ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

CONCEPT-BASED AUTOMATIC AMHARIC DOCUMENT CATEGORIZATION

By: **Meron Sahlemariam**

A THESIS SUBMITTED TO
THE SCHOOL OF GRADUATE STUDIES OF THE ADDIS ABABA UNIVERSITY IN PARTIAL FULFILLMENT FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER SCIENCE

January, 2009

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUSTE STUDIES
FACULITY OF INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

CONCEPT-BASED AUTOMATIC AMHARIC DOCUMENT CATEGORIZATION

By: Meron Sahlemariam

ADVISORS:

Daniel Yacob - External Advisor

Mulugeta Libsie (PhD) - Internal Advisor

APPROVED BY

EXAMINING BOARD:

1. Dr. Mulugeta Libsie, Advisor _____
2. _____ _____
3. _____ _____

Acknowledgements

First, I gratefully acknowledge the help, guidance and support of God in my whole life, GLORY TO GOD. Next, I would like to thank everybody that helped and supported me in a way or another. I am deeply thankful to my advisors Mulugeta Libsie(PhD) and Daniel Yacob for their constructive suggestion and encouragement during this research work.

Special thanks to Jihan Ahmed, Hirut Dessalegn, Tessema Mindaye, Workshet Lamenew, Mesfen Getachew, Tewodros Atlaw, and Zelalem sintayehu for their valuable support throughout the course of the research.

I wish to express my sincere gratitude to Dr. Ahmed Hussein for introducing me to the road of knowledge. Ahmed is not only my computer science teacher but also he is my exemplar. Dearest Ahmed, I genuinely appreciate you with lots of love and respect. The enlightenment and seed I found from is my vigor to move forward. Throughout my way of life, I will never forget the offering that I found from HiLCoE as it was my defining moment. I wish to thank all staff members of HiLCoE and I am proud of HiLCoEዬ.

My special thanks goes to my best friends Lubna Ahmed and Yonas Hailu for helping me through all those moments of this research work, good and bad. Luዬ you are my blessing from heaven, I don't have words to thank you, and you are so special. Yoniዬ you are not only a best friend, I better declare you brother, I will never forget those days. Yonas, I have learned lots of things that serve me as major life principles. Both of you, thank you through all my ability, this thesis would not be possible without your huge support, inspiration and encouragements. I am glad to have such wonderful friends.

Credit should also be given to Kirubel Berhane, Hafra Abubeker, Nesredien Suleiman, and Endashaw Kebede for your concern during my stay in the university. I thank the Ethiopian News Agency and its employees for their willingness to provide source data for this research work. Specially, Ato Dereje G/Meskel, ICT head of Ministry of information, Ato Muluneh Gebre and Ato Nesru Jemal.

And finally, I thank all my family my mother, brothers, sisters and kids, I am pleased to be a member of such a wonderful family. Thank you all for your unconditional love and prayer. Your gift to me is more than I can ever realize, I always respect and love you.

Table of Contents

List of Tables

List of Figures

List of Appendices

List of Acronyms

AI	Artificial Intelligence
ABox	Assertion Box
API	Application Programming Interface
ANC	Amharic News Classifier
ER	Entity Relationship
ENA	Ethiopian News Agency
TF	Term Frequency
FLogic	Frame Logic
IDF	Inverted Term Frequency
ICT	Information Communication Technology
IR	Information Retrieval
InfoMap	Information Map
KIF	Knowledge Interchange Format
KNN	K-Nearest Neighbors
NB	Naïve Bayes
NLP	Natural Language Processing
OCML	Operational Conceptual Modelling Language
OIL	Ontology Inference Layer
OWL	Web Ontology Language
OilED	Ontology Inference Layer Editor
OntoEdit	Ontology Editor
RDFS	Resource Descriptive Framework Schema
SMI	Stanford Medical Informatics

SQL	Standard Query Language
SWRL	Semantic Web Rule Language
SPARQL	SPARQL Protocol and RDF Query Language
SVM	Support Vector Machine
TBox	Terminology Box
TBC	TopBraid Composer
TF/IDF	Term Frequency/ Inverted Term Frequency
UML	Unified Modeling Language
W3C	World Wide Web Consortium
WebODE	Web Ontology Design Environment
WWW	World Wide Web
XML	EXtensible Markup Language

Abstract

Along with the continuously growing volume of information availability, there is a growing interest towards better solutions for finding, filtering and organizing these resources. Automatic text categorization can play an important role in a wide variety of more flexible, dynamic, and personalized information management tasks.

The process of automatic text categorization involves calculating similarities between documents and categories using the information extracted from the document. In recent years, ontology-based document categorization method is introduced to solve the problem of document classifier. Previous works on keyword-based document categorization miss some important issues of considering semantic relationships between words. In order to resolve the existing problems, this study proposes a framework that automatically categorizes Amharic documents into predefined categories using knowledge represented in the `News` ontology. At the heart of the classification system is the knowledge base that enables the representation of different domain concepts.

During the classification process, all the documents pass through pre-processing stages. Then index terms are extracted from a given document which is mapped onto their corresponding concepts in the ontology. Finally, the selected document is classified into a predefined category, based on the weighted concept.

With the help of `News` domain ontologies, this study categorizes a given Amharic document into a specific predefined category. The study shows that the use of concepts for Amharic document categorizer results in 92.9% accuracy which is a promising outcome.

Keywords: Ontology, Keyword-based, Concept-based text categorization, Knowledge representation.

Chapter One

INTRODUCTION

1.1. Overview

Humans seek information as a consequence of the need to achieve some goal. According to Wilson [1], the need for information is not fundamental unlike shelter or the need for sustenance rather it is a secondary ordinary need which arose out of the desire to satisfy the primary needs. Information by itself is related with human behavior in relation to its source, including face-to-face communication with others and passive information sources without any intervention like watching TV, reading books, newspapers, and so on [2]. Nowadays, such information resources are available in electronic forms through Web technology.

Accessing electronic resources becomes easier through the Internet. There are now several billion documents on the Web which are used by more than 300 million users globally [2]. Furthermore, the number of active users and information resources on the Web is increasing at an accelerating pace [3]. The continued rapid growth in information volume causes difficulty to find, organize, access and maintain information.

Along with the continuously increasing volume of information availability on the Web, there is a growing interest in getting better ways of accessing these resources. As the amount of information has dramatically grown, finding relevant information among the millions of information resources on the Web is becoming more difficult. Users are currently restricted to browse or follow hyperlinks from one web page to the next and syntactic keyword searches for finding significant information [3].

The need for information is vital, the storage and retrieval within size and time framework is becoming a challenge. Satisfying information need and tackling these challenges is the major issue in information retrieval (IR). IR is a sub-discipline of Information Science that is concerned with developing theories and methods of storage, organization, representation and access of information from a collection [4, 5]. There is a mechanism of storing and accessing information according to the requirement of the user. The problem of information storage and retrieval got attention because of the difficulties to get the relevant information within the required time [6].

Therefore, the organization and access of information items should provide the user with easy access to the information which might be useful or relevant to the user. There are different ways of organizing information. One of the most successful paradigms to organize such information is classifying documents into different categories which are meaningful to users. Categories signify organization of items into groups according to their similarities or shared characteristics such as Sport, Health, Politics, Economy, and so on.

There are various reasons for using document categorization. Document categorization reduces searching time, thereby facilitating the searching process. Moreover, it facilitates access, when documents are classified based on their concept similarity; we can get hint about what the document actually contains without going through it.

Document classification can be done manually or automatically. Manual text categorization is carried out by human experts. It requires a certain level of vocabulary recognition and knowledge processing. There are some problems observed with manual classification. It requires intensive human labor and affects classification results because of inconsistency due to variation in perception, comprehension, and judgment, and for the current Web based knowledge management

it is almost impossible. In contrast, automatic classification is a process of classifying documents into a number of classes using machine learning methods [7].

There are two different approaches towards the meaning and process of automatic classification. The first approach considers automatic classification as a technique of classifying documents without having any prior knowledge of the categories where the documents would be classified. The classifying process is expected to create classes based on the similarity that exist among the documents. This automatic grouping of documents is referred to as cluster analysis or unsupervised clustering [8]. The second approach considers automatic classification as a process of classifying documents into predefined classes called supervised clustering [9].

This thesis follows the second approach of document categorization as a process of automatically putting similar documents together based on their contents to a number of predefined categories. In other words, it is a procedure of collecting and partitioning a set of documents into predefined classes so that the documents in the same class are more similar than documents in another class.

Document categorization or classification can be single-label, multi-label and hierarchical [8]. In the case of single-label classification only one class is assigned for the document, but in a multi-label classification each document can be assigned an arbitrary number of multiple labels of multiple possible classes. This is because a document may contain multiple concepts. In the hierarchical categorization of documents, the main category may have a number of sub-categories [10]. For example, in a document for sport the main category called sport may have sub-categories under it like athletics, football, basketball, ground tennis, etc.

Basically, automatic document categorization can be manipulated based on keywords or concepts to categorize a given document into a specific category. Keyword-based text categorization only uses keywords which are extracted from the text to identify the category of a given document.

Mainly it works by the comparison of keywords, which means a document that is going to be categorized should contain a specific keyword that matches the represented document to be categorized into the predefined category.

Instead of using keywords, documents can also be classified by taking into consideration the concept that the document represents. Concept is a "semantics" or meaning of terms. Terms are words that describe concepts or act as synonyms for concepts. For example, the terms Football and Soccer have the same meaning, that means the concept behind both terms are the same or both terms talk about the same thing.

Hence, concept-based text categorization allows classification of documents based on meaning rather than keywords. This method extracts concepts from the document and uses those concepts to categorize the document [11]. In order to use concepts to categorize documents, the concepts should be represented in the knowledge base. To do so, representing such concepts in the knowledge base is provided using ontologies. An ontology is a systematic formalization of concepts, definitions, relationships, and rules that captures the semantic content of a domain in a machine-readable format [12].

1.2. Statement of the Problem

The research and development of Amharic text categorization systems is in its formative state. To our best knowledge only few researches were conducted in this area. From the survey of literature, it is apparent that all of the previous efforts were focused on developing keyword-based Amharic document categorizer [13, 14, 15]. The previous studies addressed a number of issues in the area of Amharic document categorization: text categorizer for Amharic documents, a tool to correct word spelling variations, enhancement to the suffix and prefix removal tool, a tool to correct word

variations due to gender marker suffixes, a tool to correct word variations due to number marker suffixes, and a tool to merge compound words written as separate words, are the major contributions to be mentioned.

However, there are still a number of issues that are not addressed yet. The foremost issue is that all of the previous studies depend only on keywords to categorize documents to a certain category. The major problem with this approach is that it ignores semantic relationship between the document's content and the designated category. A document consists of one or more ideas. It is this central idea of the document that makes it interesting to the user. Organizing the document collection using the central idea or the concept of each document will make the process of classification accurate. As opposed to keyword-based technique, this approach guarantees robust classifier as it is not influenced by word variations.

Keyword-based classification model is not so successful, mainly due to the fact that classification has not been based on the core meaning of the document rather it depends on a set of selected keywords. Using only keywords cannot guarantee satisfactory results since authors may use different keywords. In addition to failure to consider the document's semantics, none of the previous works have had addressed the issue of documents that belong to two or more categories.

Therefore, this thesis work is an attempt to explore concept-based method of automatically classifying Amharic documents into predefined categories. The suggested strategy grants better result for document classification by achieving robustness with respect to linguistic variations such as vocabulary and word choice. Moreover, multi-label document categorization and hierarchical document categorization which were not included in the previous studies are also employed.

1.3. Justification of the study

Ethiopia is a country which uses its own unique alphabet written in the Ethiopic script. Amharic uses the Ethiopic script and is the official language of the Federal Government of Ethiopia. It is spoken by more than 27 million people as the first language and 7-15 million people speak Amharic as a second language [16]. Hence, there are many users for the language. In addition to that the number of electronic documents that are published in the Amharic language is exponentially increasing and hence retrieval of these documents is becoming a problem [7].

Having these facts and in order to organize resources, the need for investigating Amharic information access has been acknowledged. A lot has to be done on information retrieval to access documents written in the Amharic language easily and efficiently.

To attain accessing Amharic documents easily through document categorization, some works have been carried out on document categorization by different researchers. However, concepts were not totally considered to categorize documents in the previous studies. Representing knowledge using concepts and accessing such knowledge is the promising way of research in different areas. This work contributes to the development of ontologies and, implicitly, to the concept-based automatic Amharic document categorization. It is believed that using semantic for Amharic document categorization makes the accessibility of the information more successful.

1.4. Objective of the Study

The general and specific objectives of this study are underlined below:

1.4.1. General Objective

The general objective of the study is to come up with an automatic Amharic document categorizer that uses concepts, which is specifically designed for Amharic documents with the capability of classifying documents based on their concept into predefined categories.

1.4.2. Specific Objective

As stated earlier, the main concern of this study is developing an Amharic document categorizer. This incorporates development of the required knowledge base. To reach to the general objective of the study, the following specific objectives are identified.

- Identifying the process of automatic Amharic document categorizer using concepts.
- Identifying different activities involved in order to construct the ontology.
- Designing a generic model for concept-based automatic Amharic text categorizer.
- Developing the full-fledged system for automatic Amharic document categorizer.
- Conducting experiments to evaluate the usability of the proposed system.

1.5. Scope and Limitation of the Study

1.5.1. Scope

The scope of the study is to propose a model and develop an automatic Amharic document categorizer using concepts. In this research work, an ontology is developed, which is limited to have some knowledge that will aid in determining whether a given Amharic document should be

classified to a specific category or not. In short, the scope encompasses formulating domain concepts, building relations between concepts and representing restrictions.

1.5.2. Limitation

- The study only used Amharic News data sets for the experimental purpose. However, the nature of the document is thought to be the same with most of others Amharic documents.
- The study considers only Amharic textual documents that contain sequence of Amharic alphabets without any figure, table, images or pictorial representations.
- This thesis represents only the knowledge in the area of `Accidents`, `Economy`, `Science and Technology`, `Environmental Preservation and Weather Condition`, and `Sport` categories for demonstration purpose.

1.6. Methodology

In order to achieve the expected result of the study, different methodologies were employed as described below:

1.6.1. Methodologies

Literature Review

Related literatures from different sources such as relevant published documents, materials and journal articles were reviewed to get an understanding of the various techniques of automatic document categorization. Available works on the development of concept-based automatic document categorization for other languages have been reviewed in depth.

Design the Framework of Concept-based Automatic Amharic Document Categorizer

Automatic document classification has various steps and methods that can be used at each stage in order to develop the classifier. Frameworks that categorize Amharic documents into predefined categories using concepts are proposed.

Prototyping

In order to implement the proposed model of Amharic document categorizer, the study developed the "`News`" ontology that contains the appropriate knowledge base with several concepts and comprises categories and sub-categories of `News` items. In addition to the knowledge base, the Amharic document categorizer is also developed.

1.6.2. Tools

In order to accomplish the study, different tools are employed. Java programming language and Jena framework are used to develop the system and TopBraid Composer (TBC) is selected as a working environment to develop the ontology. TBC allows users to write explicit, formal conceptualizations of domain models using the World Wide Web Consortium (W3C) standard languages such as Resource Descriptive Framework Schema (RDFS), Ontology Web Language (OWL), Semantic Web Rule Language (SWRL) and RDF Query Language (SPARQL) [12, 17]. In this study, an ontology language called OWL is used to formulate the knowledge.

1.6.3. Data Source

The source data for Amharic News items was the Ethiopian News Agency (ENA). For the experimental purpose, the data was manually classified into categories and sub-categories.

1.6.4. Testing and Evaluation

The outcome of the study was evaluated with the appropriate evaluation techniques to check whether the categorizer performs correctly or not. The developed prototype was tested for correctness using test data set. The result, which is automatically classified documents, was checked against the manual classification.

1.7. Application of the Study

Text classification can be applied in different ways of managing information and its schemes vary in scope and methodology but it can be divided into universal, national, general, subject specific and home-grown schemes [14].

Concept-based automatic Amharic document categorizer plays an important role in a wide variety of information management tasks and the findings of this work can be used:

- In search engines to improve search result.
- For filtering and categorizing news items for any interest group, especially for news agencies, for instance, the Ethiopian News Agency (ENA) and Walta Information Center (WIC).
- To organize and improve browsing Amharic web documents.
- For any organization which has a large collection of Amharic documents to automatically categorize documents for better management, and
- Moreover, the result of the study will play a role in academics for future study in the area of concept-based Amharic document categorization.

1.8. Thesis Organization

The remainder of this thesis report is organized as follows:

Chapter 2 introduces the background knowledge of text categorization and the steps in automatic classification. A classification of ontologies according to their level of details is presented and the use of ontologies for information integration is reviewed. The chapter also presents components of the ontology, ontology languages, tools and different methodologies for building ontologies.

Chapter 3 presents a review of related works on document categorization. This chapter discusses researches that have been conducted on text categorization for different languages that used keyword-based and concept-based automatic classification techniques.

Chapter 4 presents the proposed framework of automatic Amharic document categorizer by focusing on the aspects of pre-processing and knowledge representation. This chapter also presents the implementation issues and how the main components can be implemented. The formulation of ontology and steps to extract concepts from the ontology are discussed.

Chapter 5 presents the experimental result of the proposed design for concept-based automatic Amharic document classification.

Finally, conclusions and future work are presented in Chapter 6.

Chapter Two

LITERATURE REVIEW

2.1. Introduction

This chapter gives a brief description of the thesis background by focusing mainly on the text categorization process of Amharic documents. Text categorization is one of the applications of IR which is a process of classifying a given document to specific and predefined categories. This chapter presents the steps in document categorization including pre-processing and classification.

Pre-processing is done to prepare the document for the classifier by applying lexical analysis, normalization, stop-word removal and stemming. Classification involves assignment of a document to a predefined category using keywords or concepts. Methods of representing concepts are also addressed in this chapter through describing ontologies for the knowledge representation, different types of ontologies, languages and tools related to the ontology were presented in the subsequent sections.

2.2. Text Categorization

Typically, any classification process involves collection of similar documents into categories. Each category is a labeled group and usually it is a collection of documents related to the same topic. This implies the procedure of classifying documents into a predefined category requires checking similarity of the documents for a given category.

Text categorization can be done either manually or automatically. Manual text categorization is carried out by human experts and it requires a certain level of vocabulary recognition and knowledge processing. Manual classification requires intensive human labor that is too expensive

and is not feasible for large document collection. But, this problem can be alleviated by means of automatic text categorization [18].

Automatic text categorization is the process of automatically classifying a set of documents into predefined categories. The automatic categorization process is a combination of information retrieval (IR) technology and knowledge representation technology. In general terms, it is a process of classifying a given document into one or more predefined classes [19].

Automatic classification organizes documents automatically into predefined categories based on the similarity measure. The attribute to measure the similarity can be text (it can be a keyword or sequence of words) or concept of the document.

Based on the above similarity measure, automatic text categorization assigns a given document into categories. This process attempts to replace and save the human effort required in performing manual categorization. It consists of assigning and labeling documents using a set of predefined categories based on document contents.

There are two main steps to categorize documents automatically: pre-processing and the actual classification. The pre-processing incorporates the following activities: lexical analysis, normalization, stop-word removal, stemming, and index term selection. In the classification phase, categorization can be done using keywords or concepts. There are a number of standard machine learning techniques which have been applied on keyword-based text categorization. The commonly known are Naive Bayes classifiers, Support Vector Machines, Linear Least Squares Models, Neural Networks, and K-Nearest Neighbor classifiers. For concept-based text classification, ontology is used as a way of representing knowledge in a specific domain.

Automatic text categorization plays an important role in a wide variety of information management tasks such as:

- Real-time assignment of email or files into folder hierarchies,
- Topic identification to support topic-specific processing operations,
- Structured search and/or browsing or finding documents,
- Filtering a stream of news for a particular interest group,
- E-business setting that exhibit a large number of target categories with relatively few training cases, and
- Online tendering system.

Advantages of using classification schemes over unclassified ones includes improved subject browsing facilities, potential multi-lingual access and improved interoperability with other services [20].

2.3. Steps in Automatic Document Classification

With the aim of categorizing a given document into predefined categories, automatic text classification involves pre-processing and classification steps.

2.3.1. Pre-processing

In order to perform automatic text classification, the document must first be converted to an acceptable representation that can be used by the classifier. The pre-processing activity involves lexical analysis, normalization, stop-word removal, stemming, and index term selection as shown in Figure 2.1. An input for this process is a text document, but not every word in the text is meaningful for categorization or retrieval. For this reason documents must be processed and represented to a concise and an identifiable format or structure. This will then bring major benefit of data size reduction which increases performance in terms of memory size and processing time. The following subsection discusses each of them in detail [4].

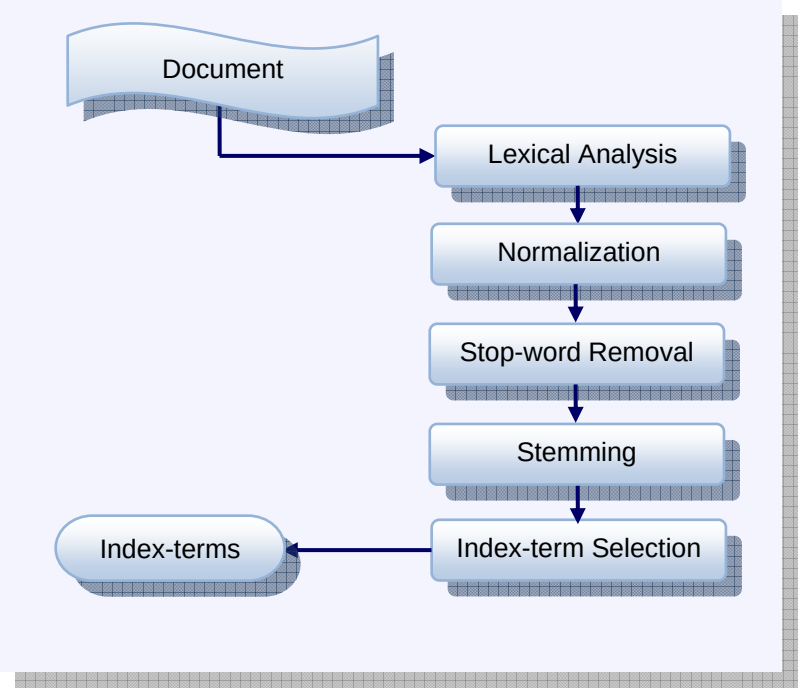

Figure 2.1:Pre-processing steps in automatic text categorization

Lexical Analysis

The first task in lexical analysis is tokenization, which is the process of converting the stream of characters in a document into tokens or list of terms, where a term is defined as a string of letters, digits or other special characters, separated by punctuation marks and spaces.

In addition to tokenization, lexical analysis includes data cleaning, which is the process of removing characters that have no meaning in the dictionary. In a given document there could be words, which are composed of letters and digits. It is the task of cleaning for removing words of such type. The cleaning avoids errors which are against the syntactical rules of the language. In order for a document processing activity to be error free, each character must be valid and the terms should be constructed following appropriate syntactical rules of the language. As the first step of enforcing validity of characters and terms dictionaries of the language should be used.

