchik, Robert (1980). "A general psychoevolutionary theory of emotion". In: *Theories of emotion* 1, pp. 3–31.

Ponte, Jay M. and W. Bruce Croft (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. Melbourne, Australia: Association for Computing Machinery, 275–281. ISBN: 1581130155. DOI: 10.1145/290941.291008. URL: https://doi.org/10.1145/290941.291008.

Poretski, Lev, Joel Lanir, and Ofer Arazy (2019). "Feel the image: The role of emotions in the image-seeking process". In: *Human–Computer Interaction* 34.3, pp. 240–277. DOI: 10.1080/07370024.2017.1359604. eprint: https://doi.org/10.1080/07370024.2017.1359604. URL: https://doi.org/10.1080/07370024.2017.1359604.

Porter, M. F. (1997). "Readings in Information Retrieval". In: ed. by Karen Sparck Jones and Peter Willett. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Chap. An Algorithm for Suffix Stripping, pp. 313–316. ISBN: 1-55860-454-5. URL: http://dl.acm.org/citation.cfm?id=275537.275705.

Porter, Martin F (1980). "An algorithm for suffix stripping". In: *Program* 14.3, pp. 130–137.

Powers, David MW (1998). "Applications and explanations of Zipf's law". In: *Proceedings of the joint conferences on new methods in language processing and computational natural language learning*. Association for Computational Linguistics, pp. 151–160.

Qin, Tao and Tie-Yan Liu (2013). "Introducing LETOR 4.0 Datasets." In: *CoRR* abs/1306.2597. URL: http://dblp.uni-trier.de/db/journals/corr/corr1306.html#QinL13.

Qin, Tao et al. (Mar. 2008). "Query-level Loss Functions for Information Retrieval". In: *Inf. Process. Manage.* 44.2, pp. 838–855. ISSN: 0306-4573. DOI: 10.1016/j.ipm.2007.07.016. URL: http://dx.doi.org/10.1016/j.ipm.2007.07.016.

Qin, Tao et al. (Aug. 2010). "LETOR: A Benchmark Collection for Research on Learning to Rank for Information Retrieval". In: *Inf. Retr.* 13.4, pp. 346–374. ISSN: 1386-4564. DOI: 10.1007/s10791-009-9123-y. URL: http://dx.doi.org/10.1007/s10791-009-9123-y.

Robertson, S. E. and S. Walker (1994). "Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval". In: *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '94. Dublin, Ireland: Springer-Verlag New York, Inc., pp. 232–241. ISBN: 0-387-19889-X. URL: http://dl.acm.org/citation.cfm?id=188490.188561.

Robertson, Stephen and Hugo Zaragoza (Apr. 2009). "The Probabilistic Relevance Framework: BM25 and Beyond". In: *Found. Trends Inf. Retr.* 3.4, 333–389. ISSN: 1554-0669. DOI: 10.1561/1500000019. URL: https://doi.org/10.1561/1500000019.

Robertson, Stephen E (1977). "The probability ranking principle in IR". In: *Journal of documentation* 33.4, pp. 294–304.

Roeck, Anne N. de and Waleed Al-Fares (2000). "A Morphologically Sensitive Clustering Algorithm for Identifying Arabic Roots". In: *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. ACL '00. Hong Kong: Association for Computational Linguistics, pp. 199–206. DOI: 10.3115/1075218.1075244. URL: https://doi.org/10.3115/1075218.1075244.

Sakai, Tetsuya (June 2014). "Statistical Reform in Information Retrieval?" In: *SIGIR Forum* 48.1, 3–12. ISSN: 0163-5840. DOI: 10.1145/2641383.2641385. URL: https://doi.org/10.1145/2641383.2641385.

Salton, Gerard and Michael J. McGill (1986). *Introduction to Modern Information Retrieval*. USA: McGraw-Hill, Inc. ISBN: 0070544840.