Normalization

After lexical analysis, it is normalization of homophones that follows. Amharic writing system has homophone characters, for example, it is common that the character **ስ** and **ሥ** are used interchangeably as **ስራ** and **ሥራ** to mean "work". Such type of inconsistency in writing words will be handled by replacing characters of the same sound by a common symbol. These characters cause unnecessary increase in the number of document representative words that causes large data size processing. Sample of such normalized Amharic words are shown in Table 2.1.

Table 2.1: Sample Normalized Amharic words

The Word in English	Spelling Variants of the Word	The Normalized Word
Work	ስራ፣ሥራ	**ሥራ**
Sun	ጸሐይ፣ ጸሀይ፣ ጸሃይ፣ ጸኃይ፣ ጸሓይ፣ ጸኀይ፣ ፀሀይ፣ ፀሐይ፣ ፀሃይ፣ ፀኃይ፣ ፀሓይ፣ ፀኀይ፣	**ፀሐይ**
Power	ሃይል፣ ኃይል፣ ሀይል፣ ሐይል፣ ሓይል፣ ኀይል፣	**ሀይል**
World	ዐለም፣ ዓለም፣ አለም፣ ኣለም	**አለም**

Stop-word removal

The next process after lexical analysis and normalization is stop-word removal. Stop-words are words that occur most frequently in documents, but are not relevant or have no impact to discriminate among documents. The common words in English such as ***of***, ***a***, and ***the***, are stop-words and such words are not used to discriminate the documents. Such frequently used words generally "glue" sentences together but they usually do not carry meanings. The most frequently used stop-words in Amharic documents are **ሆነ**, **ሁሉ**, **ነው**, and **ነበር**. In order to remove stop-words, a list of stop-words should be identified and listed. A stop-word list is a list of such non-content bearing words that have to be removed during pre-processing of the document. This process is normally done automatically by comparing words in the input text with words in a 'stop-word list'.

From the point of IR, removing such words from the documents is the best way to avoid retrieving unnecessary documents that are not relevant to satisfy the user's need. From the point of categorization, applying stop-word removal also reduces the complexity of the document representation and the number of tokens to be processed.

Stemming

Words that appear in a document often have many morphological variations. In most cases morphological variations have similar interpretation and can be considered as equivalent for the purpose of IR applications. The process of stemming is an attempt to reduce a word to its stem or root form. For example, stemming will bring the different forms of the word **ቤት** "House" (**ቤቱ**, **ቤቶች**, **ቤታችን**, **ቤቶቻችን**, **ቤታቸው**, **ቤቶቻቸው**) into their stem word **ቤት**. Thus, terms of a document are represented by stem words rather than by the original words. This also reduces the number of different terms needed for representing a document and also saves storage space and processing time.

There are a number of stemming algorithms, such as table lookup approach, successor variety, n-gram stemmers and affix removal. Nega and Willet, in [21], have developed a stemming algorithm for the Amharic language. The study presented a stemmer for processing documents and query words which facilitate searching Amharic databases.

Index term selection

So far, tokenization, normalization, stop-word removal, and stemming have been done on the documents. After these processes, to classify documents automatically, the documents should further be processed in order to be represented using their representative words.

A document is represented using a collection of index terms or words. An index term is simply any term or word that appears in a document. Commonly, an index term is a word which has its own meaning and is capable of representing a document or reflecting the content of the text.

Such a selection follows lexical analysis, normalization, stop-word removal and stemming. Out of list of terms, selection of representative words will be made based on the criteria of selecting a document. There are different techniques of selection of index terms to represent a document. Among the most widely used techniques to select document representative terms are statistical techniques. Statistical techniques for text analysis are based on Term Frequency (TF) and Inverted Document Frequency (IDF).

TF is the number of times a term occurs in a given document. It is a frequency of words in the document to determine which words were sufficiently significant to represent the document. It is based on the principle that a term which is frequently used in a document is useful to represent the document. In short, the frequency shows usefulness of the term in the document, and it is formally stated as follows [15]:

Term frequency TF (d, t), is the number of times a term t occurs in the document d and is defined as:

$$Tf(d_i,t_k) = \sum_{j=1}^{n} f\ (T_j)$$

Where d_i is the i^{th} document, t_k is the k^{th} term of document d_i and T_j is the j^{th} term in the document.

IDF is the occurrence of a term in the collection of all input documents; if a word occurs in all documents, the relevance of the document will decrease because the probability of the word to represent the document is less. That is, terms that appear in many documents are not very useful as they do not allow discriminating between documents. A formal definition of IDF is: [4]

$$Idf\,(\mathrm{t}) = log\left(\frac{\mathrm{N}}{df(\mathrm{t})}\right)$$

Where $df(\mathrm{t})$ is the number of documents including the term t, N the number of all documents.

It is possible to use either term frequency or inverted term frequency depending on the application. However, term frequency is more appropriate for this study because the study considers a single document at a time instead of collection of documents to categorize.

For this study, all the pre-processing activities are adopted from a previous study by Tessema [13]. Each of the adopted components is described in Chapter 4 of this thesis.

2.3.2. Classification

Classification involves assignment of a document to a predefined category. This can be done either using keywords or concepts.

a. Keyword-based Text Categorization

Keyword-based text categorization is a process of categorizing a given document using keywords that describe the content of the document. In order to categorize a given document using keyword-based categorization, the document should contain specific keyword that matches the represented document. The basic idea behind keyword-based categorization is that a document that describes a certain concept is more likely to have words from that domain. For instance a document that talks about “culture” should have the word “culture”.

In this method, keyword lists are used to describe contents of the document. The keywords are extracted from the document and then these keywords are used as document representatives. A document is categorized into a certain category by using these representative words [19].

Researches have been conducted in the area of keyword-based automatic text categorization for Amharic [14, 15, 22] and other languages as well [23, 24, 25]. The major problem of such kind of text classification is that it does not take the meaning of the word into account. A keyword list is a description that does not explain anything about the semantic relationships between keywords. This makes the classification process dependent on the keywords and therefore not sufficient to satisfy the users need.

b. Concept-based Text Categorization

Concept-based text categorization is an approach to categorize documents using semantics. In other words, it is the process of classifying a document based on meaning rather than the presence of keywords in the document. The content of the document is described by a set of concepts. Thus, a concept is extracted from the text and then the document is categorized according to the extracted concept [11].

Concept-based access to information has important benefits over keyword-based access. One of these benefits is the possibility to take advantage of semantic relationships among concepts to find relevant documents. Another benefit is the elimination of irrelevant documents by identifying conceptual mismatches. It also improves browsing facilities. For browsing, a potentially better approach would be to map every word onto a concept, the proper word sense, based on the word's context in the document and an ontological knowledge-base with concept descriptions and semantic relationships among concepts [4]. Concepts are represented by words; words can represent multiple concepts and different words may represent the same or very similar concepts [26].

Different areas such as academic and industrial research institutions use concept-based information retrieval tools and techniques [27]. For example [28] and [29] are researches done on concept-

based text categorization for other languages. However, as far as our knowledge is concerned there is no work done on concept-based text categorization for the Amharic language.

This study is the first attempt to use concept-based text categorization for Amharic language and it is a new and promising way of improving the categorization ability to produce the desired result. To achieve document categorization using concepts, certain domain specific intelligence or knowledge of understanding a real world object must be represented in machine understandable manner. In the case of concept-based text categorization, the knowledge to discriminate documents from one another is mandatory. To do so, the knowledge should be represented formally. The detail on knowledge representation is given in the next section.

2.4. Knowledge Representation

Knowledge refers to “justified true belief” or general awareness of information, facts, ideas, truths and principles that is codified in formal and systemic language. It can be combined, stored, retrieved and transmitted through various means, including modern Information Communication Technology (ICT) [23]. In other words, knowledge deals with the reasoning capabilities based on the existing source of information.

Such domain knowledge should be formally described and used in some ways of knowledge representation. Knowledge representation deals with providing a way of representing information and building knowledge in a specific domain area. It is described in terms of the fundamental roles that it plays, such as [30]:

- A replacement or substitute for the thing itself
- Enable to determine consequences by reasoning about the world

- It is a fragmentary theory of intelligent reasoning expressed in terms of the representation's fundamental conception of intelligent reasoning

Knowledge representation is used across various fields of subject studies. Disciplines like Artificial Intelligence (AI) were used to formalize knowledge activities as a way of modeling human thinking and representation of knowledge in machine understandable format [31]. AI has been the way of information transfer and knowledge sharing. Such systems are developed and organized using some domain knowledge to share and reuse knowledge. The initial idea of knowledge representation is that, instead of developing a system from scratch, it is better to use an existing knowledge base [32].

Knowledge representation involves abstraction and interpretation of real world knowledge using formal theories and reasoning procedures. Therefore, knowledge has to be conceptualized and represented in machine understandable mode. It means that knowledge is formalized and conceptualized in a symbolic form which can be interpreted [31].

Conceptualization is the process of representing knowledge in machine understandable way. It consists of a set of objects, concepts, and other entities about which knowledge is being expressed and of relationships that hold among them. The process of conceptualization can be achieved using ontologies, which allow the representation of knowledge in machine understandable form [32].

Typically, the knowledge has been acquired to categorize a given document into predefined categories; it is then required to represent the knowledge in an ontology language. The representation of knowledge can be attained by using ontology and offers an opportunity to significantly improve knowledge representation.

2.5. Ontology

Due to the growth and availability of information, the need for a new approach to manage information is beyond doubt. For the last few decades, the development of the World Wide Web (WWW) provided a way of accessing information easily. The volume of information in the Web tripled between the years 2000 and 2003 [33]. This fact shows that there must be some technique to extract and organize such information.

To achieve this, the focus of modern information systems is moving from data processing to concept processing. Data processing is becoming less important and the paradigm is shifted from data processing to concept processing. Such semantic way of processing information is possible to be implemented using ontologies.

The word ontology can be seen from two perspectives. The first one is the **philosophical perspective of ontology,** which defines it as "a science of being: specifically, a branch of metaphysics relating to the nature and relations of being; a particular system according to which problems of the nature of being are investigated" [32, 34]. The second perspective is the **AI-Perspective of ontology.** According to the review presented in [32, 34], ontology from AI perspective is defined as "an explicit specification of a conceptualization".

In fact, there are different definitions of ontology. However, according to Gomez-Perez *et al*. [31] a commonly agreed definition of ontology is "An ontology is an explicit and formal specification of a conceptualization of a domain of interest''. This definition stresses two key points: the conceptualization is formal and permits reasoning by computer, and a practical ontology is designed for some particular domain of interest.

Ontologies were developed in AI systems to facilitate knowledge sharing and reuse. They have been studied by several artificial intelligence research areas, including knowledge engineering,

natural-language processing and knowledge representation. More recently, the use of ontologies has also become widespread in fields such as intelligent information integration, cooperative information systems, information retrieval, electronic commerce, and knowledge management [31, 2].

Ontology is a particular theory about the nature of being or the kinds of existence. The task of intelligent systems in computer science is to formally represent these existences in machine readable format using conceptualization. Every knowledge model is committed to some conceptualization, implicitly or explicitly. An ontology is an explicit specification of this conceptualization.

Formally, an ontology consists of terms, their definitions, and axioms relating them. As a formal description, ontologies consist of concepts known as classes, relations or properties, and instances. Formalized ontologies are instruments for capturing the meanings of concepts so that they may be used for improved, automated management of information. Ontologies may cover very general concepts or represent specific and restricted domains. The selection of concepts and their level of detail will depend on the characteristics of the domains to be covered and the operations needed.

The main advantage of ontology is knowledge representation and reusability which allows reusing and sharing application domain knowledge using common vocabulary. Ontologies are used to organize knowledge in a structured way and they are preferred ways of knowledge representation in the semantic technology. It is useful to avoid building applications right from scratch and provides common mode of communication among the agents. Moreover, they facilitate representation of machine accessible information formally and explicitly with reasoning capability [32].

This study is concerned about the ontology used in intelligent computer applications. Throughout this document an ontology refers to a systematic formalization of concepts, definitions, relationships and rules that capture the semantics content of a domain in a machine readable format.

The subsequent section presents about the basic components of the ontology including concepts, relations, an individual, an attribute and functions which allow the formalization of knowledge in machine understandable way.

2.5.1. Basic Components of an Ontology

To formalize knowledge, ontologies use basic modeling component types. The two most important kinds of components in an ontology are the classes in which individuals can be categorized, and the relations that are used to create links between classes. Each of the components is discussed below [4, 62].

> ***Classes/Concepts*** can be anything about which something is said and, therefore, could also be the description of a task, function, action, strategy, reasoning process, and the like. It is used to capture knowledge about a kind of thing and represent characteristics that may apply to many individuals. Classes are like generic nouns that are applied to distinct and named individuals such as machine, human, dog, company. An individual member of a class has the general character of the class and as an individual it may have other characteristics and relationships. Individual members of a class can be described in some detail, including their characteristics and relations with other individuals in the same or other classes.

Predicates are the most important type of relations in an ontology. Predicate terms explicitly represent relations that may link two or more items. For example, two people are related by the predicate father:

```
George H. W. Bush is the father of George W. Bush.
```

Predicates may connect classes, an individual and a class, or multiple individuals:

```
Republicans are type of politician.
George W. Bush is a Republican.
George W. Bush leads the United States.
```

They are formally defined as any subset of a product of n sets, that is: R: C_1 x C_2 x ... x C_n. There are various kinds of relationships [17, 35]. The main types of semantic relationships are [36, 37]:

- **Synonym** is a kind of semantic relation. Two words are synonyms when they have the same meaning and the substitution of one for the other never changes the truth value of a sentence in which the substitution is made.
- **Antonym** is a semantic relation indicating oppositions in meaning between terms.
- **Hypernym** is a semantic relation identified as the super ordination relation, and is generally known as the Has Kind of relation or simply HasA. Hypernymy is the reverse of hyponymy.
- **Hyponym** is a semantic relation identified as the subordination relation, and is generally known as the Is Kind of relation or simply IsA.
- **Meronym** is a kind of transitive and asymmetrical part-whole relationship (between the whole and its parts). It is also known as part of, MemberOf, SubstanceOf, ComponentOf, etc.

- **Holonym** is the reverse of meronym and generally identified as HasPart (also HasMember, HasSubstance, HasComponent, and so on).

Relations are classified according to other important characteristics such as reflexive, symmetric, or transitive. Reflexive relations are those that can relate something to itself, e.g., "equal". Symmetric relations are those that hold between two arguments in both directions, for example, "spouse" is a symmetric relation. Transitive relations are those that can transfer along related items, for example, if a transitive relation R holds between A and B and it also holds between B and C then that relation holds between A and C [32].

Individuals, also called instances, are those described using the concepts of an ontology. Typically, individuals are described as being members of some class. An individual member of a class has the general character of the class and it may have other characteristics and relationships as well [32].

Haile G/ Selassie, the famous Ethiopian runner, for example, is an instance of the class of Athlete. Individuals may also have name strings (e.g., "Haile G/Selassie"), attributes ("Male","1.65m"), and relations to other individuals "Haile G/Selassie is the husband of Alem Haile".

Attributes in general denote simple qualities that are secondary characteristics of objects (e.g. red, solid, short), in contrast to the essential properties represented by classes. In many cases, qualities represented by attributes include physical states of matter, colors, size, and the like [32].

Functions are special cases of relations in which the n^{th} element of the relationship is unique for the n-1 preceding elements. Examples of functions are Mother-of and Price-of-a-

used-car that calculates the price of a second-hand car depending on the car-model, manufacturing date and number of kilometers [32].

The other issue related to the ontology is the way of knowledge modeling technique. Ontologies can be constructed using different knowledge modeling techniques. In the 1990's, AI modeling techniques, based on frames and first order logic, were used to construct ontology. In the last few years, knowledge representation techniques based on description logic have been used. There are also ontology modeling techniques using software engineering techniques, and database techniques. Each technique is described below [32].

Modeling Ontologies Using AI Modeling Techniques: AI based technique of ontology modeling uses components to model ontologies. There are different components used in order to model ontologies using AI modeling techniques, which are classes, functions, formal axioms and instances [32].

Every concept that is going to be described by the ontology is represented using ***classes***. The concept can also represent abstract concepts such as intention, belief, feeling or specific concepts such as people, computer, table, and so on. Accordingly, classes and sub-classes are used to represent abstract or concrete concepts that maintain some relationship in order to represent the domain knowledge fully or completely. The relationship or associations within the classes are represented using ***relations***. Domain knowledge which are always true and that cannot be formally defined using other components are described with ***Formal axioms***. ***Instances*** are treated in the ontology as individuals or specific elements.

Modeling Ontologies Using Knowledge Representation Techniques: this technique uses Description Logic to model the ontology. Description logic is a way of describing structured knowledge in terms of concepts and restriction on roles. The theory is divided into two parts:

Terminology Box (TBox) which contains terminological knowledge or definition of concepts and roles, and an Assertion Box (ABox) which contains extensional knowledge. This technique allows the representation of ontologies using ***concepts*** to represent classes of objects, ***roles*** to define relationship between concepts, and ***individuals*** to represent instances of concepts and the values of their roles [32].

Modeling Ontologies using Software Engineering Techniques: this technique uses a software engineering tool called Unified Modeling Language (UML) to model ontologies. This technique is easy to understand and use, especially for users who have no experience in AI. Concepts and their attributes, relations between concepts and axioms can be represented using UML class diagram [32].

Modeling Ontologies using Database Techniques: this technique uses an Entity Relationship (ER) diagram. The ER notation allows modeling classes using ER-entities, attributes with ER-attributes and relations between classes with ER-relations between entities. Formal axioms can be represented as integrity constraints using first order logic and production rules. Instances can be created using Standard Query Language (SQL) through insert statement [32].

The ontology modeling technique that has been used in this work is knowledge representation techniques. This technique allows describing or representing structured domain knowledge using concepts, relations and instances.

2.5.2. Types of Ontologies

According to the information the ontology needs to express and the richness of its internal structure ontologies are categorized as knowledge representation, top-level, task, domain-task, domain, method ontologies and application [32].

Knowledge Representation Ontologies: use classes, relations and attributes to formalize knowledge and capture the representation of primitives to formalize knowledge. The most known knowledge ontology is the frame ontology which is used to capture knowledge representation using the frame based approach.

Top-level Ontologies: are described over the domain of all knowledge. They identify concepts which are universal to every domain or describe very general concepts like space, time, matter, and so on, which are independent of a particular problem or domain.

Task Ontologies: are ontologies that formally specify the terminology associated with the type of task, for example, scheduling, planning, etc. They allow terms to solve problems associated with a task that may or may not belong to the same domain.

Domain-Task Ontologies: are the same as task ontologies but they provide vocabularies to solve problems that belong to the same domain, but not across domains.

Domain Ontologies: can be domain theory ontologies which represent subject areas, such as medicine, geography, government, transportation, and domain data ontologies that contain mainly instance-level information or describe the vocabulary related to generic domain, for example, data ontologies like diagnosing by specializing the terms introduced in the top-level ontology.

Method Ontologies: formally specify the definitions of relevant concepts and relations applied to specify reasoning a process to achieve a particular task.

Application Ontologies: are dependent on a particular application which contains all terms, concepts and relations that are needed to model a particular application under consideration.

2.5.3. Ontology and Logic

The main issue in knowledge representation is on theories and systems for expressing structured knowledge and accessing the capability of reasoning with it. This section explains how logic and rules are used on the ontology to create inferences that manipulate and produce new knowledge, i.e. infer new knowledge from existing information [38].

Logic is the primary reasoning mechanism for problem solving and it is a science that deals with the principles of valid reasoning. Logic has an impact on the organization of knowledge representation and provides language for expressing knowledge with high expressive power. It allows setting criteria to distinguish acceptable and unacceptable arguments [38].

Description Logics (DL) is an important powerful class of logic-based knowledge representation languages. In addition, it incorporates the methods of reasoning and the validity of the results. Moreover, first-order logic provides the complete proof system with consequences. Predicate logic is formulated as a set of axioms and rules that can be used to derive additional true statements [37].

In logic, rule is a major component for constructing valid inferences. Rule-based approaches are flexible and allow the representation of knowledge using conditional rules. One use of rules is to capture commonsense or background information. These rules can be applied to individuals in order to get extra knowledge.

Inference rules are the powerful tools of ontology because they enable to extract new knowledge from previously known information. It allows an application to derive facts that are true but not explicitly stated. Inference is the process of producing new assertions from existing ones by the use of inference engine. The inference engine is the control of the execution of reasoning rules. There are varieties of inference engines that derive additional information in an abstract processing way. Some of them are pellet, SWRL and Jena Rules, Jean and SwiftOWLIM:

- **Pellet**: is an OWL DL inference engine based on a tableaux algorithm. The tableaux reasoner checks the consistency of a knowledge base and all the other reasoning services. Pellet is written in Java and it is an open source. Pellet is the most complete for classification of relatively small ontologies, but performance may be slow for larger models especially for those with large instance data [39, 40].
- **SWRL and Jena Rules**: Rules can either be in SWRL or Jena rules language. SWRL is intended to be the rule language based on OWL and all rules are expressed in terms of OWL concepts (classes, properties and individuals). Jena is used to execute them in both cases [41].
- **Jean Reasoner**: make use of Jena rules in order to infer additional knowledge. Jena rules engine is faster but less complete than other reasoners such as Pellet [41].
- **SwiftOWLIM**: a high-performance semantic repository developed in Java based on TRREE – a native RDF rule-entailment engine and does not provide all of the OWL DL inferences [42].

Out of the above inference engines this study utilizes Jena inference engine which is described in Chapter 4 of this thesis.

2.5.4. Languages and Tools for Ontology Processing

2.5.4.1. Ontology Languages

For any kind of information system development, use of appropriate language and tool is vital. There are several ontology languages that provide mechanisms for creating all the components of an ontology. These languages are divided into two types: classic ontology specification languages and web-based ontology specification languages.

Classic Ontology Specification Languages include [35, 43]:

- ***KIF*** (Knowledge Interchange Format) is one of the major languages for knowledge modelling and exchange based on first order logic and it defines objects, functions and relations with functional terms and equality. It allows knowledge interchange between various information systems.
- ***Ontolingua*** is the most famous language based on KIF and Frame Ontology. It supports the development of ontology by using KIF expressions, exclusively the Frame Ontology vocabulary and by using both languages at the same time depending on the developer preferences.
- ***CycL*** is Cyc's knowledge representation language and is similar to first-order predicate calculus with some expressions.
- ***LOOM*** is based on descriptive logic which provides classification of concepts and uses various kinds of specifications such as an explicit and expressive declarative model specification language.
- ***FLogic (Frame Logic)*** uses first order logic and frame to represent concepts, relations functions and axioms. It includes objects, inheritance, polymorphic types, query methods and encapsulation.
- ***OCML (Operational Conceptual Modelling Language)*** fundamentally focuses on the logical rather than implementation level modelling. The main objective of OCML is to support knowledge level modelling by providing different styles of modelling such as informal, formal or operational.