Sanderson, Mark (2010). "Test Collection Based Evaluation of Information Retrieval Systems." In: *Foundations and Trends in Information Retrieval* 4.4, pp. 247–375. URL: http://dblp.uni-trier.de/db/journals/ftir/ftir4.html#Sanderson10.

Saracevic, Tefko (1975). "RELEVANCE: A review of and a framework for the thinking on the notion in information science". In: *Journal of the American Society for Information Science* 26.6, pp. 321–343. DOI: 10.1002/asi.4630260604. URL: https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/asi.4630260604.

Saracevic, Tefko (1997). "The Stratified Model of Information Retrieval Interaction: Extension and Applications". In: *Proceedings of the ASIST Annual Meeting* 34, p. 313. ISSN: 0044-7870. URL: https://www.learntechlib.org/p/83920.

— (2007). "Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II: nature and manifestations of relevance". In: *Journal of the American Society for Information Science and Technology* 58.13, pp. 1915–1933. DOI: 10.1002/asi.20682. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asi.20682. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.20682.

Savolainen, Reijo (2014). "Emotions as motivators for information seeking: A conceptual analysis". In: *Library Information Science Research* 36.1, pp. 59 –65. ISSN: 0740-8188. DOI: https://doi.org/10.1016/j.lisr.2013.10.004. URL: http://www.sciencedirect.com/science/article/pii/S0740818814000085.

Savoy, Jacques (2003). "Report on CLEF 2002 Experiments: Combining Multiple Sources of Evidence". In: *Advances in Cross-Language Information Retrieval*. Ed. by Carol Peters et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 66–90. ISBN: 978-3-540-45237-9.

Schone, Patrick and Daniel Jurafsky (2000). "Knowledge-free Induction of Morphology Using Latent Semantic Analysis". In: *Proceedings of the 2Nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning - Volume 7*. ConLL '00. Lisbon, Portugal: Association for Computational Linguistics, pp. 67–72. DOI: 10.3115/1117601.1117615. URL: http://dx.doi.org/10.3115/1117601.1117615.

Shaw, Joseph A. and Edward A. Fox (1994). "Combination of Multiple Searches". In: *THE SECOND TEXT RETRIEVAL CONFERENCE (TREC-2*, pp. 243–252.

Sheldon, Daniel et al. (2011). "LambdaMerge: Merging the Results of Query Reformulations". In: *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*. WSDM '11. Hong Kong, China: ACM, pp. 795–804. ISBN: 978-1-4503-0493-1. DOI: 10.1145/1935826.1935930. URL: http://doi.acm.org/10.1145/1935826.1935930.

Shokouhi, Milad and Luo Si (Jan. 2011). "Federated Search". In: *Found. Trends Inf. Retr.* 5.1, pp. 1–102. ISSN: 1554-0669. DOI: 10.1561/1500000010. URL: http://dx.doi.org/10.1561/1500000010.

Si, Luo and Jamie Callan (2005). "Modeling Search Engine Effectiveness for Federated Search". In: *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '05. Salvador, Brazil: ACM, pp. 83–90. ISBN: 1-59593-034-5. DOI: 10.1145/1076034.1076051. URL: http://doi.acm.org/10.1145/1076034.1076051.

Si, Luo et al. (Feb. 2008). "An Effective and Efficient Results Merging Strategy for Multilingual Information Retrieval in Federated Search Environments". In: *Inf. Retr.* 11.1, pp. 1–24. ISSN: 1386-4564. DOI: 10.1007/s10791-007-9036-6. URL: http://dx.doi.org/10.1007/s10791-007-9036-6.

Smucker, Mark D. and Charles L.A. Clarke (2012). "Time-Based Calibration of Effectiveness Measures". In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '12. Portland, Oregon, USA: Association for Computing Machinery, 95–104. ISBN: 9781450314725. DOI: 10.1145/2348283.2348300. URL: https://doi.org/10.1145/2348283.2348300.