The World Wide Web Consortium (W3C) recommends a number of standard Web-Based Ontology Specification Languages as part of the Semantic Web stack [35]. These include:

- ***XML (EXtensible Markup Language)*** is a mark up language that provides a format for describing structured data. It facilitates a new generation of Web based data viewing and manipulating applications. It is published as the W3C recommendation standard.
- ***OIL (Ontology Inference Layer***) combines widely used modelling primitives from frame-based languages with formal semantics and reasoning services provided by description logics. OIL is considered as an extension to XOL but it does not support axioms. It is encoded in W3C standards such as RDF/RDF-Schema and XML/XML-Schema.
- ***OWL (Web Ontology Language***) is the most recent developed standard ontology language from the W3C. There are three species of OWL based on their expressiveness feature. These are OWL-Lite used in situations where only a simple class hierarchy and simple constraints are needed. OWL-DL is much more expressive compared to OWL-Lite based on description logic, and OWL-Full is the most expressive one which is used in situations where very high expressiveness is important than decidability.

2.5.4.2. Ontology Tools

A number of development environments or tools are available to build ontologies. These tools are aimed at providing support for the ontology development process. The most known ontology implementation environments are WebODE [43], OilED [44], OntoEdit [45], DAG-Edit [32], Chimaera [46], Protégé [47], and TBC [48]. The following paragraphs present a brief description of these ontology tools.

WebODE was developed using the Artificial Intelligence Lab at the Technical University of Madrid (UPM). It is used for Web servers with Web interface but not used for standalone

applications. The core of this environment is the ontology access service, which is used by all the services and applications plugged into the server, especially by the WebODEs Ontology Editor. WebODE not only allows the edition of ontologies but also provides the development of other ontology tools and ontology-based applications.

OILEd was initially developed as an ontology editor for OIL ontologies, in the context of the European IST On-To-Knowledge project. The tool has an easy to use frame interface and allows users to make use of the web ontology language (OIL). It uses reasoning to support ontology design to facilitate the development of ontologies and intended primarily as a prototype to test and demonstrate ideas. It does not provide functionality for collaborative ontology development such as versioning, integration and merging of ontologies.

OntoEdit was developed by AIFB in Karlsruhe University. It is an extensible and flexible environment, based on a plug-in architecture, which provides functionality to browse, create and edit ontologies. It includes plug-ins (FLogic, XML, RDF/S, DAML+OIL), etc. with two versions: OntoEdit Free and OntoEdit Professional.

DAG-Edit was created for the development of Gene Ontology (GO) bio-ontologies (in Description Logics) based environment. It is an open source software that was implemented in Java that provides an interface to browse, query and edit GO or any vocabulary with a directed acyclic graph (DAG) data structure.

Chimaera is developed in the Knowledge based systems Laboratory at Stanford University. It is a software system that supports users in creating and maintaining distributed ontologies on the Web. The user interacts with Chimaera through a browser using Netscape or Internet explorer. It allows users to load knowledge bases in differing formats, organizing taxonomies, resolving name conflicts, browsing and editing ontologies.

Protégé is one of the most widely used editing tools and has been developed by the Stanford Medical Informatics (SMI) of Stanford University. It is an open source, standalone application with an extensible architecture. The core of this environment is the ontology editor, creation, visualization, and manipulation in various representation formats. Furthermore, Protégé can be extended by way of a plug-in architecture and a Java-based Application Programming Interface (API) for building knowledge-based tools and applications.

TopBraid Composer (TBC) is the recently used editing tool for defining, testing and managing semantic models using the W3C standard languages RDFS, OWL, SWRL and SPARQL. It is a very flexible platform that enables Java programmers to add customized extensions or to develop stand alone Semantic Web applications.

Although, there are different ontology editors that can manage OWL, the one used in this research is TBC Version 3.3.1.1, which provides a flexible plug-in architecture and plenty of different functionalities.

2.5.5. Methodologies for Building Ontologies

Building ontology involves defining classes in the ontology, arranging the classes in a taxonomic hierarchy (subclass - super class), defining properties and describing allowed values for these properties, and filling in the values for properties and instances. The following sections discuss some of the major methodologies in ontology development.

2.5.5.1. Cyc

Cyc is an ontology development methodology created in the 1980s by Microelectronic and Computer Technology Corporation. It is a huge knowledge base and created with common sense

knowledge. It uses its own language called CycL. In order to develop the Cyc different steps are carried out which are briefly discussed as follows [32].

Step 1: Manual coding

Coding of knowledge is done manually because of the lack of handling common sense knowledge in natural language. It also includes encoding the knowledge, examining or checking the rationality of knowledge and augmentation of the knowledge base.

Step 2: Knowledge coding

It is coding the knowledge using an already stored knowledge on the Cyc knowledge base. This process can be performed with the help of tools for analyzing natural language or common sense knowledge.

Step 3: Knowledge Codification

Knowledge codification is mainly performed by tools using an already stored knowledge in the Cyc knowledge base and users are recommended to use only the system knowledge source.

2.5.5.2. Uschold and King's Method

The basic idea of developing ontologies in this method is based on experience on Enterprise Ontology development which is a collection of terms and definitions relevant to business enterprises. The ontology was developed under the Enterprise Project by the Artificial Intelligence Applications Institute at the University of Edinburgh, Scotland. In order to build ontology in this method the following processes must be performed [2, 32].

Step 1: Identifying the Purpose and the Scope

This step determines the domain that the ontology will cover by addressing issues like, for what purpose one is going to use the ontology? for what types of questions the information in the ontology should provide answers? and who will use and maintain the ontology?.

Step 2: Building the Ontology

This process performs three activities under it. The first activity is **ontology capturing**, and it is an identification of key concepts and relations in the domain of interest which are used to capture knowledge. To identify concepts in the ontology Uschold and King's suggested three strategies: Top-down which first identifies the most abstract concepts and then specializing into more specific concepts, Middle-out that identifies the basic terms first then specifying and generalizing terms, and Bottom-up which first identifies specific concepts and then generalizing them into more abstract concepts.

The second activity, **coding**, explicitly represents the knowledge in a formal language after the basic concepts and terms are identified. The last activity in building ontology is **integrating the existing ontologies** which helps to use ontologies that already exist. Checking whether there exists an ontology with respect to a specific domain or not is important to refine and extend the existing resources.

Step 3: Evaluation

After building the ontologies, the evaluation and making technical judgment of the ontologies in the software environment have to be considered.

Step 4: Documentation

It involves the documentation of the product detail in clear and exhaustive way to each and every one of the completed phases and generated products.

2.5.5.3. Gruninger and Fox

This method is also based on the experience in developing TOVE (TOronto Virtual Enterprise) project ontology. It provides a mechanism to build logical model of knowledge that is to be specified on the ontology. To build the logical model, specification that is going to be met by the ontology is informally described, and then formalization of description is made. This method proposed the following steps [32, 49].

Step 1: Identifying the motivating scenarios

According to this method to develop ontologies, specification of application for related scenario is required that will use the ontologies. The motivating scenarios are story problems or examples that describe the requirements of the ontology which are not addressed by the existing ontologies and provide possible solution to the scenario problem (requirement).

Step 2: Formulating informal competency questions

Based on the scenario obtained from step 1, a set of informal competency questions, which are written in natural language, are identified in order to be answered by the ontology. The ontology must be able to represent these questions using its terminology and be able to characterize the answers using the axioms and definitions.

Step 3: Specifying the terminology using first order logic

Once the informal competency questions are available, it is possible to extract set of terms from the questions that will be represented using concepts, relations and attributes in the first order logic language. These terms are useful to specify the terminology in a formal language.

Step 4: Writing competency questions in a formal way using formal terminology

Once the competency questions are posted informally and the terminology of the ontology has been defined, the competency questions are defined formally.

Step 5: Specifying axioms and definitions for the terms in the ontology within the formal language

Ontology axioms are used to specify definition of terms and constraints on their interpretation. They are defined as first order sentences using axioms to define terms and constraints for the object in the ontology. Axioms must be provided to define the semantics, or meaning of the terms. If the axioms are not sufficient to represent competency questions, it is possible to add more objects axiom in order to represent the questions and solutions.

Step 6: Establishing conditions for characterizing the completeness of the ontology

After the competency question is formally stated the conditions under which the solution to the questions must be defined.

2.5.5.4. KACTUS

In 1996 Berneras proposed this method in the KACTUS project. The objective of this project was to investigate the feasibility of knowledge reuse in complete technical system. Developing ontology in this approach depends on the application development, which means every time when an application is built, it exploits its own ontology for the application to represent the knowledge. When an application is developed with this method the following steps are taken into consideration [32, 49]:

Step 1: Specification of Application

This step provides a mechanism to specify context to the application and components view that the application models.

Step 2: Preliminary design based on relevant top-level

To obtain several views of the global model, terms and tasks which are developed during the previous process are used. This design process also searches and considers an already existing ontology that was developed for other applications.

Step 3: Ontology refinement and structuring

In order to arrive at a definitive design, the principles of minimum coupling can be used to assure that the modules are not very dependent on each other and are as coherent as possible, looking to get maximum homogeneity within each module.

2.5.5.5. METHONTOLOGY

This methodology was created in AI lab from the Technical University of Madrid (UPM). It allows knowledge level construction of ontologies including identification of ontology development process, lifecycle based on evolving prototypes and particular techniques for carrying out each activity. Ontology development process refers to the activity carried out on building ontologies. There are three categories of activities to achieve ontology development process [32, 49].

Project Management Activity includes planning, control and quality assurance. Planning identifies tasks to be performed, how tasks will be performed, how much time it takes and what resources are needed for completion. Control, assures that the planned tasks are completed as it was intended to be performed. Quality assurance is responsible for the quality of the product.

Development Oriented Activity includes specification, conceptualization, formalization and implementation. Specification identifies the intended uses of the ontology and the end users. Conceptualization structures the domain knowledge to be meaningful models. Formalization

transforms the conceptual model into formal model and implementation builds computable models in a computational language. Finally, if it is necessary, maintenance updates the ontology.

Support Activity includes knowledge acquisition, evaluation, integration, documentation, and configuration management. Knowledge acquisition acquires knowledge from a given domain. Evaluation makes a technical judgment of the ontologies. Integration reuses ontologies that are already available and incorporating them in order to build a new ontology. Documentation generates clear, detail and exhaustive description of each and every one of the phases completed. Configuration Management records all the versions of the product to control the changes.

2.5.5.6. SENSUS

This method is used for natural language processing. The main idea is linking domain specific terms to broad coverage ontology. Its current content was obtained by extracting and merging information from various electronic source of knowledge. It contains 50,000 concepts organized in a hierarchy including terms which are not domain specific. To build an ontology in a particular domain the following steps are required [32, 49]:

Step 1: Identification of seed terms

Identification of possible terms to describe the concept on the domain has to be carried out.

Step 2: Link manually the seed terms to SENSUS

These seed terms which are identified from step 1 are linked manually to SENSUS.

Step 3: Add paths to the root

All the concepts in the path from the seed terms to the root of SENSUS are included.

Step 4: Add new domain terms

Key domain terms that could be relevant within the domain and have not yet appeared are added.

Step 5: Add complete sub tree

Finally, for nodes that have a large number of paths through them, the entire sub tree under the node is sometimes added based on the idea. If many of the nodes in a sub tree have been found to be relevant, then the other nodes in the sub tree are likely to be relevant as well.

So far methodologies for building ontologies have been discussed and the differences between them depends on: the set of activities performed through the ontology development stages, the degree of dependency of the ontology with the application using it, and strategy to identify concepts from the most concrete to the most abstract, from the most abstract to the most concrete or from the most relevant to the most abstract and the most concrete [32]. Based on these differences Table 2.2 summarizes each of the methodologies.

Table 2.2: Summary of the methodologies for building ontologies

Feature	**Cyc**	**Uschold & king**	**Gruninger & Fox**	**KACTUS**	**METHONTOLOGY**	**SENSUS**
Life cycle proposal	Evolving prototype	Non-proposed	Evolving prototype or incremental	Evolving prototype	Evolving prototype	Non-proposed
Strategy with respect to the application	Application independent	Application independent	Application semi-dependent	Application dependent	Application independent	Application semi-dependent
Strategy to identify concepts	Non-specified	Middle-out	Middle-out	Top-down	Middle-out	Non-specified

Basically, a series of approaches have been reported in Table 2.2 for developing ontologies. During this study, among the various alternatives the Uschold and King's Method is selected to develop the ontology. Detail of the selected methodology is discussed in section 4.3.2.1 of Chapter 4.

2.6. Summary

This chapter explained what document categorization is as one of the IR applications dealing with the steps in automatic text categorization. Ontology modeling techniques such as AI modeling, knowledge representation, software engineering, and database technique were also presented in this chapter. This chapter also reviewed literatures on ontologies for knowledge representation. The different types of ontologies were presented. The use of ontologies for knowledge sharing was also reviewed.

By reviewing the different technologies available in the area of ontology development, we can see that there are major aspects in the field such as ontology encoding languages, ontology editing environments and ontology development methodologies. In the area of ontology encoding languages, several languages are used including OWL, which is selected to encode the knowledge base in this study. Also ontology editing environments are essential tools for developing ontologies regardless of the encoding language.

A series of methods and methodologies for developing ontologies have been also discussed in this chapter. During the study, the selected methodology is based on the one proposed by Uschold and King's.

Chapter Three

RELATED WORK

3.1. Introduction

Different researchers have tried to address the problem of manual document categorization for different languages such as: English [4, 28], Chinese and Japanese [29, 50], Arabic [24] and Amharic [13, 14, 15]. In recent years, Amharic language related researches have led to an increasing awareness of Amharic language resources processing and digital information access. To this end, some works have been carried out such as: Sisay and Haller [51] worked on Amharic word formation and lexicon building, Nega and Willet [21] on stemming, Daniel [52] at the collection of an untagged corpus, Yonas [53] on Ethiopic Online Handwriting Recognition System Using Simplified Ethiopic Script, Tessema [13] on Amharic Search Engine, and Atelach [54] on dictionary-based Amharic English Information Retrieval. Some of the relevant works are reviewed in this chapter.

The works which are related to this study are briefly reviewed in section 3.2 and 3.3. The text categorizations for different languages are presented including Amharic language in section 3.2.2 along with concept-based text categorization.

3.2. Keyword-based Automatic Text Categorization

Many researches on text categorization have been done for different languages such as English [27], Chinese and Japanese [29, 50] and Arabic [24]. These researches categorize a given document into predefined classes using keywords. Some Automatic Text Categorizations for the Amharic language were also done [13, 14, 15]. Zelalem [14] attempted to develop an Amharic News Classifier (ANC) that has the capability of classifying Amharic news items into predefined classes

automatically based on their content. Surafel [22] investigated the application of machine learning techniques to automatic document categorization of Amharic news items. Yohannes [15] has also carried out a research on this domain. All researches were carried out based on keywords. The results of these researches showed that classification can be automated and good results could be obtained. The detailed descriptions of the studies are presented in the subsequent sections.

3.2.1. Automatic Text Categorization for Non-Amharic Languages

The study in [27] focused on the application of K-Nearest Neighbor (KNN) and Rocchio classifier for the English language. After the analysis of these techniques, the authors proposed a new KNN based classifier by unifying the strengths of KNN and Rocchio. According to the paper, text categorization usually comprises three components which are data pre-processing, classifier and document categorization. The data pre-processing step is used to transfer the initial documents into their complete representation. It includes document conversion, word stemming, feature selection, dictionary construction, and feature weighting. The classifier construction implements a function of inductive learning from the training data set and the document categorization implements the actual document categorization.

The studies in [29, 50] present an N-gram language modeling approach for Asian languages text categorization. The studies considered key factors in language modeling and their influence on classification. As mentioned in the papers, there are some difficulties to categorize text written in Chinese and Japanese languages. This is mainly due to the fact that Chinese and Japanese writing systems do not have explicit white space between words. In addition, there is a lack of standard benchmark data sets for these languages. By comparison to three standard text classifiers, which are Naïve Bayes (NB), Ad hoc N-gram and Support Vector Machine (SVM), the language

modeling approach consistently demonstrates better classification accuracies while avoiding word segmentation and feature selection.

The results for the language model classifiers are better than the other approaches for both Chinese and Japanese languages. Furthermore, the experiments indicate, SVM classifiers do not perform well in Chinese and Japanese text classification as they did in English text classification.

The study in [24] deals with automatic classification of Web documents for the Arabic language. The objective of the study was categorizing Arabic web documents into predefined classes. In the pre-processing stage documents are analyzed in order to remove stop-words, vowels are stripped and roots are extracted for words in the document. According to the study, stemming will not yield satisfactory result for categorization of Arabic documents due to the behavior of the language. Arabic is a non-concatenate language and the stem obtained by suppressing of prefix and postfix additions is not the same for words derived from the same origin. The root extraction is concerned with the transformation of Arabic word derivation to their single common root.

After the pre-processing, the classification modules classify a given document into its target class. The study uses learning module, which is a statistical machine learning algorithm, NB, in order to categorize new documents into predefined classes. This module is used to learn from the set of labeled documents based on NB learning algorithm.

3.2.2. Automatic Text Categorizer for Amharic Language

The work of Zelalem [14] is the pioneer in the area of automatic Amharic document categorization. The objective of the study was to investigate the characteristics of Amharic news items for Ethiopian News Agency (ENA) and designed a prototype that has the capability of automatically classifying news items into their predefined classes based on their content.

As discussed in the thesis, automatic document classification has different steps and there are different methods that can be used at each step. The methods are based on machine learning and statistical or natural language processing. Zelalem applies statistical techniques of automatic classification. Statistical techniques include document analysis, generation of document and class vectors based on document and class representatives, and matching document and class vectors to determine the class of a document.

Pre-processing activities were also performed in the study. After a word is identified, characters were searched and replaced to bring common forms of words with different spelling, stemming and stop word removal. During the study, key terms were stemmed using a simple depluralization technique. A database of stop words, which contains the most frequently occurring Amharic words, was also developed by identifying word frequency and calculating document frequency for each of the high frequency words and calculates weight for terms.

In addition to these, matching was done for the automatic classification process, once all the classes are represented through centroid vectors. The system can classify new documents by matching the document vector with the centroid vector of each class. The document is then routed to the most similar class. The similarity between the document vector and the class vectors were computed to determine the class in which the document belongs.

Surafel [22] investigated the application of machine learning techniques to automatic document categorization of Amharic news items. The machine learning techniques NB and K-Nearest Neighbors (KNN) classifier were used. As the author stated, the main requirement of the classification scheme is to provide sufficient background information on any topic. The tool supports different classification methods such as NB, KNN, TFIDF, SVM and probabilistic.

The other research work in this domain is the one done by Yohannes [15]. The objective of the study was to develop or adopt processing tools for Amharic text classification and evaluate the performance of selected classifiers for Amharic text classification tasks. Yohannes's focus was on developing a document pre-processing scheme which facilitates efficient automatic classification of Amharic documents. During the study, the author followed two steps: pre-processing and categorization. The pre-processing activities were performed to prepare the document and they include sampling, category selection, cleaning misclassification cases, data cleaning, data transformation and scaling, data conversion and document processing. Moreover, in document processing word identification (feature word selection to represent a document), stop word removal, stemming, controlling spelling variation and identification of compound words were performed.

The author developed tools to process the source data which are specific to Amharic news documents of ENA. The tools developed by Yohannes are targeted to correct word spelling variations due to pronunciation differences, to correct word variations due to number marker suffixes, to merge compound words (when they may result in semantic loss if separated) written as separate words. Furthermore, enhancement to the suffix and prefix removal tool developed in a previous study to perform semantic analysis before stripping-off affixes from words, and to correct word variations due to gender marker suffixes was done.

3.2.3. Summary of works on Keyword-based Automatic Text Categorization

The investigation on the related works of automatic text categorization for Amharic language shows that the studies have some weaknesses. One of the weaknesses is that all of them consider a single-label classification which assigns a document to a single class. Hence, it is not possible to assign a document to multiple classes. But in a multi-label classification, each document can be

assigned to an arbitrary number of *m* labels of *n* possible classes. In the case of Multi-label classification, a document may contain multiple keywords and assigned to multiple classes. Furthermore, existing keyword-based text categorization methods can categorize a document into a predefined category which is not suitable for a document that includes certain terms in different meanings. They also miss information when different terms with the same meaning about the desired content are used.

3.3. Concept-based Automatic Text Categorizer

As the discussion made in Chapter 2, text categorization can be done based on concepts. Researches that employed concepts in the area of text categorization are discussed in the subsequent sections.

3.3.1. Concept-based Categorizer for Non-Amharic Languages

In this section, research works which are related with concept-based approach are discussed for non-Amharic languages. It presents researches which use ontologies and concepts for text categorization of non-Amharic languages.

The study in [28] proposed a method of using ontology hierarchy in automatic topic identification. The objective of the study was to identify a topic from a given web document for English language using ontology. The authors proposed a method of identifying the topic for a web document by exploiting a web ontology hierarchical structure. Thus, the study stated that the content of the text is better represented by related mapped concepts. Using related concepts, it is possible to capture the semantics relation found among the words in the text.

The fundamental idea behind the study is that, the input of the system is a Web document and the output is the predicted topic of the target document. Generally the system has three main

components: extraction module, mapping module, and optimization module. The extraction module is responsible for extracting important sentences from the document. After all the sentences have been extracted from the Web document, these sentences will be stemmed and sense-tagged.

The mapping module is responsible for accepting the output of the extraction module as an input and mapping keywords onto the words of ontology concept. For keywords which are not mapped to the corresponding concept, the alternative way is used as an extended concept. The extended concept is an option to use external concept as a “middle man” in order to map between concepts and keywords. Additionally, concept weighting was used to show the importance of the concept that discriminate between the important concept and less important ones using keyword frequency and type of mapping. The keyword frequency indicates how frequently the particulate concept was mentioned in the document. Type of mapping indicates whether the mapping uses a concept that belongs to the extended concept or the internal concept. If the keyword is directly mapped to the internal ontology, then the type of mapping will be direct; otherwise the mapping is done from the extended concepts as indirect mapping. The optimization module is responsible to shrink the ontology tree into an optimized tree where only active concepts and the intermediate active concepts are chosen. The study also incorporated an external linguistics knowledge-base to enrich the ontology concepts called the extended ontology.

The study in [29] presented ontology based text categorization for the Chinese language in which the domain ontologies are automatically acquired through morphological rules and statistical methods. According to the authors, to build the domain ontology for a new domain, they collect domain keywords and concepts by finding relationships among keywords. The study adopts a semi-automatic domain ontology acquisition tool (SOAT) to construct a new ontology from a domain corpus.