Snajder, Jan and Bojana Dalbelo Basic (2009). "String Distance-Based Stemming of the Highly Inflected Croatian Language". In: *RANLP*. RANLP 2009 Organising Committee / ACL, pp. 411–415.

Steichen, Ben and Luanne Freund (2015). "Supporting the Modern Polyglot: A Comparison of Multilingual Search Interfaces". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, pp. 3483–3492. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702541. URL: http://doi.acm.org/10.1145/2702123.2702541.

Steichen, Ben et al. (2014). "Towards Personalized Multilingual Information Access - Exploring the Browsing and Search Behavior of Multilingual Users". In: *User Modeling, Adaptation, and Personalization*. Ed. by Vania Dimitrova et al. Cham: Springer International Publishing, pp. 435–446. ISBN: 978-3-319-08786-3.

Steidinger, Alexander (2000). "Comparison of different Collection Fusion Models in Distributed Information Retrieval". In: *DELOS*.

Stenger, Irina et al. (2017). "Modeling the impact of orthographic coding on Czech–Polish and Bulgarian–Russian reading intercomprehension". In: *Nordic Journal of Linguistics* 40.2, 175–199. DOI: 10.1017/S0332586517000130.

Strobl, Carolin et al. (2007). "Bias in random forest variable importance measures: Illustrations, sources and a solution". In: *BMC Bioinformatics* 8.1. DOI: 10.1186/1471-2105-8-25. URL: https://doi.org/10.1186/1471-2105-8-25.

Strobl, Carolin et al. (2008). "Conditional Variable Importance for Random Forests". In: *BMC Bioinformatics* 9.307. URL: http://www.biomedcentral.com/1471-2105/9/307.

Svore, Krysta M., Maksims N. Volkovs, and Christopher J.C. Burges (2011). "Learning to Rank with Multiple Objective Functions". In: *Proceedings of the 20th International Conference on World Wide Web*. WWW '11. Hyderabad, India: ACM, pp. 367–376. ISBN: 978-1-4503-0632-4. DOI: 10.1145/1963405.1963459. URL: http://doi.acm.org/10.1145/1963405.1963459.

Tax, Niek, Sander Bockting, and Djoerd Hiemstra (Nov. 2015). "A Cross-benchmark Comparison of 87 Learning to Rank Methods". In: *Inf. Process. Manage.* 51.6, pp. 757–772. ISSN:

0306-4573. DOI: 10.1016/j.ipm.2015.07.002. URL: https://doi.org/10.1016/j.ipm.2015.07.002.

Taylor, John R and Anthony P. Grant (Mar. 2014). *Lexical Borrowing*. URL: https://www.oxfordhandbooks.com/view/10.1093/oxfordhb/9780199641604.001.0001/oxfordhb-9780199641604-e-029.

Taylor, Michael et al. (2008). "SoftRank: Optimizing Non-smooth Rank Metrics". In: *Proceedings of the 2008 International Conference on Web Search and Data Mining*. WSDM '08. Palo Alto, California, USA: ACM, pp. 77–86. ISBN: 978-1-59593-927-2. DOI: 10.1145/1341531.1341544. URL: http://doi.acm.org/10.1145/1341531.1341544.

Tsai, Ming-Feng, Yu-Ting Wang, and Hsin-Hsi Chen (2008). "A Study of Learning a Merge Model for Multilingual Information Retrieval". In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '08. Singapore, Singapore: ACM, pp. 195–202. ISBN: 978-1-60558-164-4. DOI: 10.1145/1390334.1390370. URL: http://doi.acm.org/10.1145/1390334.1390370.

Tsai, Ming-Feng et al. (2007a). "FRank: a ranking method with fidelity loss". In: *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007*, pp. 383–390. DOI: 10.1145/1277741.1277808. URL: https://doi.org/10.1145/1277741.1277808.

Tsai, Ming-Feng et al. (2007b). "FRank: A Ranking Method with Fidelity Loss". In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '07. Amsterdam, The Netherlands: ACM, pp. 383–390. ISBN: 978-1-59593-597-7. DOI: 10.1145/1277741.1277808. URL: http://doi.acm.org/10.1145/1277741.1277808.