The study compiled the concepts within documents in a training set and used these concepts to understand documents in a testing set. Knowledge representation framework called Information Map (InfoMap) was used to perform natural language understanding. As the authors stated, the important behavior of InfoMap is that it extracts events from a sentence by capturing the topic words. InfoMap was used as domain ontology as well as an inference engine that consists of concepts, sub-concepts and relationships between concepts. Also the InfoMap serves as a dictionary to segment words and to identify the domain of the sentences and associated keywords.

In this study, for each input news clip C, the system splits it into sentences S_i. The sentences are scored and categorized according to the automatically acquired domains. Thus, every sentence has an individual score for each domain Score (D, S_i). According to the study, to get the total domain score of the entire document, the system adds up the scores of every sentence in the text. The domain which has the highest score is the domain into which the text is categorized.

The study in [64] presented Text Classification Based on Domain Ontology. The authors proposed a text topic classification model based on domain ontology by using Vector Space Model (VSM). According to the authors, to identify a text topic classification it uses association of documents with the corresponding concepts of the ontology. The ontology provides a hierarchical structure which allows treating each concept as a category. This structure is in the form of a tree. In this study, the concepts in different levels have different abstract degrees. The level is higher means the concept is more generalized and contains more eigenvectors. The lower concepts are the sub-classes, instances or hypogyny's attributes, that are concepts more important than the upper ones.

The study constructs the Vector Space based on the Ontology. A model of constructing attribute eigenvector through ontology itself without any training data is proposed. All the non-leaf nodes are called primary concepts. And the leaf nodes are called minor concepts. All hypogyny concepts

of each main concept are used to construct the eigenvectors. So, the primary concepts attribute eigenvectors contain all of its hypogyny attribute eigenvectors and itself.

The study in [55] proposes to find an efficient spam email filtering method using adaptive ontology. The objective of the study was to use an ontology to help classifying emails. They used text classifier as a main tool for email classification, which classifies the incoming messages as spam or legitimate using classification methods.

In this study, four classification methods were evaluated: Neural Network, Support Vector Machine classifier, Naive Bayesian classifier, and J48 classifier. Ultimately, the best classification method was obtained from the training datasets based on the effect of different datasets and different features.

The study builds an ontology and make use of a software called Waikato Environment for Knowledge Analysis (Weka) for implementation of J48 decision tree and Jena to make ontology based on sample dataset. To classify the message as spam or not the study creates a decision tree based on the classification method. Then the decision tree is mapped into the ontology and finally queries are done on the ontology using test data.

3.3.2. Concept-based Study for Amharic Language

So far there is no research on concept-based text categorization for the Amharic language. However, Solomon [56] has done a research on concept-based approach to analyze a semantic technology for Ethiopic Church Manuscript, Art and Music. The objective of the study was to design an ontology that appropriately models the knowledge contained in Ethiopic Church Manuscripts, Art and Music domain of interest within the semantic web.

Solomon discussed that semantic web technologies set a ground meaning, knowledge organization across time and culture through the mapping of theories. Semantic web makes use of ontologies and logic to affect representation, storage and retrieval of knowledge on the Web. Various ontology development methodologies were proposed and are in use for formalizing knowledge contained in a domain of interest. Some of the proposed ontologies are Cyc, SENSUS, KACTUS and METHONTOLOGY. Solomon uses METHONTOLOGY as a methodology in the ontology development process to design, model and develop domain ontologies.

As indicated in the thesis, one of the important tasks in formalism of ontologies is identification of terms and their relationship within the domain of interest. Such a task is improved partly in making use of existing controlled vocabularies, taxonomies and thesaurus definitions in a domain of interest. In developing the prototype, the proposed model has two phases. The first phase was marked by a close participation of the domain expert for reduction of ontology formalization and use of an integrated ontology development platform.

The study formalizes conceptual ontologies using OWL formalism compared with acquisition of sample instance data for consistency and inference checking the phrase. In the second phase, it mainly concentrates on realization of the proposed model. The study spans the design of form-based interface, selection and customization of third party keyword-based search engine, partial digitization of documents into RDF and web pages, and implementation of semantic search engine for a single web domain.

3.4. Summary

This chapter reviewed different research attempts to develop text categorization systems for various languages. The review showed that two major approaches can be followed for text categorization. The first approach is keyword-based automatic text categorization and the second is categorization using concepts.

Keyword-based text categorization uses only keywords to categorize a given document. This approach does not consider the semantics of the document. Unlike keyword-based systems, concept-based text categorization systems use semantics of the document instead of keywords.

Keyword-based text categorization does not possess the common sense knowledge required to extract information from textual representations. As a result, it does not take into account anything about semantic relationships between keywords. As described in Chapter 2, it is possible to use a valid synonymous word that does not exist in the document. In this case, keyword-based text categorization uses keywords to categorize the document. But what if the document describes the same concept with different synonyms?

Documents can be categorized based on concepts instead of using only keywords. Concept-based access to information has important benefits over keyword-based. One of these benefits is the ability to take advantage of semantic relationships among concepts to find relevant documents. The other benefit is the elimination of irrelevant documents by identifying conceptual mismatches.

Currently commercial and experimental concept-based information retrieval tools are available and most of these tools offer text categorization facilities. Therefore, to make the categorization result more accurate, concepts shall be used to categorize text automatically. As a result, this thesis makes use of concepts as a new and promising way of improving the categorization process for Amharic

documents. To achieve the concept-based text categorization, the study manipulates concepts to categorize a given document. Therefore, in automatic text categorization, concept has to be considered to get successful result on the area. So the aim of this study is to come up with a solution to the problem of keyword-based automatic categorization for Amharic documents.

Chapter Four

DESIGN AND IMPLEMENTATION OF CONCEPT-BASED AUTOMATIC AMHARIC DOCUMENT CATEGORIZER

4.1. Introduction

This chapter briefly describes the proposed design and implementation of concept-based automatic Amharic document categorizer to categorize a given Amharic document into predefined categories. In the design and implementation process of concept-based automatic document categorization the major activities are pre-processing, ontology formulation, reasoning and classification. Pre-processing includes lexical analysis, normalization, stop-word removal, stemming, and index term selection.

Ontology formulation is the other main activity which includes identification and building of concepts, relationships, concept taxonomies and build glossary of terms. As presented in Chapter 2, an ontology consists of terms which are organized in taxonomy, their definitions and properties relating them. The reasoning process followed in this study is also described in detail, including the Jena inference engine, concept extraction from the ontology and rules used for the reasoning procedure.

The classifier that classifies a given document into predefined categories is also discussed in this chapter. In order to develop a classifier that best meets the criteria, the necessary design criteria have been identified and presented in this chapter. Moreover, all activities related to the implementation are presented in this chapter.

4.2. Design Criteria

When ontologies are designed, the question of when and how to represent something in an ontology is a design decision. As quoted in [32, 63], Gruber suggested five design criteria for ontologies which are presented below:

Clarity: Definitions should be formal and should effectively communicate the intended meaning of defined terms without any ambiguity by giving appropriate and sufficient conditions.

Coherence: Internal and logical consistency of the axiom definitions must be maintained. Concept definitions in the knowledge base including the rules and restrictions should be coherent.

Extendibility: It should offer a conceptual foundation for a range of predictable tasks, and the representation should be formal and capable to extend the knowledge base.

Minimal encoding bias: The conceptualization should be specified at the knowledge level without depending on a particular symbol-level encoding. An encoding bias results when the representation choices are made purely for the convenience of notation or implementation.

Minimal ontological commitment: An ontology should require the minimal ontological commitment to support the intended knowledge sharing activities. An ontology should make as few claims as possible about the ontology being modeled.

To achieve the above design criteria, this study uses different techniques. For *Clarity* of the knowledge base, each concept is clearly stated without any ambiguities with other concepts. To maintain *Coherence* of the concepts, rule and restriction which do not contradict with one another

were used. *Extendibility* is achieved by giving scope to extend and define new concepts based on the existing terms in a way that does not require much revision of the existing definitions. In the same way, to have the *Minimal encoding bias* generic notations in order to represent the knowledge base were used. Since *ontological commitment* is based on consistent use of terms, ontological commitment can be minimized by specifying only those terms essential to the communication of knowledge. So, the ontological commitment is attained by the use of domain experts in the area of News.

4.3. Architecture of Automatic Amharic Document Categorizer

As mentioned in Chapter 1, the aim of this research work is to make use of concepts as a way of improving the categorization process for Amharic documents. The study also presented the structural design of the system in order to attain the objective. Figure 4.1 shows the general architecture of the concept-based automatic Amharic text categorization system. It is structured into three modules based on the data and process flow between the components. The pre-processing module is responsible to the target document processing. Domain knowledge is represented in the knowledge base module which includes the reasoning process. Lastly, the classification module selects a specific category out of the list of concepts that represent categories. The input of the system is a document and the output will be a concept which is also the predicted category of the target document.

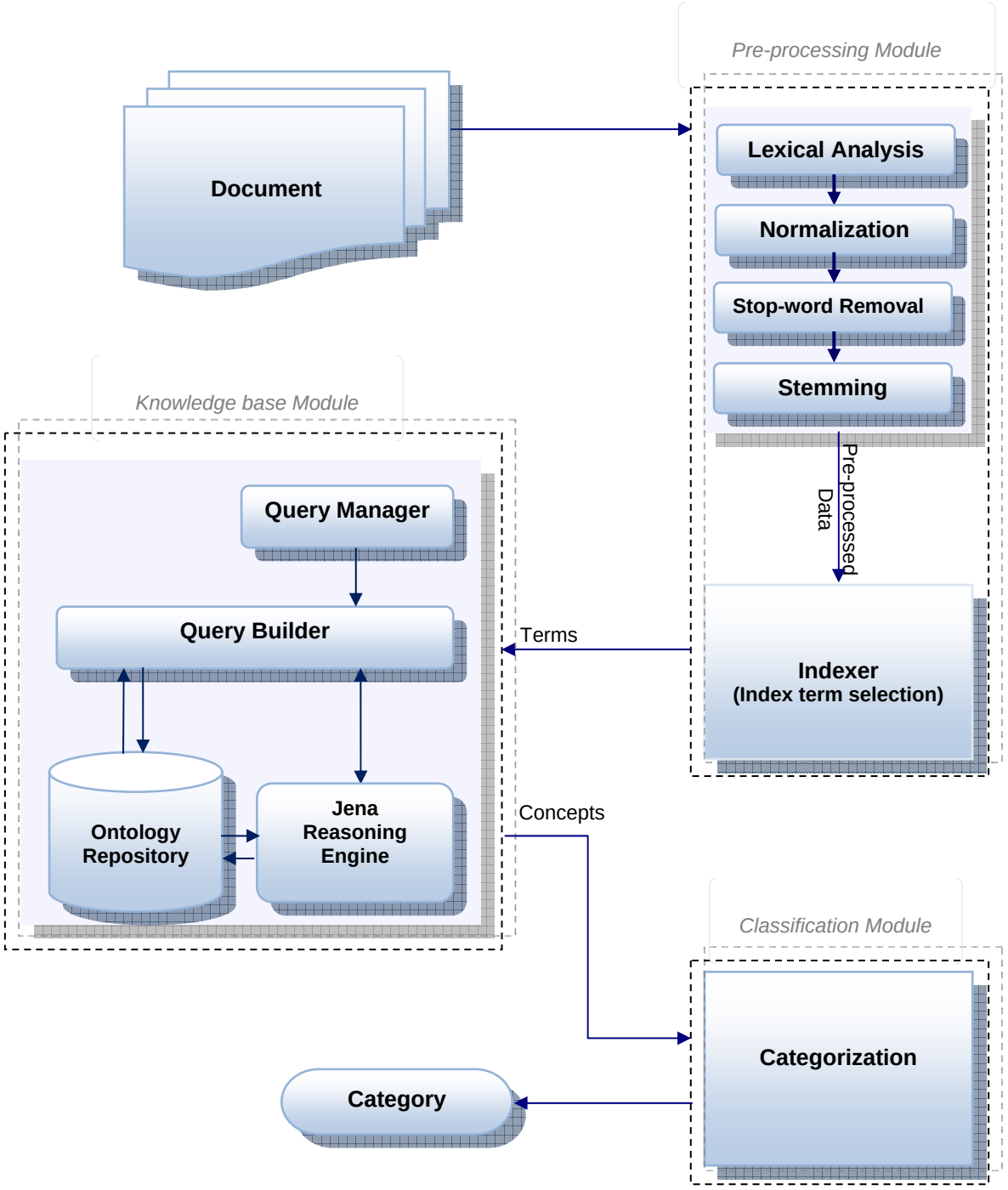

Figure 4.1: The general architecture of concept-based automatic Amharic text categorizer

As shown in Figure 4.1, the input to the pre-processing module is a document that is going to be classified by the system. After acting on the input document, the pre-processing module generates a set of index terms. Then the pre-processing module requests the knowledge base to get the concept where the index term belongs. The knowledge base checks the index terms and returns a specific concept to the classification module. After that, the classification module accepts list of concepts and depending on the information received from the knowledge base, it generates the final output which is the actual category. The succeeding sections explain each module in detail.

4.3.1. Pre-processing Module

The first module of the automatic Amharic document classifier is the pre-processing module. The module is responsible to accept the input document and produce a set of index terms after carrying out lexical analysis, normalization, stop-word removal, stemming, and index-term selection.

For this work, a pre-processing module is adopted from the work of Tessema [13]. In the coming sub-sections, the adopted components are described to show how each method of the module is implemented.

Lexical Analysis

Lexical analysis is the first step in the pre-processing of the input document. Tessema [13] used a string tokenizer to construct words from a sequence of characters. The input for this sub-module is the actual document which is going to be categorized.

This sub-module reads a sequence of characters as a string and tokenizes them using predefined list of delimiters such as new line and space. However, some unnecessary words may be created during the tokenization process. For example, if "." is included in the delimiters list "ዓ.ም" will

be tokenized as "ዓ" and "ም" which is not a keyword. To alleviate such problem, a term with a single character and terms which are less than three characters are ignored. The output of this process is a list of words.

Normalization

As presented in Chapter 2 concerning homophones, in Amharic writing system there are characters with the same pronunciation but different symbols. During this study, homophones are handled automatically by replacement of such characters. For example, the different forms of the word 'Hailu', which are **ሀይሉ**, **ሃይሉ**, **ሐይሉ**, and **ኃይሉ** are all converted to the common form **ሀይሉ** by changing the first character of the three words. Table 4.1 shows sample character replacements used in [57].

This component handles:

- The replacement of Amharic alphabets that have same pronunciation and use, but different representation with common alphabet.
- Shorter form of characters that are usually written using forward slash ("/") and period ("."), for example **ጠቅላይ ሚንስትር** can be written as **ጠ/ሚንስትር**, **አዲስ አበባ** as **አ/አ** **ዶክተር** as **ዶ/ር**.

Table 4.1: Sample of Normalized Characters

Characters to be replaced	Replaced character
ሐ፣ኀ፣ኃ፣ሃ፣ሓ	ሀ
ዐ፣ዓ	አ
ሠ ሡ ⋮ ሦ	ሰ ሱ ⋮ ሶ
ኸ	ከ
ጎ	ገ
ዉ	ው

The next activity done in the normalization is word expansion. The expander accepts a text as a sequence of characters. Then it checks each character of the text and if the character is found to be in short word list then it returns the corresponding expanded form of the text. The final result of this process is an expanded form of a word or the original text.

Stop-word Removal

After the normalization process is done using the procedures discussed in the previous section (Normalization), the existence of the word in the stop-word list is checked. This module accepts list of words and then removes the stop-words. This process removes most frequently occurring words from the document that do not change the meaning of the document.

In this thesis, two kinds of stop words were identified: news specific stop-words such as **አመልክቷል**, **አስታውቋል**, **ገልጸዋል**, **ታውቋል**, etc, and common stop-words, which are manually identified from the collection. In addition to this, stop-words are taken from previous studies [14]. Hence, a total of 462 Amharic stop-words are used.

Stemming

Stemming is performed simply by removing the prefixes and suffixes of the words. As it has been discussed in Chapter 2, Amharic stemmer was developed by other researchers. Tessema [13] implemented a stemming function and in this study we simply adopted the stemmer.

Indexer

During this study, index terms or terms that have the capability to represent the document are selected using Lucene. Lucene is a mature, free, open source, high performance, and scalable information retrieval library. It provides a simple Java API that allows different applications to integrate indexing and searching capabilities [58].

Basically, to index data with Lucene the document must be converted into a stream of plain-text tokens which is the format that Lucene can process. The process of indexing with Lucene is broken down into three main operations: converting the data to a form that can be accepted by Lucene, analyzing, and saving it in the index file.

In the data conversion process, the data to be indexed must be in the format of textual file. Before the data is prepared for indexing, Lucene analyzes the data to make it more suitable for indexing. Lucene has built-in analyzer for English, German and Russian languages. However, it is not possible to analyze Amharic documents. From the previous studies, Tessema [13] has done analyzer for Amharic language, which is also adopted in this thesis.

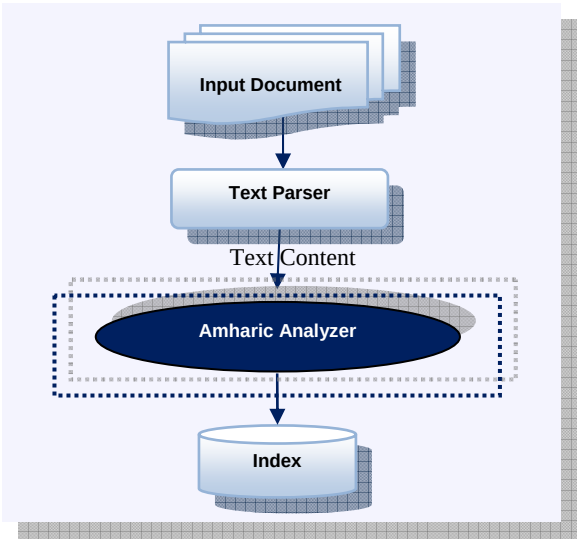

Figure 4.2: Lucene Amharic indexing structure

Figure 4.2 shows the structure of Lucene Amharic indexer. The input document is added to the Amharic analyzer, and then the analyzer first performs the pre-processing operations and stores the index terms in a data structure called inverted index.

4.3.2. The Knowledge Base Module

This module serves as a knowledge base to categorize a given document into predefined categories. To represent knowledge in the ontology, concepts are the main components. Each concept is formalized in the ontology using classes, sub-classes and relationships between classes. During this study, words are used to represent concepts and a single concept can have list of words/terms i.e., a concept can be represented using multiple words. It is possible to say that the ontology is a database of all concepts, words, and categories and how they relate to each other.

Therefore, the ontology contains the represented knowledge in a specific domain. Having that there must be some way of accessing and using the knowledge base in the ontology. To do so, it is necessary to have a means towards mapping terms on top of the ontology concepts and relation between concepts. The content of the document is better represented by these related mapped concepts. Using these related concepts, it is possible to capture the semantic relations found among the words in the text.

The result of this module is a concept, when the pre-processing module requests the knowledge base to get the concept this module returns the corresponding concept to the classification module. The ontology maps the terms with the corresponding concepts and returns a specific concept that a term represents. Figure 4.3 shows the graphical representation of the ontology architecture.

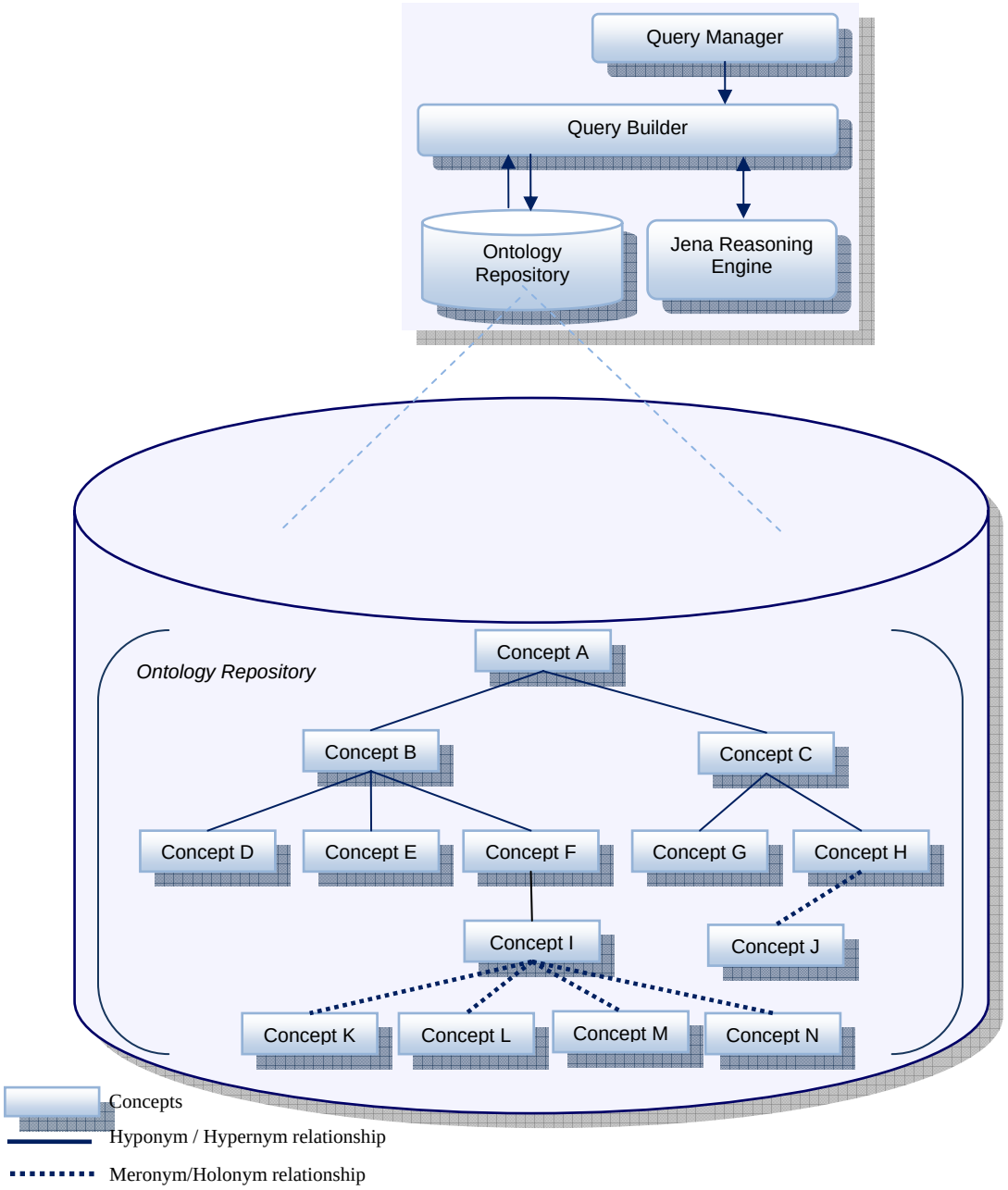

Figure 4.3: The architecture of the ontology

As shown in Figure 4.3, N number of domain concepts can be stored in the ontology and each concept can be represented in M number of other concepts. Relationships between concepts are shown using lines, the dotted lines show the Meronym/Holonym relationships and solid lines show Hyponym/Hypernym relationships.