Üstün, Ahmet and Burcu Can (2016). "Unsupervised Morphological Segmentation Using Neural Word Embeddings". In: *Statistical Language and Speech Processing: 4th International Conference, SLSP 2016, Pilsen, Czech Republic, October 11-12, 2016, Proceedings*. Ed. by Pavel Král and Carlos Martín-Vide. Cham: Springer International Publishing, pp. 43–53. ISBN: 978-3-319-45925-7. DOI: 10.1007/978-3-319-45925-7_4.

Vogt, Christopher C. (1999). "Adaptive Combination of Evidence for Information Retrieval". PhD Thesis. PhD thesis.

Vogt, Christopher C. and Garrison W. Cottrell (1998). "Predicting the Performance of Linearly Combined IR Systems". In: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '98. Melbourne, Australia: ACM, pp. 190–196. ISBN: 1-58113-015-5. DOI: 10.1145/290941.290991. URL: http://doi.acm.org/10.1145/290941.290991.

Voorhees, Ellen M., Narendra K. Gupta, and Ben Johnson-Laird (1995). "Learning Collection Fusion Strategies". In: *Proceedings of the 18th Annual International ACM SIGIR Conference*

on Research and Development in Information Retrieval. SIGIR '95. Seattle, Washington, USA: ACM, pp. 172–179. ISBN: 0-89791-714-6. DOI: 10.1145/215206.215357. URL: http://doi.acm.org/10.1145/215206.215357.

Voorhees, Ellen M., Narendra K. Gupta, and Ben Johnson-laird (1995). "The Collection Fusion Problem". In: *In Proceedings of the Third Text Retrieval Conference (TREC-3*, pp. 95–104.

Wagner, Robert A. and Michael J. Fischer (Jan. 1974). "The String-to-String Correction Problem". In: *J. ACM* 21.1, pp. 168–173. ISSN: 0004-5411. DOI: 10.1145/321796.321811. URL: http://doi.acm.org/10.1145/321796.321811.

Wang, Lidan, Jimmy Lin, and Donald Metzler (2010). "Learning to Efficiently Rank". In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '10. Geneva, Switzerland: ACM, pp. 138–145. ISBN: 978-1-4503-0153-4. DOI: 10.1145/1835449.1835475. URL: http://doi.acm.org/10.1145/1835449.1835475.

Welmers, William E. (1973). *African Language Structures*. Berkeley / Los Angeles: University of California Press.

Wu, Shengli (2012). *Data Fusion in Information Retrieval*. Springer Publishing Company, Incorporated. ISBN: 3642288650, 9783642288654.

Xia, Fen et al. (2008). "Listwise Approach to Learning to Rank: Theory and Algorithm". In: *Proceedings of the 25th International Conference on Machine Learning*. ICML '08. Helsinki, Finland: ACM, pp. 1192–1199. ISBN: 978-1-60558-205-4. DOI: 10.1145/1390156.1390306. URL: http://doi.acm.org/10.1145/1390156.1390306.

Xu, Jun and Hang Li (2007). "AdaRank: A Boosting Algorithm for Information Retrieval". In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '07. Amsterdam, The Netherlands: ACM, pp. 391–398. ISBN: 978-1-59593-597-7. DOI: 10.1145/1277741.1277809. URL: http://doi.acm.org/10.1145/1277741.1277809.

Xu, Luyan, Xuan Zhou, and Ujwal Gadiraju (2019). "Revealing the Role of User Moods in Struggling Search Tasks". In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR'19. Paris, France: Association for Computing Machinery, 1249–1252. ISBN: 9781450361729. DOI: 10.1145/3331184.3331353. URL: https://doi.org/10.1145/3331184.3331353.

Xu, Yunjie (Calvin) and Zhiwei Chen (May 2006). "Relevance Judgment: What Do Information Users Consider Beyond Topicality?" In: *J. Am. Soc. Inf. Sci. Technol.* 57.7, pp. 961–973. ISSN: 1532-2882. DOI: 10.1002/asi.v57:7. URL: http://dx.doi.org/10.1002/asi.v57:7.

Xu, Zeshui (2005). "An overview of methods for determining OWA weights". In: *International Journal of Intelligent Systems* 20.8, pp. 843–865. ISSN: 1098-111X. DOI: 10.1002/int.20097. URL: http://dx.doi.org/10.1002/int.20097.

Yager, Ronald R. (Jan. 1988). "On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decisionmaking". In: *IEEE Trans. Syst. Man Cybern.* 18.1, pp. 183–190. ISSN: 0018-9472. DOI: 10.1109/21.87068. URL: http://dx.doi.org/10.1109/21.87068.

Yager, Ronald R. and Alexander Rybalov (1998). "On the fusion of documents from multiple collection information retrieval systems". In: *Journal of the American Society for Information Science* 49.13, pp. 1177–1184. URL: https://ideas.repec.org/a/bla/jamest/v49y1998i13p1177-1184.html.

Yarowsky, David and Richard Wicentowski (2000). "Minimally Supervised Morphological Analysis by Multimodal Alignment". In: *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. ACL '00. Hong Kong: Association for Computational Linguistics, pp. 207–216. DOI: 10.3115/1075218.1075245. URL: https://doi.org/10.3115/1075218.1075245.

Yee, Ka-Ping et al. (2003). "Faceted Metadata for Image Search and Browsing". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '03. Ft. Lauderdale, Florida, USA: Association for Computing Machinery, 401–408. ISBN: 1581136307. DOI: 10.1145/642611.642681. URL: https://doi.org/10.1145/642611.642681.

Yue, Yisong et al. (2007). "A Support Vector Method for Optimizing Average Precision". In: *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '07. Amsterdam, The Netherlands: ACM, pp. 271–278. ISBN: 978-1-59593-597-7. DOI: 10.1145/1277741.1277790. URL: http://doi.acm.org/10.1145/1277741.1277790.

Zhai, Chengxiang and John Lafferty (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*. SIGIR '01. New Orleans, Louisiana, USA: ACM, pp. 334–342. ISBN: 1-58113-331-6. DOI: 10.1145/383952.384019. URL: http://doi.acm.org/10.1145/383952.384019.

Zheng, Zhaohui et al. (2007a). "A General Boosting Method and Its Application to Learning Ranking Functions for Web Search". In: *Proceedings of the 20th International Conference on Neural Information Processing Systems*. NIPS'07. Vancouver, British Columbia, Canada: Curran Associates Inc., pp. 1697–1704. ISBN: 978-1-60560-352-0. URL: http://dl.acm.org/citation.cfm?id=2981562.2981775.

Zheng, Zhaohui et al. (2007b). "A regression framework for learning ranking functions using relative relevance judgments". In: *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam,*

The Netherlands, July 23-27, 2007, pp. 287–294. DOI: 10.1145/1277741.1277792. URL: https://doi.org/10.1145/1277741.1277792.

Zuccon, Guido (2016). "Understandability Biased Evaluation for Information Retrieval". In: *Advances in Information Retrieval - 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20-23, 2016. Proceedings*, pp. 280–292. DOI: 10.1007/978-3-319-30671-1_21. URL: https://doi.org/10.1007/978-3-319-30671-1_21.

Zuccon, Guido and Bevan Koopman (2014). "Integrating Understandability in the Evaluation of Consumer Health Search Engines". In: *Proceedings of the Medical Information Retrieval Workshop at SIGIR co-located with the 37th annual international ACM SIGIR conference (ACM SIGIR 2014), Gold Coast, Australia, July 11, 2014.* Pp. 32–35. URL: http://ceur-ws.org/Vol-1276/MedIR-SIGIR2014-08.pdf.