As depicted in the architecture of the ontology, there are different components and each of them is described as follows:

Query Manager represents components of the reasoner semantic framework for querying, inference and presenting query results by making use of the ontology repository and the reasoning engine.

Query Builder is a component responsible to formulate queries for the index terms to be processed by the query manager.

Jena Reasoning Engine is used to derive additional knowledge which is entailed from the ontology and the rules associated with the reasoner. Rules are integrated to infer knowledge in the ontology, such as attribute values, relation instances, etc. The primary use of this mechanism is to support the use of languages such as RDF and OWL which allow additional facts to be inferred from instance data and class descriptions.

Ontology Repository is the actual knowledge base of the domain area. It contains concepts that are represented using classes, subclasses and relationships among concepts using properties. The capability or knowledge to categorize a document is conceptualized in this repository. Conceptualization consists of objects, concepts, and other entities about which knowledge is being expressed and relationships that hold among them. Furthermore, this component serves as a data store which contains the instances of the concepts or classes.

4.3.2.1. Ontology Development Process

Constructing the domain ontology is an important step in the development of knowledge based systems. The advantage of such domain representation is knowledge sharing and re-use of

knowledge. This section describes the selected methods and methodologies to develop the ontology, and the selected tools and languages. The ontology formulation using the methodologies and the language are also discussed in detail.

Selection of Methodologies

There are a number of methodologies that specifically address the issue of the development and maintenance of ontologies. In Chapter 2, methods and methodologies for developing ontologies have been discussed to represent and formalize the domain knowledge.

During this study out of various alternatives, the methodology which is very much similar to the one proposed by Uschold and King's is followed as shown in Figure 4.4. We have chosen this methodology for the following reasons:

- It allows a way of developing ontologies in a simple and standardized manner.
- It allows inheritance from knowledge engineering which recalls knowledge base systems development in the sense that it clearly identifies an acquisition, coding and evaluation stage.
- It allows building independent applications that are totally independent of the use to which the ontology will be put.
- It allows developing ontologies from scratch.

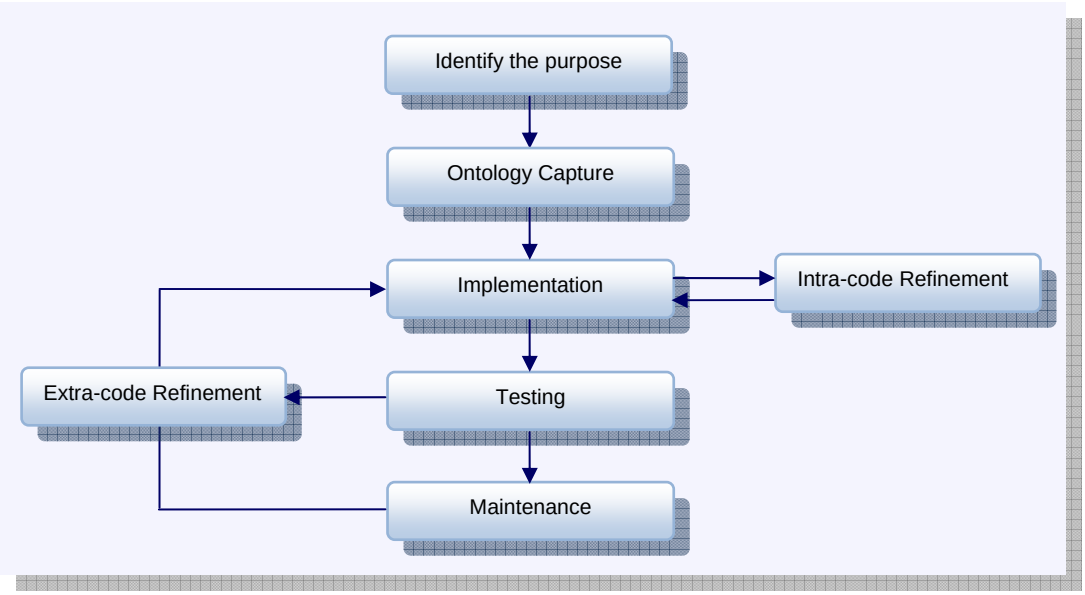

Figure 4.4: The Methodology to develop the ontology

Figure 4.4 shows an overview of the methodology followed for developing the ontology. The steps of the methodology include: identifying the purpose, ontology capture, coding, refinement, testing and maintenance. These steps are described in some detail in the following subsections.

Identify the purpose of the ontology

Before building an ontology, one has to decide what kind of ontology is required, and how the ontology should be built. In order to identify the purpose, it is important to answer the following questions: why is the ontology being built, what is its intended use, and who are its users. In this thesis, the ontology will contain knowledge in the domain of text categorization. It is being built to categorize Amharic news documents through the process of categorization.

Ontology capture mechanism

Ontology capture consists of three different stages: Determining the scope of the ontology, selecting a method to capture the ontology and defining the concepts in the ontology. Determining the scope involves identifying all the key concepts and relationships in the domain.

The process of defining concepts involves taking closely related terms and grouping them as concepts or categories. We selected a top-down approach for categorization due to the following reasons:

- The domain to develop the ontology comprises a hierarchical structure of layers. It uses a top-down approach to provide abstraction of concepts. Therefore, its layers and the elements in these layers will represent the concepts in the ontology linked in a top-down manner.
- Once all the higher level classes are defined, all other lower level classes can be derived from higher level classes.

In order to capture the ontology this study carries out the following activities:

- *Building the glossary of terms*: that includes all the relevant terms of the domain of interest, their descriptions and their synonyms. All possible terms are taken from domain experts.
- *Building concept taxonomies*: when the glossary of terms contains a sizable number of terms, building concept taxonomies to define the concept hierarchy is carried out.
- *Building relationship*: once the taxonomy has been built and evaluated, the conceptualization activity builds relationships between concepts of the same or different concept taxonomy.
- *Building the concept*: once the concept taxonomies and relationships have been generated, we specify which are the properties and relations that describe each concept of the taxonomy in a concept and their instances.
- *Defining rules*: we identify first which rules and restrictions are needed in the ontology and then represent each of them.

Implementation of the ontology

During the implementation phase, the ontology is represented using a formal language. The main classes in the ontology are formalized as concepts with their attributes and properties between concepts. This study uses the ontology development as an iterative process, which involves developing a preliminary ontology that is refined with time.

Refinement

Refinement consists of two phases: Intra-coding refinement, and Extra-coding refinement. Intra-coding refinement involves the refinement that is done during the implementation phase. As the code is being developed, and if somehow some errors are discovered or new requirements come up, the code is refined to correct the errors or fulfill the new requirements. On the other hand, extra-coding refinement refers to the changes done to overcome the errors that are uncovered during testing and maintenance.

Testing

Testing uncovers defects in functional logic and implementation and is carried out at all stages of the ontology development. Once the knowledge base has been created, tests will be carried out to uncover defects in the ontology. Depending upon the problems encountered, appropriate changes will be carried out to the ontology to overcome the shortcomings.

Maintenance

Maintenance is the most important aspect of developing any software including ontologies. Maintenance can be corrective, adaptive or perfective [59]. Corrective maintenance involves considering the problems faced while querying the ontology and correcting the ontology to overcome these problems. Adaptive maintenance involves modifying the ontology to fulfill new

requirements in the future. Perfective maintenance involves improving the ontology for further refinement.

Selection of Tools and Languages

To implement concept-based text categorization system for Amharic documents, ontology is used to represent domain specific knowledge. Once the main concepts are identified and the ontology components are designed, the next step is to implement it using one of the widely used ontology languages. To find out which tool and language would best suit for creating an ontology, tools were assessed according to some significant factors. The main requirements of an ontology language are a well defined syntax, a formal semantics, convenience of expression, an efficient reasoning support system and sufficient expressive power [17, 35, 38]. All the possible tools were evaluated according to the following criteria [17]:

- Software architecture and evolution
- Interoperability with other tools
- Knowledge representation
- Inference services attached to the tool
- Usability
- Well defined syntax and semantics

In Chapter 2, different languages for building ontologies have been discussed. For the purpose of this study, OWL is selected as ontology development language. The language is preferred based on the above requirements.

In addition to the language, TopBraid Composer (TBC) is selected as an implementation environment for the knowledge base of the system. As presented in Chapter 2, TBC is a modeling

tool for the creation and maintenance of ontologies. It is a complete editor for RDF/OWL models, as well as a platform for other RDF-based components and services. TBC is built upon the Eclipse platform and uses Jena as its underlying API which provides the following facilities.

- TBC uses OWL Description Logic to run logical consistency checks and to classify classes and instances. The system has an open source Description Logics Pellet reasoner built-in as its default inference engine.
- TBC can also handle traditional rule bases either in the Jena Rules format or SWRL. Both types of rules are executed with the internal Jena rules engine to infer additional relationships among resources.
- TBC can visualize arbitrary relationships among RDF/OWL resources in a graphical format.
- It is simple to import and export: Just to choosing a local file, it will be opened in multiple windows.
- Navigation is easy and users of TBC can choose the file in an easy way.

4.3.2.2. Ontology Formulation

One of the fundamental activities of this research work is to design and implement an ontology that appropriately models the knowledge in a specific domain. The ontology developed for this study has its own purpose, domain area and scope.

Purpose: The purpose of the ontology is to represent knowledge that should have the capability to categorize a given Amharic document into predefined categories.

Domain: Amharic News is the domain of interest to demonstrate the proposed design. The domain is selected because of the availability of the corpus for knowledge

representation and testing. The ontology should be flexible and extensible enough to handle additional concepts from this domain.

Scope: The scope of the ontology is limited to have some knowledge that will aid us in determining whether a given Amharic document should be classified to a specific category or not. In short the scope encompasses formulating domain concepts, building relations between concepts and representing restrictions.

Based on the methodology selected to develop the ontology, this study formulates the ontology using glossary of terms, concept taxonomies, and relations. Each of them is described below.

Build glossary of terms

During the development of the ontology, terms which are relevant to express the domain knowledge are first obtained from domain experts in the area. From ENA (Ethiopian News Agency) two domain experts were used to extract their knowledge. To achieve the ontological commitment, it is a must to use expertise on the selected area. In this study, all the knowledge for each category is taken from the experts to have an ontological commitment. After the terms are prepared from the expertise, glossary of terms is built using the terms.

Build concept taxonomies

Typically the glossary of terms contains around 1786 terms. This much sizable number of terms demands to build concept taxonomies to define the concept hierarchy. The concepts hierarchy is build based on the predefined categories in the Amharic news domain. Figure 4.5 shows a sample of the graphical representation of the `Terms` concepts glossary.

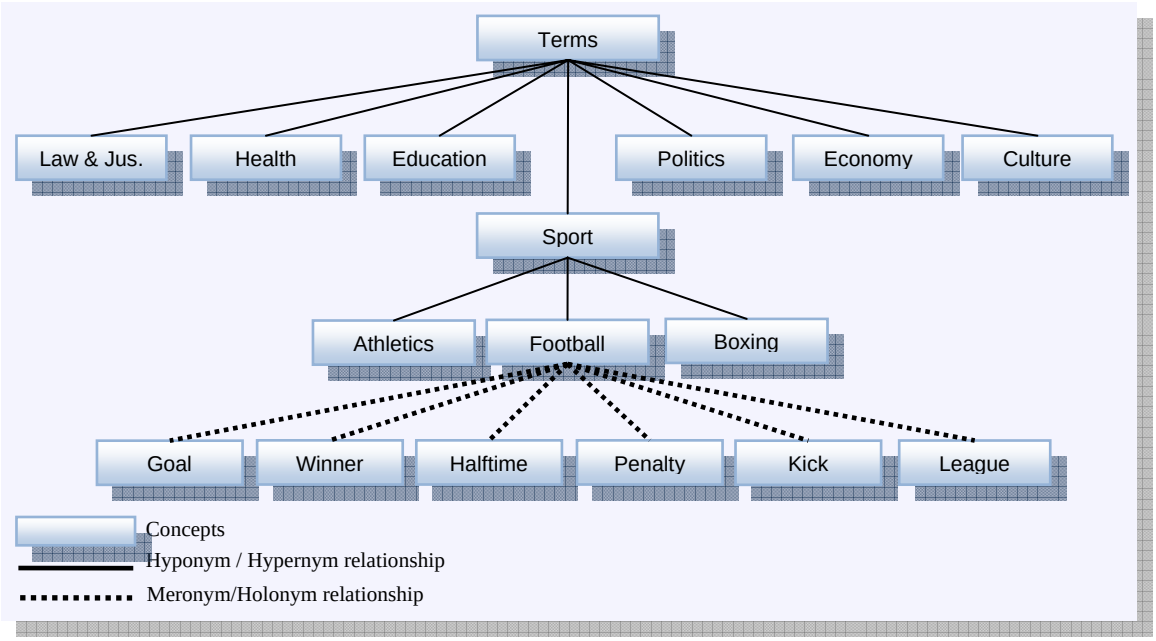

Figure 4.5: The Term concepts taxonomy

It shows all the concepts of the taxonomy called `Term`, which contains all possible terms or synonyms per category. Each category has its own list of terms that represent the concepts. For example, the category `Football` has terms such as `Penalty`, `Goal`, `kick`, `League`, and so on. Once the taxonomy has been built, the next activity is to build relationship between concepts of the same or different concept taxonomy.

Build Relationship

As discussed in Chapter 2, concepts are linked by various kinds of relationships. To do so, **Properties** was used to create relations between individuals i.e. properties link two individuals together and precisely applied to a class. In OWL there are various kinds of properties such as: ***ObjectProperties,*** which relates objects to other objects; ***DatatypeProperties,*** which relates objects to data type values; ***FunctionalProperty*** defines a property that has at most one unique value for each object; ***InverseFunctionalProperty*** defines a property for which two different

objects cannot have the same value, and ***SymmetricProperty*** defines a symmetric property. As depicted in Figure 4.6, properties or relationships are used in this study such as:

news:hasProfessional: relates the concept called Team and Professional. Each team has professionals including the concepts Player, Athlete, Coach and Referee.

news:IsA : relates the concept called Goalkeeper and Player. It shows that Goalkeeper is also a type of player, and the same is true for Attacker, Midfielder and Defender.

news:hasCategory: this property relates concepts called News and Category. News has a specific category and sub-category. The main category can be sport and the sub-category can be Football or Athletics.

news:kindOf: relates concepts Sport with Footbal and Athletics. Footbal is a kind of sport and Athletics is also a kind of sport.

news:hasAgency: relates concepts News and Agency. News has a specific agency that organizes the News.

news:hasMedium: relates concepts Agency and Medium. Agency uses some ways of presenting the news and each Agency has Medium.

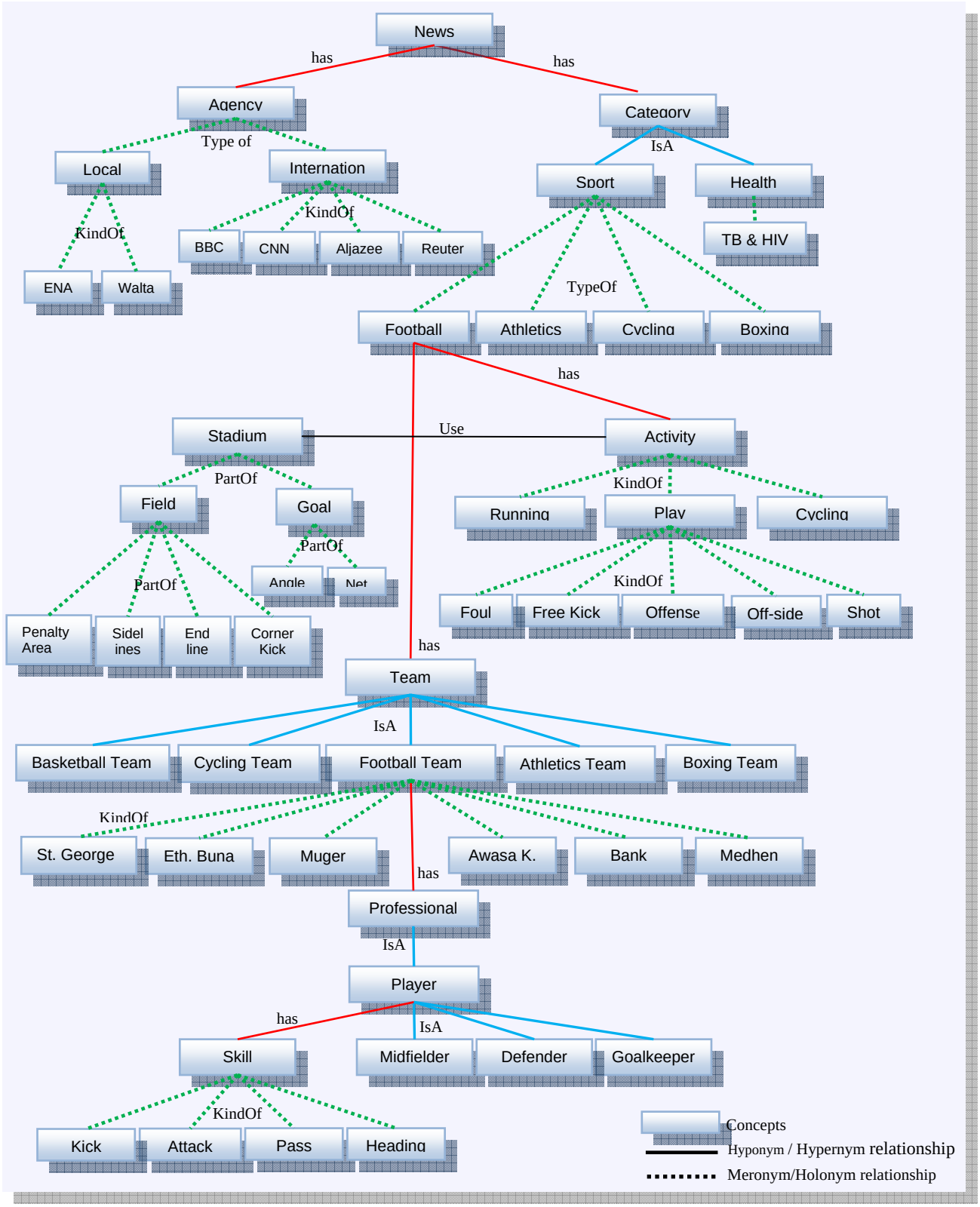

Figure 4.6: Concepts and Relations in the News ontology

In addition to the above relations, there are also relations used in the `News` ontology: ***owl:sameAs, owl:equivalentClass, rdfs:subClassOf,*** and others as listed in Annexes B and C.

Build the Concepts

After the concept taxonomies and relationships have been identified, the next activity is defining domain concepts and their instances. Based on the ontology development process, various concepts are formulated as knowledge base. The `News` ontology consists of domain concepts and their related sub-concepts such as, categories, attributes, and activities.

As depicted in Figure 4.7., the developed `News` ontology contains the following high level concepts: `News`, `Category`, `Accidents`, `Economy`, `Environmental Preservation and Weather Condition`, `Science` and `Technology`, and `Sport`.

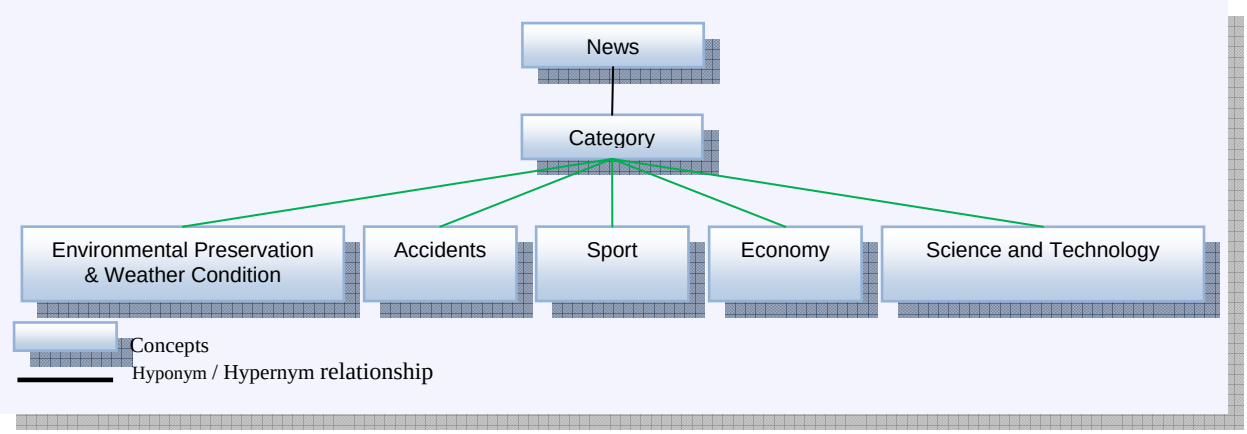

Figure 4.7: High level concepts in the News ontology

In the same way, each of the high level concepts has sub-concepts or second level concepts. As shown in Figure 4.8., these concepts have their own sub-concepts, where categories (classes) are organized in the form of a hierarchy.

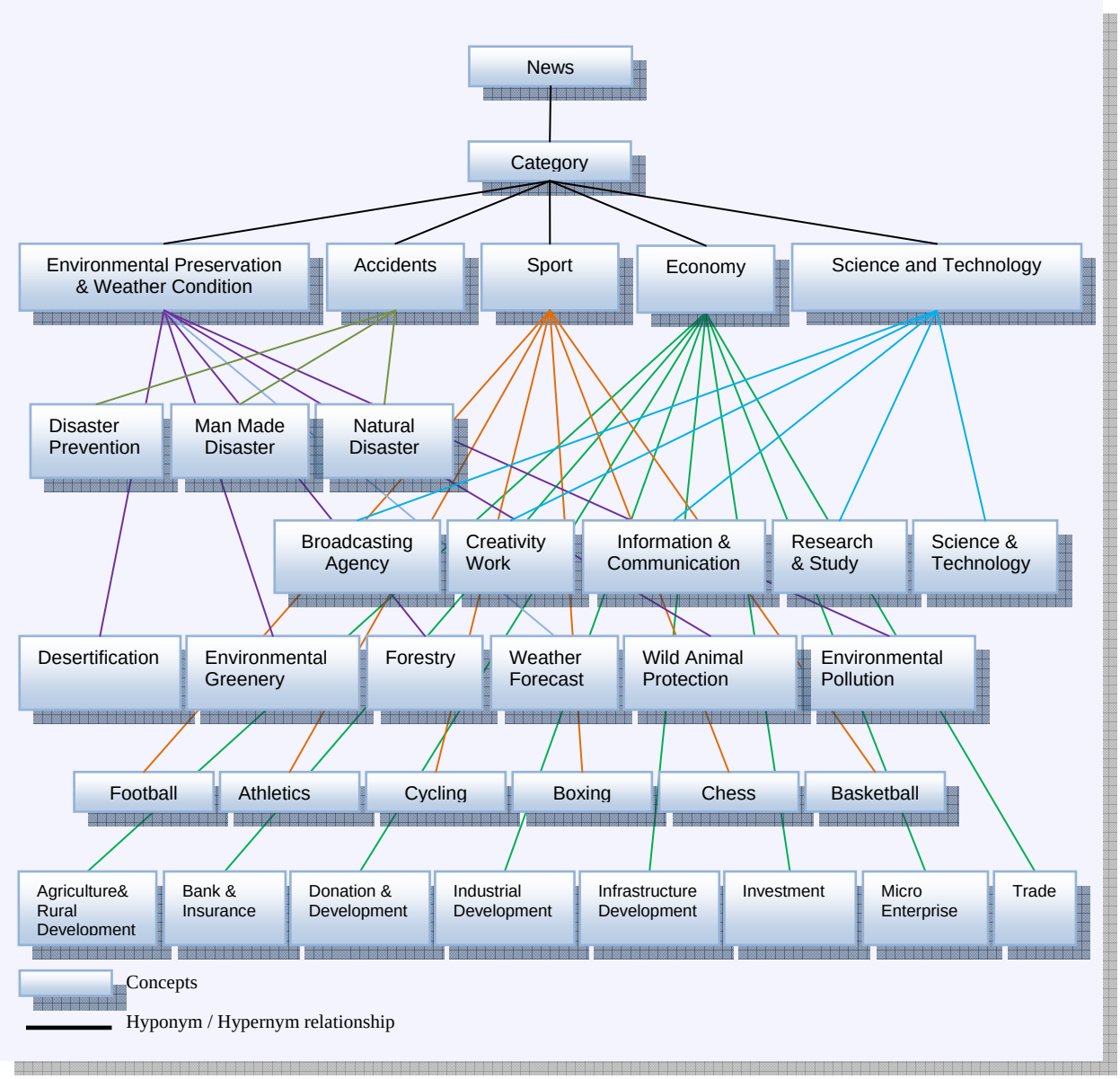

Figure 4.8: Second level concepts in the News ontology

The `News` ontology is used to represent all possible categories (`Accidents`, `Economy`, `Environmental preservation and Weather Condition`, `Science and Technology`, and `Sport`) and sub-categories. The concept `News` is represented based on the attributes of the domain. The basic attributes of the concept `News` are: `title`, `Agency`, `Medium`, `location`, `source`, and `publication date`.

The high level concepts in the News ontology are described as follows:

- Economy: This class is used to represent the knowledge related to economy including the related concepts such as: agriculture, development, donation, investment, banking and insurance, production, distribution, exchange, and consumption of goods and services, and so on.
- Environmental preservation and Weather Condition: knowledge represented in this class is related to the environmental, forestry, weather forecasting, animal protection, pollution, desertification, and so on.
- Sport: This class contains different concepts that should be incorporated in the knowledge base. For example the concept called rule is included in the sport class, which is used to represent sport rules like, in a regulation of soccer game, each team has 11 players. The players can use any part of their bodies to hit the ball, except their hands or arms. Players generally use their feet and head as they kick, dribble, and pass the ball toward the goal. One player on each team guards the goal and tries to prevent the other team from scoring. Such rule is represented in the ontology using the concept Rule. The other main concept in the sport class is Team. The concepts named Professional and Player are hierarchically represented in this concept. Team stores the knowledge about players. In a regulation of football game, each team has professionals Player. One player from each team plays the position of goalkeeper, defender, midfielder, and attacker. Moreover, concepts related to players basic skills: kicking, dribbling, passing, heading, and trapping are incorporated in this concept. Figure 4.9 shows the graphical representation of the Team ontology.

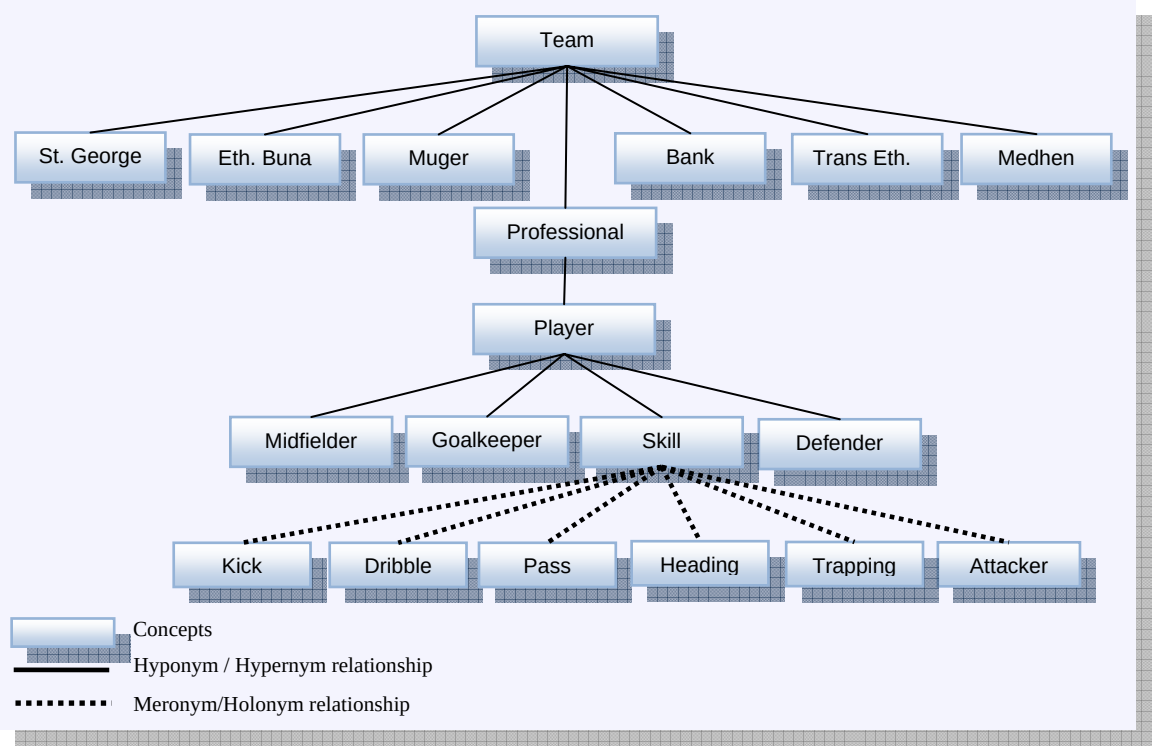

Figure 4.9: The Team ontology

- `Science and Technology`: this class represents concepts which can be mentioned related to science and technology such as science, information and communication, creativity, research and studies and broadcasting agency. For example, the concept `Agency` is used to capture information about an organization that gathers information about events and supplies it to the media. The attributes that describe the agency are `name, description, site of the agency`, and `the medium`. This concept has a relationship with the concept `News`, which means, a news can be gathered and organized by some agencies (a minimum of one agency). The concept `Agency` is graphically represented in Figure 4.10.

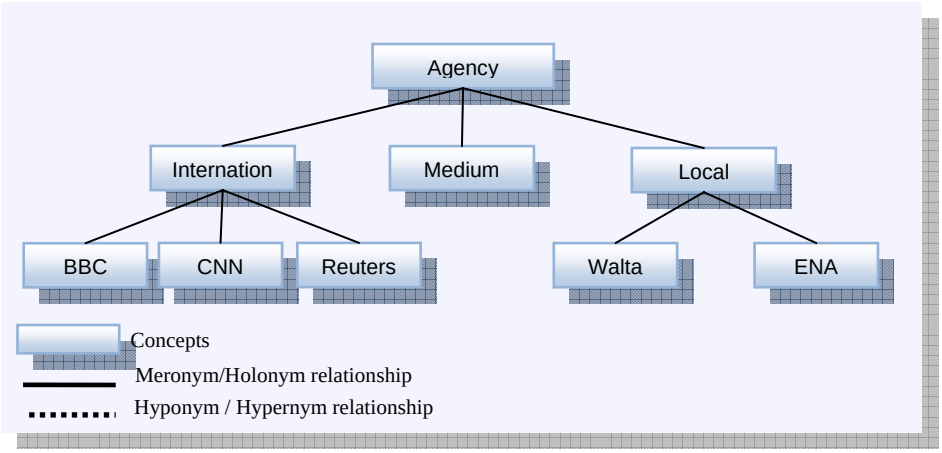

Figure 4.10: The Agency ontology

- `Accidents`: this class contains knowledge related to manmade disaster, natural disaster and disaster prevention. For example, the concept natural disaster incorporates gradual and sudden type of disasters such as earthquake, flood, tornadoes, etc.

To implement the ontology, **concepts** or **classes** are used as the main entities of OWL. They are interpreted as sets of individuals in the domain or sets that contain individuals. A class contains descriptions that specify the conditions which must be satisfied by an individual for it to be a member of the class. In OWL classes are defined using ***owl: Class*** element and listed in Annex A.

Besides the ***owl:classes*** and ***owl:properties, Objects*** are significant in order to make the knowledge representation concrete**.** ***Objects*** or instances listed in Annex D are interpreted as particular individual of a domain.

As shown in Figure 4.11, the class `Activity` contains all the activities of football such as `play`, `kick`, `shot`, `curving shots`, `off-side`, `offense`, `free kick`, and `goal kick` as an instance of the class play. Such basic activities have the capability to represent the concept of playing football.

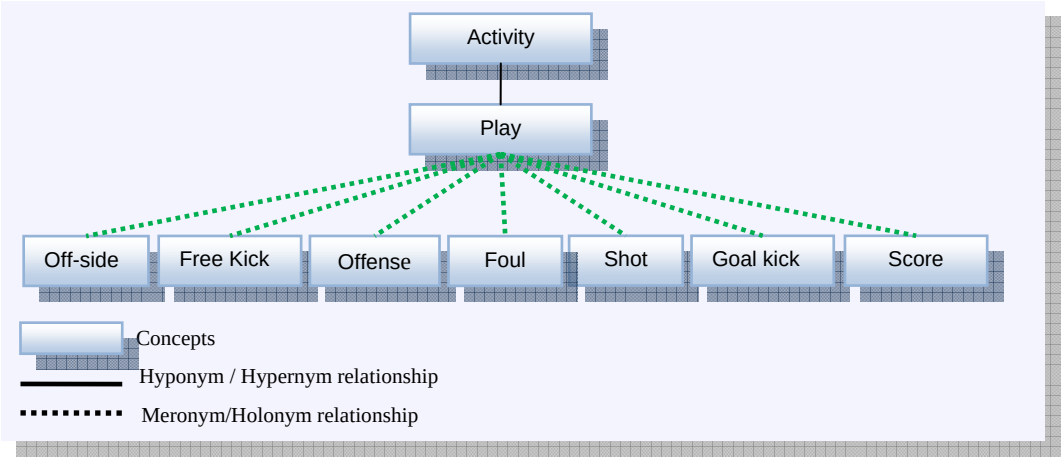

Figure 4.11: The Activity ontology

Some of the instances used in the `News` ontology are presented below including the description of the instances `owl:subclassOf`, `rdfs:labels` and relation `IsA.`

```
news:EthiopiaBuna
      a news:FootballTeam ;
      rdfs:label "ኢትዮጵያ ቡና" ;
      news:IsA news:Team ;
      =      news:St.George, news:MugerSymento, news: Kiraybetoch.
news:St.George
      a news:FootballTeam ;
      rdfs:label "ቅዱስ ጊዮርጊስ" ;
      news:IsA news:Team ;
      =     news: EthiopiaBuna, news:MugerSymento, news: Kiraybetoch.
news:Attacker
      a news:Player ;
      rdfs:label " አጥቂ" ;
      news:IsA news:Player ;
      =       news:Striker .
news:Goalkeeper
      a news:Player ;
      rdfs:label "ግብ ጠባቂ" ;
      news:IsA news:Player .
```

From the above instances, `EthiopiaBuna` and `St.George` are created from the concept `team`, and `Attacker` and `Goal keeper` are instances of the concept `player.` For example, in the instance `EthiopiaBuna,` the description shows `EthiopiaBuna` is a Football team, the label of the instance is "ኢትዮጵያ ቡና", it is a subclass of `team` and the instance `EthiopiaBuna` is the same as or equivalent class with `St.George, MugerSymento` and `Kiraybetoch.`

4.3.2.3. Reasoning

So far, pre-processing with the aim of extracting document representative index terms and ontology formulation to represent the knowledge are presented. However, having the represented knowledge and pre-processed data is not sufficient to categorize the document successfully. Besides the knowledge representation and pre-processing, reasoning is desirable to make use of concepts in the classification process.

The whole idea behind the reasoning process is making the classifier have a reasoning capacity using inference engines. There are various inference engines that derive additional information in an abstract processing way. The most known and commonly used are Pellet, SWRL and Jena Rules, Jean Reasoner, and SwiftOWLIM.

Even though, in Chapter 2, different types of inference engines are discussed, Jena rules engine is faster than the others. Due to this reason and the availability of resources, Jena semantic framework is used to implement the reasoning capability. Jena is an open source toolkit for processing RDF, OWL and other semantic web data and it is composed of RDF Processing API, OWL Processing API, A rule-based reasoning engine and SPARQL query engine [43, 61].

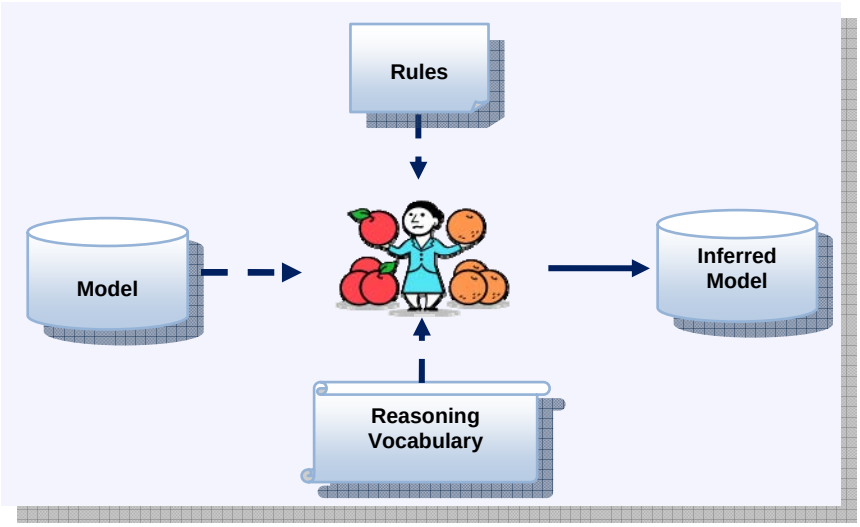

Figure 4.12: Jena Inference Mechanism

Figure 4.12 shows the overall structure of Jena inference mechanism, including Model, Rules, Reasoning vocabulary and Inferred Model.

- **Model** is one of the key components of Jena's approach to handle RDF/OWL data. This model contains concepts, properties and instances in graphs represented by the Model interface. The Model is represented as a set of statements (subject, predicate and object).
- **Rules** are the other main component of this engine. There are predefined relationships in the ontology. Besides those rules, additional rules are used in this study to create relationships between concepts.
- **Reasoning Vocabulary** is the other component for querying, inferencing and presenting query results using the designed ontologies. A common feature of Jena reasoner is that it creates a new model which appears to contain the triples that are derived from the reasoning process.
- **Inferred Model** contains additional information derived from the data model based on rules. This model is created using data, rule and inference mechanism. Moreover, it also helps to gain more control over the processing or to access additional reasoner features.

Rules of Reasoning

In the concept extraction activity, it shows how the system extracts the represented knowledge using the relationships between concepts. For example, there is a concept called `Team` that belongs to the concept `Football` and `Athletics`. So to create a relationship between these two concepts it is required to use additional relationships. But how does the system manage to know the relationship between the two concepts? The identification of concept relationship is carried out by Jena reasoner via ontology inference. To do so, the reasoner has to define a set of reasoning rules. To create this, relation rules must be defined, which contain the triple (subject, predicate, and object) pattern that describes the relations between concepts. Rules defined and used in the reasoning procedure are listed in Annex F.

From the list of rules listed in Annex F, for example "`Rule1`" is used for further reasoning. These rules are necessary in Jena to create relation between concepts `Team` with `Football` and `Athletics`. For example,

```
[Rule1:
      (news:Football rdfs:subClassOf news:Sport)
      (news:FootballTeam rdfs:subClassOf news:Team)
->    (news:FootballTeam rdfs:subClassOf news:Football)
]
```

This rule describes the relation between the concept `Football` and `Team`. If the concept `Football` is subclass of `Sport` and `FootballTeam` is a subclass of `Team` then the above rule makes the `FootballTeam` as a subclass of `Football`.

```
[Rule2:
      (news:Football rdfs:subClassOf news:Sport)
      (news:Soccer rdfs:subClassOf news:Synonym_sport)
->    (news:Soccer owl:equivalentClass news:Football)
]
```

Rule 2 declares equivalence of concepts, which is interpreted as the concept `Footbal` is equivalent or the same as to the concept `Soccer`.

Concept Extraction from the Ontology

As mentioned before, a concept is associated with its term, definition, property and instance. Using Jena, specific to this implementation, accessing the represented knowledge in the ontology and using the feature of reasoner based on rule is done as follows:

Step 1. Creating the Model Factory Class. In this study, access to the inference mechanism is done using model factory. As mentioned before, there are different types of reasoners and for each reasoner there is a factory class (reasoner factory). After creating the reasoner, a set of RDF/OWL data is attached to create the inference model that contains additional information from the inference engines.

```
Create the initial model from the ontology
Create the Ontology Model using Model Factory method called
      createOntologyModel (ontology file);
Create the complete Model from the ontology and rules
Create Generic Rule Reasoner using the method called
      GenericRuleReasoner(Rule file));
Create Inference Model using the method called
      createInfModel(Generic Reasoner, Ontology Model);
```

Step 2. Creating the query string, which is a textual way of accessing the knowledge stored in the repository using QueryFactory. The QueryFactory has various Create methods to read textual query from a file or from string and return a QueryObject which encapsulates a parsed query.

The query string is constructed using SPARQL which is a W3C standard recommendation to query RDF graphs. It comes with a notation similar to the relational database query language SQL, but focuses on triple matching.

```
Create the query string
Create query using the method called
      QueryFactory.create(queryString);
```

Step 3. Creating instances of QueryExecution, using the query string and the model created in step 2.

```
Execute the query string using the method
      QueryExecution(query Inference Model) ;
```

Step 4. Execute the execSelect to execute the Query Execution and the method returns the result set.

```
Access the query result using
      ResultSet rs = qexec.execSelect() ;
```

Step 5. Accessing the result set, that allows iterating over each query solution. Alternatively ResultsetFormatter can be used to output the query results in various formats. Queries to the created model return not only those statements that were presented in the original model but also additional statements that can be derived from the data using rules or other inference mechanisms implemented by the reasoner.

Throughout the process of ontology formulation, domain concepts are represented using classes and relations. In addition to the knowledge base by means of reasoning, the system has the ability to reason out and create additional concepts using the knowledge base and the rules. Finally, the categorization process continues as described in the next section.

4.3.3. Classification Module

List of concepts from the knowledge base module serve as input for this module and identification of where the document belongs to, i.e., the target category is selected on a specific concept from list of concepts. An input document for this process is accepted from the user which is going to be categorized. To facilitate this, a simple user interface as shown in Figure 4.13 is provided to accept the path of the input document.

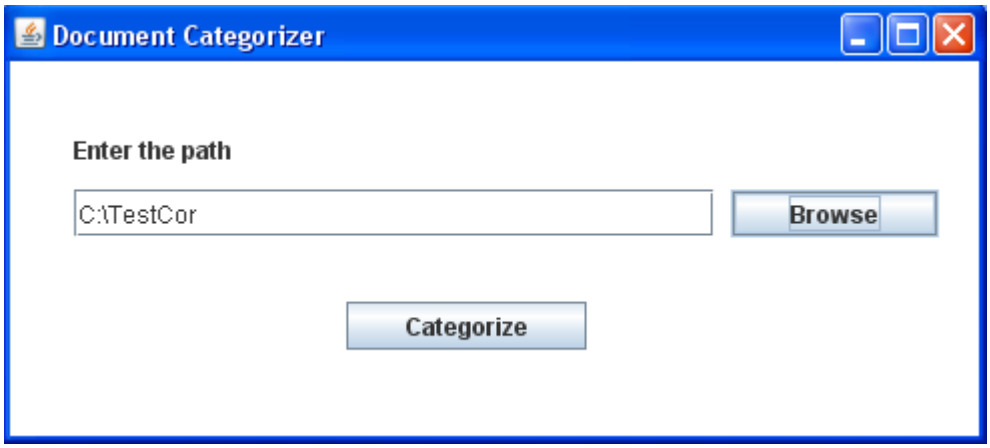

Figure 4.13:User Interface

For example, the user selects a document given in Annex E, and then the selected document passes through the pre-processing phase. As the description made earlier in section 4.3.1, index term selection is done using Lucene indexer. Table 4.2 shows the index terms for the selected document with their frequency greater than one.

Table 4.2: Sample of index terms for the selected document

No.	Index Term	Frequency
1	ማለፍ	2
2	ምት	2
3	ሩጫ	2
4	ሴት	2
5	ቅጣት	2
6	አሸናፊ	2
7	አትሌቲክስ	2
8	አገልግሎት	2
9	እግር	2
10	ክለብ	2
11	ክልል	2
12	ወንድ	2
13	ወጣት	2
14	ውሀ	2
15	ዞን	2
16	ዳኛ	2
17	ግብ	2
18	ጥሎ	2
19	ጽህፈት	2
20	ኪሎሜትር	3
21	ስፖርት	4
22	ከተማ	4
23	ዋንጫ	4
24	ወረዳ	5
25	ቡድን	7
26	ውድድር	8

Lucene generates around 78 potential index terms for the selected document. Out of these 78 terms, 26 of them are used as index terms. These index terms are chosen based on their frequency. In this study, through experiment index terms are selected with frequency greater than or equal to two. Choosing index terms with n number of frequency depends on the condition; it is possible to take the value of n from one up to the maximum number of frequency.

Figure 4.14 shows the analysis made in order to select the number of index terms for the document. Through this process, the analysis is made on the Sport document collection, out of the 170 documents the number of documents categorized with frequency greater than or equal to one is 5, greater than or equal to two is 109, greater than or equal to three is 32, greater than or equal to four is 7, and greater than or equal to five is 18. Specific to this thesis, the decision made on the frequency of index terms is based on the maximum number of documents to be categorized. Most of the documents are categorized with the frequency of two and above. Therefore, throughout the experiment, index terms are selected starting from two up to the maximum frequency.

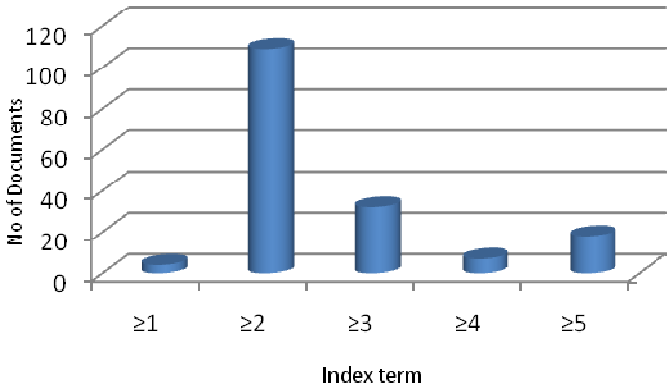

Figure 4.14 Categorized no of documents per index term frequency

Once the index terms are identified for the selected document, the next step is to inquire the knowledge base to get concepts. The index terms which are extracted from a given text will be mapped onto their belonging concepts in the ontology. In the knowledge base, concepts are extracted based on terms using the ontologies. The pre-processing module queries the ontology by passing index terms and then the knowledge base returns the concepts to the classification module where the term belongs to.

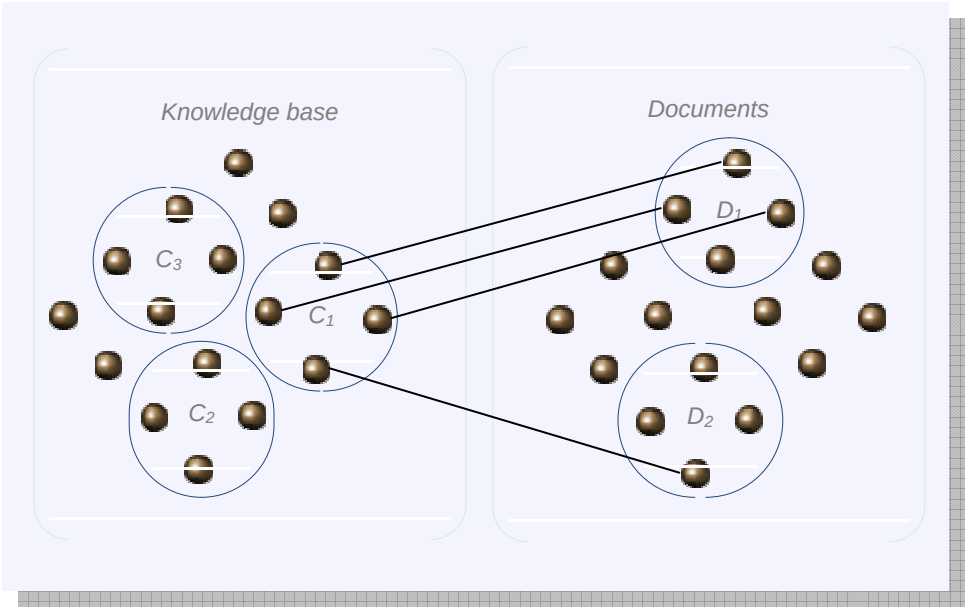

Figure 4.15: Mapping between Documents and Concepts

As depicted in Figure 4.15, each index term is mapped to the corresponding concepts in the knowledge base. For example, document D_1 contains n number of index terms D_1 (DT_1, DT_2 DT_3…DT_n), and the concept C_1 is represented in n number of concept terms C_1 (CT_1, CT_2, CT_3…CT_n) in the knowledge base. So, the mapping is done from the document term DT_i to the concept term CT_j.

As shown in Figure 4.16, the system checks whether or not the index term exists in the knowledge base. If the concept is found, it associates the document term with the corresponding concept. However, there is a possibility that the index term may not be able to be mapped onto its corresponding concept because there is no such concept available in the `News` ontology. This situation requires an alternative way to map the index term onto the external knowledge.

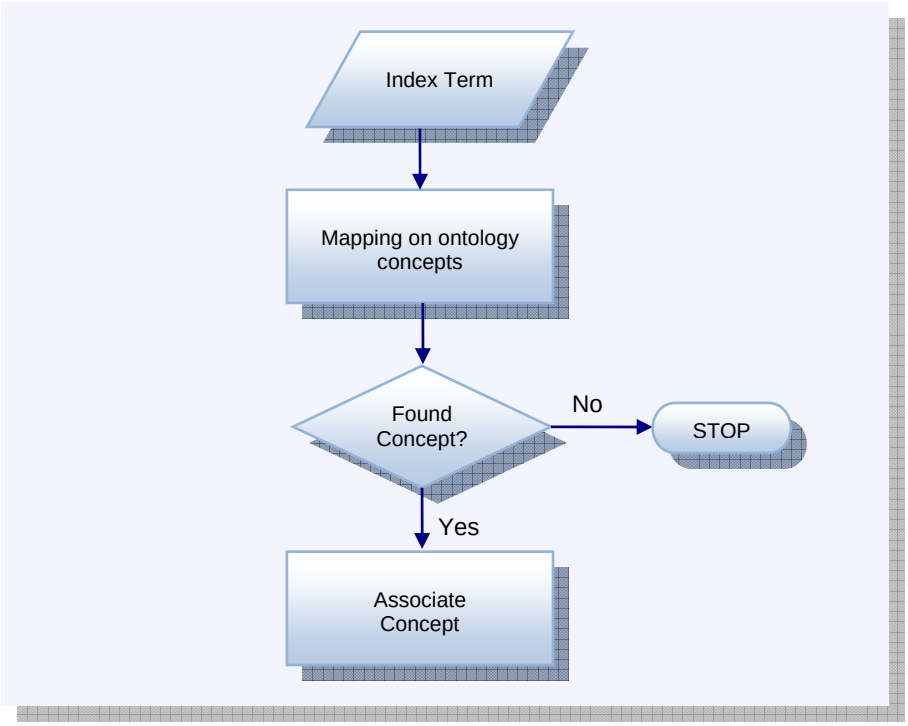

Figure 4.16: The flowchart of the mapping between terms and the News ontology concepts

For the previous example, index terms for the selected document corresponding to their respective concepts are given in Table 4.3.

Table 4.3: Sample of index terms with the corresponding concept

No.	Index Term	Concept
1	ማለፍ	Football
2	ምት	Football
3	ሩጫ	Athletics
4	ሴት	*
5	ቅጣት	Football
6	አሸናፊ	Sport
7	አትሌቲክስ	Athletics
8	አገልግሎት	*
9	እግር	Football
10	ክለብ	Sport
11	ክልል	*
12	ወንድ	*
13	ወጣት	*

No.	Index Term	Concept
14	ውሀ	*
15	ዞን	*
16	ዳኛ	**Both**
17	ግብ	Football
18	ጥሎ	*
19	ጽህፈት	*
20	ኪሎሜትር	Athletics
21	ስፖርት	Sport
22	ከተማ	*
23	ዋንጫ	Sport
24	ወረዳ	*
25	ቡድን	Sport
26	ውድድር	Sport

Note: the * symbol indicates that the knowledge base returns nothing for the specified index terms.

The final result of this module is a single category where the document belongs. To do so, as shown in Table 4.3, it accepts list of concepts from the knowledge base and classifies a given document based on the concept that accepts from the knowledge base.

In order to discriminate between the important and the less important concepts, a weight is assigned to each concept. The determinant that influences a weight given to a concept is the occurrence of a concept based on the number of term frequency. The term frequency indicates how frequent a particular concept is mentioned in the document. The higher the frequency, the more important the concept is considered to be.

Hence, to weight concepts, this study considers concepts, generic concepts and term frequency. There are sets of generic concepts Gc, such as `Accidents`, `Economy`, `Science and Technology`, `Environmental Preservation and Weather Condition`, and `Sport`.

$$Gc = \{Gc_1, Gc_2, Gc_3 \ldots Gc_n\}$$

$$C = \{C_1, C_2, C_3 \ldots C_n\}$$

Where Gc is the set of generic concepts and C is a set of concepts

To determine weight for a concept C_i, the classifier first checks whether or not the concept is a generic concept. If the knowledge base returns a generic concept then the classifier ignores the concept; otherwise it weights the concept. If term T_j belongs to the concept C_i then the weight of concept C_i will be the sum of term frequency of the j^{th} term of a document d, $f(T_j)$.

If $(C_i = Gc_h)$ then

Do nothing

Else if $(T_j \text{ belongs } (C_i))$

$$W_{Ci} = \sum_{j=1}^{n} f(T_j)$$

Based on the above method, the selected document concepts should be weighted to determine the category of the document. For example, from Table 4.3 we have three concepts namely `Sport`, `football` and `Athletics`. Out of the index terms, **ውድድር**, **ቡድን**, **ዋንጫ**, **ስፖርት**, **ክለብ** and **አሸናፊ** belong to the concept `Sport`. Index terms **ኪሎ**, **ሜትር**, **አትሌቲክስ** and **ሩጫ** belong to the concept `Athletics`. The terms **ግብ**, **እግር**, **ቅጣት**, **ምት** and **ማለፍ** belong to the concept `Football`, and **ዳኛ** belongs to both `athletics` and `Football`.

Table 4.4: Sample of concepts with the corresponding weight

No.	Concept	Weight
1	Sport	-
2	Football	12
3	Athletics	12

Table 4.4 shows the list of concepts with their corresponding weight. In order to select a specific concept, this study used maximum weight of the concepts ± some constant *k*, where *k* is decided through the experiment. The reason for inclusion of ± some constant *k* is to categorize the document into multiple categories.

$$Co = Max \{W_{C1}, W_{C2}, W_{C3} \ldots W_{Cn}\}$$
$$Cs = Co \pm k$$

Where Co is the maximum of weighted concept Wc_i and Cs is the selected concept or category of a given document, which is Co ± *k*. Based on the above concept weight process, both concepts `Football` and `Athletics` have the same weight and Co = Max {0, 12, 12} thus the selected concepts are those concepts with values equal to Cs. Therefore, the document is categorized into `Football` and `Athletics` as shown in Figure 4.17.

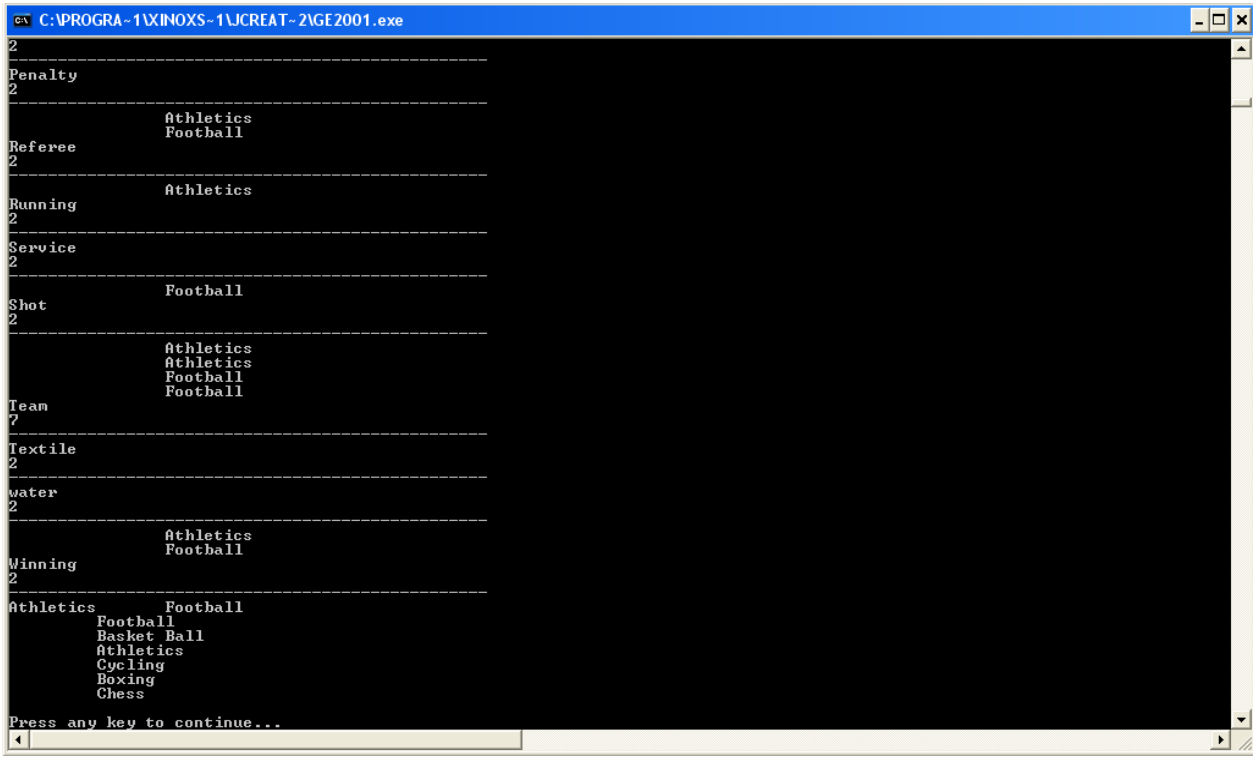

Figure 4.17: The result of the classifier for the sample selected document

4.3.4. Inter Module Communication

Overall, pre-processing, knowledge base and classification modules communicate with each other. The input to the pre-processing module is a flat file which is going to be classified by the system. After the document passes through the pre-process, the result of the pre-processing module is a list of Amharic index terms used as an input for the knowledge base module. Therefore, Amharic index terms should be translated into the corresponding English version.

After the translation is done, the knowledge base module accepts those index terms from the pre-processing module and returns a concept to the classification module. Afterwards, the classification module accepts list of concepts and depending on the weight of the concept, it determines the actual category of a given document. The final output of the classification module is the target category or sub-category.

The Need for Amharic to English Translation

Specific to this study, one of the implementation issues is the interpretation of Amharic terms. At the beginning of the study, the assumption was the working environment that represents the concept, which is TBC, allows Unicode representation. However, the tool is not capable to represent Unicode characters as a concept representation; it only allows Unicode values as a label. During the development, concepts are represented in the ontology using English and the actual document is represented in Amharic. Therefore, there must be some way of filling the gap between the document and the represented knowledge.

On the way, there are three options to resolve the language issue. The first possibility is using labels which show extra information about an item by modifying its label or icon. They can be

used to obtain information about the state of an item without having to look its properties. So it is possible to make actual representation of the item in English and Amharic as labels.

The second option is using turtle, which is a textual language that allows writing down an RDF graph in a compact textual form. Turtle allows encoding Amharic characters using UTF-8. The language has its own syntax and grammar. It consists of a sequence of triple statements or a sequence of (subject, predicate, object) terms, separated by white space and terminated by '.'.

The third possibility is using a simple Amharic to English translator. The translator is a kind of simple table look-up. Accordingly, it contains list of terms or words in Amharic that have meaning equivalent to that of the English terms. It accepts an input term from file and it translates a given Amharic term into its corresponding English version.

The first approach is not feasible because of the limited usage of labels. It is not possible or it is difficult to represent concepts using labels. A concept can be represented using one or more terms and each term may have synonyms. Representing all possible synonyms using labels is also not possible. The second approach needs to represent the whole knowledge stored in the ontology to be translated into RDF textual format. So the process of writing each concept and relationship between concepts in appropriate textual format is a time consuming process. Due to these constraints, using translator is the better way to tackle the language difficulty. The translator serves as an intermediator between the pre-processing and the knowledge base modules.

4.4. Summary

In this chapter we described the basic design criteria and elements of the framework for automatic Amharic document categorization. The main elements of the framework such as the pre-processing, the knowledge base and classification module, have been presented.

During this study, out of various alternatives, the selected methodology has been discussed. The methods that we described illustrate how the knowledge is represented in the ontology. The study used Java as implementation languages and OWL as an ontology language to represent the domain knowledge.

This chapter also presented the main components of implementation including ontology formulation, reasoning and categorization. Sample concepts, relations, instances and rules have also been discussed.

Chapter Five

EXPERIMENT

5.1. Introduction

This chapter presents evaluation of the framework, which is an integral part of the development of any system. The evaluation of the framework is an important part of our work. However, performing this evaluation was not a straightforward activity because there is no standard benchmark or well-defined criteria for evaluating the ontology.

To conduct the experiment we have followed set of procedures which consist of set of activities. In the subsequent pages of this thesis we will discuss the procedures and the results.

5.2. Experimental Procedure

To evaluate the automatic Amharic document categorizer, the following procedures are followed.

5.2.1. Data Collection

As mentioned in Chapter 4, we have considered News documents to develop the knowledge base and demonstrate the automatic Amharic document categorization process. The data source for this study was ENA. The reason why ENA is selected as a data source is the availability of large collection of `News` items which are labeled through an established manual categorization scheme.

The Ethiopian News Agency's news database contains 152,855 Amharic news items starting from October 2001 up to November 2008, however, not all of these news items are useful for the classification process because of errors made during the data entry and there are also documents which are written in English.

ENA uses news items which are classified into 13 categories including a number of sub-categories under each of the categories. The 13 news categories and their sub-categories are shown in Table 5.1.

Table 5.1: News categories used in ENA

No	Category name	Category code	No of sub-categories
1	**አደጋዎች** (Accidents)	**አደዎ**	3
2	**ባህል ጉዳዮች** (Culture)	**ባሕጉ**	8
3	**ኢኮኖሚ** (Economy)	**ኢኮኖ**	11
4	**ትምህርት** (Education)	**ትምህ**	16
5	**የአየር ፀባይ** (Environmental)	**የአፀ**	6
6	**ዓለም አቀፍ ጉዳዮች** (Foreign Affair)	**ዓአጉ**	8
7	**ጤና ጥበቃ** (Health)	**ጤናጥ**	10
8	**ሕግና ፍትህ** (Law and Justice)	**ሕናፍ**	6
9	**ብሔራዊ ፖለቲካ** (Politics)	**ብፖለ**	9
10	**ሳይንስና ቴክኖሎጂ** (Science and Technology)	**ሣንቴ**	4
11	**ማህበራዊ** (Social Affairs)	**ማ·ነበ**	11
12	**ስፖርት** (Sport)	**ስፖት**	5
13	**ሌሎች የፈርጅ ዓይነቶች** (Other Classes)	**ሌፈዓ**	3

5.2.2. Sample Selection

Out of the 152,855 Amharic news items available in ENA database and from the ENA web site, 975 of them were prepared for testing. The data preparation includes the conversion of non Unicode characters into Unicode format. During this study all the 975 Amharic news documents are prepared for the experimentation purpose. The whole collection is in the categories and sub-categories of `Accidents`, `Economy`, `Science and Technology`, `Environmental Preservation and Weather Condition`, and `sport`.

5.2.3. Manual Classification

The next activity is labeling for the experimentation purpose. The 975 Amharic news documents were manually classified into categories and sub-categories. Documents creating main category, following the sub-categories were identified by domain experts. Throughout the whole manual classification process three persons were involved. Moreover, domain experts were also involved in the approval of manually classified categories and sub-categories. The manually classified document helps for checking the final result of the automatic Amharic document categorizer.

5.3. Evaluation

Evaluation of the classifier is done with the evaluation parameter that compares the number of documents which are classified correctly and incorrectly. Typically, the comparison is done amid the document classified using the automatic classifier and that of the manually classified documents.

Precision and recall, which are the evaluation parameters of Information Retrieval, are used in text classification. Precision is the ratio of the number of documents classified correctly to the total number of documents in a given category.

$$P = \frac{TC}{TC+FC}$$

Where, TC denotes the number of documents which are classified correctly and FC denotes the number of documents which are classified incorrectly.

Recall is the ratio of TC and the whole documents belonging to the category,

$R = \frac{TC}{TC+MC}$, Where MC denotes the number of documents which are missed by the classifier, i.e., documents neither classified correctly nor incorrectly.

5.4. Result

The experiments are carried out on the Amharic News documents in the test set. That is, each document in the test set was classified using the above procedure and the result was recorded. Then, the result was compared with the class code assigned manually. The results that are obtained for each category are shown in Tables 5.2 through 5.6.

Table 5.2: Classified documents for sport category

Topic Concept	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Athletics	76	75	0	1	1.0	0.98	98.6
Basketball	1	1	0	0	1.0	1.0	100
Boxing	2	2	0	0	1.0	1.0	100
Chess	6	6	0	0	1.0	1.0	100
Cycling	5	4	1	0	0.8	1.0	80.0
Football	85	84	0	1	1.0	0.98	98.8
Athletics, Football & Cycling	1	0	1	0	0	0	0
Football and Cycling	6	5	1	0	0.83	1.0	83.3
Athletics and Football	5	5	0	0	1.0	1.0	100
Total	187	182	3	2	0.98	0.98	97.3

As depicted in Table 5.5, for `Sport` category from the total of 187 input documents 182 of them are correctly classified. The rest 3 are wrongly classified and 2 of them are missed by the classifier. The total precision value is 0.98 and recall is 0.98, which is 97% accurate.

Table 5.3: Classified documents for Science and Technology category

Topic Concept	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Broadcasting Agency	14	13	0	1	1.0	0.92	92.8
Creativity Work	9	4	1	4	0.8	0.5	44.4
Information & Communication	31	26	2	3	0.92	0.89	83.8
Research and Study	34	31	1	2	0.96	0.93	91.1
Total	88	74	4	10	0.94	0.88	84.0

The above table depicts the experimental result found for `Science and Technology` category. Out of the 88 input documents considered, 74 of them are correctly classified, 4 of them are wrongly classified and 10 of them are missed by the classifier. The total precision value is 0.94, and recall is 0.88 with 84.0% accuracy.

Table 5.4: Classified documents for Environmental category

Topic Concept	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Desertification	2	1	0	1	1.0	1.0	50.0
Environmental Greenery	11	10	0	1	1.0	1.0	90.9
Environmental Pollution	9	5	0	4	1.0	0.5	55.5
Forestry	41	37	0	4	1.0	0.90	90.2
Weather Forecast	12	10	0	2	1.0	0.83	83.3
Wild Animal Protection	1	1	0	0	1.0	1.0	100
Total	76	64	0	12	1.0	0.84	84.2

Table 5.4 illustrates the experimental result of `Environmental` category. The categorizer correctly classified 64 documents and miss-classified 12 documents out of 76 input documents. The precision and the recall of the system for documents that fall into environmental category is 1.0, and 0.84 respectively. This total accuracy of the system is evaluated to be 84.2%.

Table 5.5: Classified documents for Economy category

Topic Concept	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Agriculture & Rural Development	252	250	1	1	0.99	0.99	99.2
Bank and Insurance	42	41	1	0	0.97	1.0	97.6
Donation and Development	67	63	4	0	0.94	1.0	94.0
Industrial Development	14	14	0	0	1.0	1.0	100
Infrastructure Development	45	34	7	4	0.82	0.89	75.5
Investment	46	44	0	2	1.0	0.95	95.6
Mines and Energy	8	8	0	0	1.0	1.0	100
Overall Economy Growth	3	3	0	0	1.0	1.0	100
Trade	72	65	4	3	0.94	0.95	90.2
Water Resource	26	25	1	0	0.96	1.0	96.1
Total	575	547	18	10	0.96	0.98	95.1

As represented in the above table (Table 5.5.) the system classified is for `Economy` category, out of the 575 documents 547 of them are correctly classified with 0.96 precision and 0.98 recall values. The total accuracy is 95.1%.

Table 5.6: Classified documents for Accidents category

Topic Concept	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Disaster Prevention	12	8	0	4	1.0	0.66	66.6
Man Made Disaster	25	24	0	1	1.0	0.96	96.0
Natural Disaster	12	7	0	5	1.0	0.58	58.3
Total	49	39	0	10	1.0	0.79	79.5

As depicted in Table 5.6., the system classified is for `Accident` category, out of the 49 documents 39 of them are correctly classified with 1.0 precision and 0.79 recall values. The total accuracy is 79.5%.

Table 5.7: Classified documents for all categories

Description	No of Input document	TC	FC	MC	P	R	Accuracy Percentage %
Accidents	49	39	0	10	1	0.79	79.5
Economy	575	547	18	10	0.96	0.98	95.1
Environmental Preservation and Weather Condition	76	64	0	12	1.0	0.84	84.2
Science and Technology	88	74	4	10	0.94	0.88	84.0
Sport	187	182	3	2	0.98	0.98	97.3
Total	975	906	25	44	0.97	0.95	92.9

As it is depicted in Table 5.7, it was found that the classifier gave correct result on the average for 92.9% of the test documents. That is, from the 975 documents 906 of them are correctly classified, the rest 69 are classified as incorrect and missed by the classifier.

Table 5.8 depicts documents which are wrongly and missed classified. The result for each document with category the classifier categorizes into and the expected category or sub-category is listed. The reason for this documents are analyzed and discussion in the next section.

Table 5.8: List of wrongly classified document

Category	Document code	Categorized into	Expected Category/Sub-category
Accidents	Acc18	No cat	Manmade disaster
	Acc2	No cat	Disaster prevention
	Acc25	No cat	Disaster prevention
	Acc28	No cat	Disaster prevention
	Acc31	No cat	Disaster prevention
	Acc5	No cat	Natural disaster
	Acc16	No cat	Natural disaster
	Acc27	No cat	Natural disaster
	Acc29	No cat	Natural disaster
	Acc44	No cat	Natural disaster
Economy	Econ248	No cat	Agriculture
	Econ321	Trade	Agriculture
	Econ60	Trade	Donation and Development
	Econ68	Disaster Prevention	Donation and Development
	Econ332	Agriculture	Donation and development
	Econ360	Trade	Donation and development
	Econ38	Donation and Development	Infrastructure Development
	Econ62	Athletics	Infrastructure Development
	Econ69	Water Resource	Infrastructure Development
	Econ92	No cat	Infrastructure Development
	Econ141	Donation and Development	Infrastructure Development
	Econ171	Donation and Development	Infrastructure Development
	Econ179	No cat	Infrastructure Development
	Econ287	No cat	Infrastructure Development
	Econ299	Industrial Development	Infrastructure Development
	Econ341	Trade	Infrastructure development
	Econ399	No cat	Infrastructure development
	Econ213	No cat	Investment
	Econ323	No cat	Investment
	Econ34	Information & communication	Trade
	Econ76	Athletics	Trade
	Econ97	Industrial Development	Trade
	Econ210	No cat	Trade
	Econ219	No cat	Trade

Category	Document code	Categorized into	Expected Category/Sub-category
Economy	Econ223	Overall economic growth	Trade
	Econ339	No cat	Trade
	Econ65	Donation and development	Water resource
	Econ185	Donation and development	Bank and insurance
Environmental Preservation and Weather Condition	En20	No cat	Environmental pollution
	En25	No cat	Environmental pollution
	En34	No cat	Environmental pollution
	En81	No cat	Environmental pollution
	En65	No cat	Desertification
	En14	No cat	Environmental Greenery
	En45	No cat	Forestry
	En48	No cat	Forestry
	En54	No cat	Forestry
	En61	No cat	Forestry
	En9	No cat	Weather Forecast
	En17	No cat	Weather Forecast
Science and Technology	Sc2	No cat	Broad casting agency
	Sc11	No cat	Creativity work
	Sc20	Manmade disaster	Creativity work
	Sc35	No cat	Creativity work
	Sc55	No cat	Creativity work
	Sc62	No cat	Creativity work
	Sc29	No cat	Information & communication
	Sc54	Manmade disaster	Information & communication
	Sc63	No cat	Information & communication
	Sc71	No cat	Information & communication
	Sc83	Manmade disaster	Information & communication
	Sc44	No cat	Research & studies
	Sc60	Natural disaster	Research & studies
	Sc68	No cat	Research & studies
Sport	Sport20	No cat	Football
	Sport32	No cat	Athletics
	Sport52	Football	Football and Cycling
	Sport82	Athletics and Cycle	Cycling
	Sport92	Athletics and Cycle	Athletics, Cycle and Football

5.5. Discussion

Evaluating the quality of domain ontologies is not straightforward. Reusing an ontology for several applications can be a practical method for evaluating domain ontology. Since text categorization is a general tool for information retrieval and knowledge management, we tested the ability of domain ontology to categorize `News` items in this thesis.

Out of the whole collection, 69 of them are classified as incorrect and missed by the classifier. The review showed that they were recorded as incorrectly classified because of the selected index term frequency, the constant value *k* and the knowledge represented in the knowledge base.

For the wrongly classified documents, representative terms are within the index terms which have the frequency value of one. So, as it is discussed in Chapter 4, this study only considers index term greater than or equal to two. Due to this reason the system wrongly classified the documents. For example document "`Sport52`" and "`Sport92`" are wrongly classified because of the index terms frequency. Index terms which are taken from these documents are not representative terms to discriminate the documents. In the case of misclassified documents, for each document the index terms frequency is one; there is no index term which is greater than or equal to two, so such documents are not capable to be classified by the system. For example, documents "`Sport20`" and "`Sport32`" are classified as missed, because all of the index terms for both documents are one.

The other reason for wrongly classifying documents is the constant *k* that is used to calculate weight of a concept. Out of the entire collection, a single document called "`Sport82`" contains index terms which belong to both concepts `Athletics` and `Cycling`. The expected category for this document is `Athletics`. However, the weight of concept `Athletics` calculated from the weight of `Cycling` plus or minus constant *k*. So the concept `Athletics` contains index terms

that are common for both concepts. Therefore, because of the constant value of *k* the weight for both documents becomes equal and classified as both `Cycling and Athletics`.

In addition to the above reason, the result primarily depends on the knowledge represented in the ontology. From the wrongly classified documents, most of the concepts contained in the documents are similar with each other. However, index terms which are extracted from the documents are included in some other concepts. As a result, the reasoner tries to find related concepts and map those index terms onto the related concepts but not the exact concept. For example, document "`Econ332`" is wrongly classified, because the document contains index terms such as "`Farmer`". In the knowledge base, the concept "`Agriculture`" contains an index term "`Farmer`". Hence, the reasoner maps it into the concept "`Agriculture`", which is the wrong category. Due to these reasons, the categorizer incorrectly classifies the above documents.

On the other hand, it can be seen that larger number of `News` items are classified correctly. This shows that category concepts for correctly classified documents are plainly represented in the knowledge base that distinguishes it from other categories.

Therefore, it is apparent that the classification process is primarily governed by the represented knowledge in the ontology. In general, having a complete and clearly stated knowledge for each category and sub-category decides the final result of the categorization process. And this shows that as the knowledge base gets richer, the performance of the system will be enhanced considerably.

During this study, for five categories including sub-categories the result found from the experiment is 92.9%, which shows the accuracy of the classifier is good. However, it is a promising way to get more accurate results closer to 100.

Chapter Six

CONCLUSION AND RECOMMENDATIONS

6.1. Conclusion

With the advancement of technology, the focus shifted more towards implementing algorithms and designing computer programs to automating systems for storing and retrieving information. The representation, storage and organization of information should provide the user with easy access to the information in which s/he is interested. The focus of information retrieval is to support the user to get relevant information according to the requirement. To facilitate the use of information storage and access it is important to organize documents, thereby automatically putting similar documents together based on their contents to a number of predefined categories.

As a major work in information storage and retrieval, various researches are carried out in the area of automating classification. The results of these research works have showed that classification can be automated and good results could be obtained. The results also showed that the use of automatic classification techniques for Amharic documents is achievable and very promising.

This research work has attempted to look into the techniques of automatic classification. Throughout the study, to categorize a given document into a predefined category, the document passes through the pre-processing and classification processes. Pre-processing involves lexical analysis, normalization, stop-word removal, stemming and index terms selection. Classification involves the process of querying, getting access to the knowledge base, and categorizing the documents based on the concepts.

In order to categorize a given document, the knowledge that contains concepts in the `News` domain is represented using ontologies. After the representation of domain concepts, document

representative terms are extracted from the document as index terms. Using the index terms that are extracted from the document and the knowledge base, a given document is classified into predefined categories.

The specific tasks undertaken to meet the objective and the result obtained are:

- Identifying the process of constructing the knowledge base using ontology.
- In the presented research work, we concentrated on the knowledge base construction process via having domain specific ontology, which is the `News` ontology.
- An initial prototype for automatic Amharic document categorizer is developed.
- Testing the developed automatic Amharic document categorization system with the News collection of Amharic documents.
- Categorizing a given document into a single-label and multi-label classification is achieved through this process.
- Hierarchical documents categorization is also done, that deals with problems where categories are organized in the form of a hierarchy.

As the result of these processes, regarding the techniques of automatic classification, using concepts for Amharic document categorization obtained a good result, i.e. 92.9% of the test collection documents are correctly classified. However, the technique needs to be supported with extended knowledge and preprocessing techniques, like concept extraction from the document instead of using index terms, which have a great contribution in the pre-processing.

6.2. Contribution of the study

Some of the main contributions of the study are listed below:

- A generic model is proposed for concept-based automatic Amharic text categorization that takes advantage of existing semantic technology.
- The knowledge base that is domain specific to `News` items, i.e., the ontology, could be used in the Semantic Web, or to correlate with other Semantic applications.
- The study tries to show the possibility of achieving Single-label and Multi-label classification, and Hierarchical documents categorization which are not considered in the previous works.
- In addition, the study contributes to the growth of semantics technology as well as to text categorization. Concept-based automatic Amharic text categorizer paves the way for text categorization with semantic technologies.

6.3. Recommendations

The results found in this research showed that classification can be done automatically for Amharic documents using concepts. However, it is also learnt that further research and developmental effort is needed so as to enable the complete exploitation of this technology.

The ontology development and deployment environment is rich with ideas that could further improve the process of knowledge based systems construction. In this section, a number of such ideas are listed. Those ideas deal with future research issues and some features that are not of a research nature but that are needed to provide a better result.

- Concept extraction: during this study we used a list of single terms as document representative words. However, it would be better to use sequence of terms by identifying relationships between words in the text using lexical database and identifying groups of words which form closely tied conceptual groups.
- Extending into multi lingual documents: basically, the main characteristics of the ontology are being sharable and reusable. To enable knowledge sharing and reuse, it is necessary to represent concepts and relations in multiple languages. Instead of automatically categorizing only Amharic documents, it is possible to incorporate other languages such as English and to make it multi lingual.
- Extending the knowledge base: this study only represents the knowledge in the area of `Accidents`, `Economy`, `Science and Technology`, `Environmental Preservation and Weather Condition`, and `sport`. All possible knowledge related to the `News` domain should be represented in the knowledge base which needs exhaustive representation of concepts in the `News` domain.
- External Knowledge: In the process of extracting concepts from the knowledge base, index terms are mapped on the corresponding concepts of the ontology. However, there is a possibility that the term may not exist because of the limited number of concepts available in the `News` ontology. This situation requires an alternative way of mapping onto the external knowledge base concept. The alternative way is to use the extended concept in order to map between the external concept and the existing knowledge base. Having such external knowledge makes the existing knowledge base more powerful to incorporate possible concepts.

- Amharic lexical database: having Amharic lexical database that offers information related to various semantic relationships among words is essential. It has a potential to be used in order to represent concepts rather than words and the relationships between concepts including the corresponding synonyms and antonyms. Having the feature representations with this database information is believed to result in a significant improvement of the classification process.

References

[1] T.D. Wilson, “Human Information Behavior”, volume 3 No 2,2000

[2] Wiley John, ”Towards The Semantic Web - Ontology-driven Knowledge Management”, 2002

[3] Brussels, “Ontologies - Introduction and Overview”, Unpublished MSc Thesis Vrije Universiteit Brussel, 2004

[4] Salton, G. and Micheal J. McGill. “Introduction to Modern Information Retrieval”, New York: McGraw-Hill Book Company, 1983.

[5] Baeze-Yates & Riberio-Neto, ”Modern Information Retrieval”, 1990

[6] C. J. van RIJSBERGEN B.Sc., “ Information Retrieval”

[7] Jacob Lumbroso ,”Basic Facts about the Amharic Language”, October 07, 2007

[8] Peter Willett, “Recent trends in hierarchical document clustering”: A critical review. Information Processing & Management, 24(5):577-597, 1988. Available from http://www2.parc.com/istl/projects/ia/sg-clustering.html, last accessed on January 19, 2009

[9] E. Rasmussen,”Clustering Algorithms”, in Information Retrieval Data Structures and Algorithms, William Frakes and Ricardo Baeza-Yates, editors, Prentice Hall, 1992 Available from http://www.dcs.gla.ac.uk/~iain/keith/data/pages/103.htm, last accessed on October, 16

[10] Surafel Teklu. “Automatic categorization of Amharic news document”: A machine learning Approach, Master thesis, Addis Ababa University, 2003.

[11] Barry Smith, “Beyond Concepts: Ontology as Reality Representation”

[12] Deborah Nichols and Allan Terry ,“User's Guide to Teknowledge Ontologies” Teknowledge Corp. December 3, 2003

[13] Tessema Mindaye, “DESIGN AND IMPLEMENTATION OF AMHARIC SEARCH ENGINE”, Master thesis Addis Ababa University, July 2007

[14] Zelalem Sintayehu, “Automatic classification Amharic news items”: the case of Ethiopian News Agency, Master thesis Addis Ababa University, 2001.

[15] Yohannes Afework, “Automatic Amharic Document Categorization”: the case of Ethiopian News Agency Master thesis Addis Ababa University, 2007

[16] http://www.alsintl.com/resources/languages/Amharic/, “Amharic”, last accessed on January 18,2009

[17] Grigoris Antoniou1 and Frank van Harmelen, “Web Ontology Language: OWL”, Department of Computer Science, University of Crete, ga@csd.uoc.gr, Department of AI, Vrije Universiteit Amsterdam

[18] “Understanding Knowledge Societies” In twenty questions and answers with the Index of Knowledge Societies, United Nations New York, 2005

[19] Fabrizio Sebastiani ,“Text Categorization”,Dipartimento di Matematica Pura e Applicata, Universit`a di Padova, 35131 Padova, Italy

[20] Wang, Y., Zang, H., Spencer, B., and Yan, Y. “A Text Categorization Approach for Match-Making in Online Business Tendering “, October 2005

[21] Nega Alemayehu and Peter Willett, “Stemming of Amharic Words for Information Retrieval”, University of Sheffield, Sheffield, UK

[22] Surafel Teklu. “Automatic categorization of Amharic news document”: A machine learning Approach, Master thesis, Addis Ababa University, 2003.

[23] Fuchun Peng, Xiangji Huang, Dale Schuurmans, Shaojun Wang ,”Text Classification in Asian Languages without Word Segmentation”

[24] Mohamed EL KOURDI, Amine BENSAID, Tajje-eddine RACHIDI, “Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm”

[25] Hele-Mai Haav and Tanel-Lauri Lubi, “A Survey of Concept-based Information Retrieval Tools on the Web”

[26] Rifat Ozcan, Y. Alp Aslandogan, “Concept Based Information Access Using Ontologies and Latent Semantic Analysis”, Technical Report CSE-2004-8

[27] Gongde Guo,Hui Wang, David Bell, Yaxin Bi and Kieran Greer, “An KNN Model-based Aprocah and Its Application in Text Categorization”

[28] Sabrina Tiun, Rosni Abdullah, Tang Enya Kong, “Automatic Topic Identification Using Ontology Hierarchy”

[29] Shih-Hung Wu, Tzong-Han Tsai, Wen-Lian Hsu, “Text Categorization Using Automatically Acquired Domain Ontology”

[30] Randall Davis, Howard Shrobe, and Peter Szolovits, “What Is a Knowledge Representation?”

[31] R. Davis, H. Shrobe, and P. Szolovits, What is a Knowledge Representation? AI Magazine, 14(1):17-33, 1993

[32] Asuncion Gomez-Perez, Mariano Fernandez and Oscar Corcho, “Ontological Engineering”,2004

[33] John Wiley ,“Semantic Web Technologies”, Copyright 2006 John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, England Telephone (þ44) 1243 779777

[34] Brussels, “Ontologies - Introduction and Overview”, Unpublished MSc Thesis Vrije Universiteit Brussel, 2004

[35] Louisa Casely-Hayford, “A comparative analysis of methodologies, tools and languages used for building Ontologies”, CCLRC Daresbury Laboratories

[36] M. Lynne Murphy, “Semantic Relations and Lexicon” ,University of Sussex

[37] Fekade Getahun, Solomon Atnafu, “The Use of Semantic-based Predicates Implication to Improve Horizontal Multimedia Database Fragmentation”, Department of Computer Science, Faculty of Informatics, Addis Ababa University, 1176 Addis Ababa, Ethiopia

[38] Berners-Lee, G¨odel, and Turing, “Thinking on the Web”, Copyright © 2006 John Wiley & Sons, Inc.

[39] http://pellet.owldl.com. ,last accessed on January 19, 2009

[40] Evren Sirin and Bijan Parsia, “Pellet: An OWL DL Reasoner”, MINDSWAP Research Group, University of Maryland, College Park, MD

[41] http://jena.sourceforge.net, last accessed on September 08,2008

[42] http://www.ontotext.com/owlim/, last accessed on September 08,2008

[43] Julio C. Arpírez, Oscar Corcho, Mariano Fernández-López, Asunción Gómez-Pérez, “WebDE: a Scalable Workbench for Ontological Engineering”

[44] Grigoris Antoniou and Frank van Harmelen, “A Semantic Web Primer”, The MIT Press Cambridge, Massachusetts London, England 2004

[45] Sean Bechhofer, Ian Horrocks, Carole Goble and Robert Stevens “OilEd: a Reason-able Ontology Editor for the Semantic Web”

[46] York Sure, Juergen Angele, and Steffen Staab ,” OntoEdit: Guiding Ontology Development by Methodology and Inferencing ”

[47] Deborah L. McGuinness, Richard Fikes, James Rice, and Steve Wilder, “The Chimaera

Ontology Environment”

[48] Matthew Horridge, Holger Knublauch, Alan Rector, Robert Stevens, Chris Wroe, “A Practical Guide To Building OWL Ontologies Using The Prot´eg´e-OWL Plugin and CO-ODE Tools” http://protege.stanford.edu/ “ ”,last accessed on January 19, 2009

[49] Biplab K. Sarker, Peter Wallace and Will Gill, “Some Observations on Mind Map and Ontology Building Tools for Knowledge Management”, Research & Development, Innovatia Inc., Saint John E2L 4R5, Canada

[50] Fernández López, M., “Overview Of Methodologies For Building Ontologies”

[51] Fuchun Peng, Xiangji Huang, Dale Schuurmans, Shaojun Wang ,”Text Classification in Asian Languages without Word Segmentation”

[52] Sisay Fissaha and Johann Haller “Application of corpus-based techniques to Amharic texts“

[53] Daniel Yacob, “Developments towards an electronic Amharic corpus”, 2005

[54] Yonas Hailu, “Ethiopic Online Handwriting Recognition System Using Simplified Ethiopic Script”, Master thesis Addis Ababa University, 2007

[54] http://www.ontotext.com/owlim/. last accessed on January 19, 2009

[55] Atelach Alemu Argaw, Lars Asker, Rickard Cöster and Jussi Karlgren. "Dictionary-based Amharic - English Information Retrieval". In Proceedings of Cross Language Evaluation Forum (CLEF 2004), Bath, UK. September 2004.

[56] Seongwook Youn, Dennis McLeod ,“Spam Email Classification using an Adaptive Ontology”,Department of Computer Science, University of Southern California, Los Angeles, CA. USA

[57] Solomon Nega,”Analysis of Semantic Technologies for Ethiopic Manuscripts, Art and Music”, 2007

[58] Semahene Ayalew, “Automatic Clustering of Amharic News Items”, Oct 2007

[59] Erik Hatcher and Otis Gospodnetic, “Lucene in Action”, Manning Publications Co, 2005.

[60] Pressman, R.S.,“Software Engineering, A Practitioner’s Approac”,. 4th Edition European Adaptation by D. Ince, McGraw-Hill.,1997

[61] http://www.nl.edu/library/Tutorials/organizationofinformation.cfm,“information organization”, last accessed on September 8, 2008

[62] Michael C. Daconta, Leo J. Obrst, Kevin T. Smith, “The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management”, Copyright © 2003 by Michael C. Daconta, Leo J. Obrst, and Kevin T. Smith.

[63] Thomas R. Gruber, “ Toward Principles for the Design of Ontologies Used for Knowledge Sharing”, August 23, 1993

[64] Huazhen Gu1, Kuanjiu Zho, “Text Classification Based on Domain Ontology”, Journal of Communication and Computer, ISSN1548-7709, USA, May 2006, Volume 3, No.5

Annexes

Annex A : Classes

- news:ability
- news:Accident_SubCategory
- news:Accidents
- news:Activity
- news:Agency
- news:Agriculture_and_Rural_Development
- news:Antiques
- news:Area
- news:Art
- news:Athlete
- news:Athletics
- news:AthleticsRule
- news:AthleticsTeam
- news:Ballgame
- news:Bank_and_Insurance
- news:Beachball
- news:Birth_and_Death
- news:Body
- news:Boxing
- news:Broadcasting_Agency
- news:Card
- news:Category
- news:Children_and_Mothers_Health
- news:Citizenship_and_Immigrant
- news:Coach
- news:Competitor
- news:Constitutiona_Affairs
- news:Content Type
- news:Contestant
- news:Corruption
- news:Creativity_Work
- news:Crime_Affairs
- news:Cultural_Treatment
- news:Culture
- news:Culture_SubCategory
- news:Democracy_and_Good_Governor
- news:Game
- news:Genocide
- news:Desertification
- news:Diplomatic_Relation
- news:Disaster_Prevention
- news:Discussion_Rules_and_Regulations
- news:Disease_Protection
- news:Distance_Learning
- news:Donation_and_Development
- news:Duration
- news:Economy
- news:Economy_SubCategory
- news:Edroch
- news:Education
- news:Education_SubCategory
- news:Educational_Coverage
- news:Educational_Institution
- news:Elderies
- news:Election
- news:Elementary
- news:Employee_and_Emplacement
- news:Environmental_Greenery
- news:Environmental_Pollution
- news:Environmental_Preservation_and_Weather_Condition
- news:Environmental_SubCategory
- news:Expected
- news:Federation_and_Clubs
- news:Field
- news:Flag
- news:Football
- news:Football1
- news:FootballRule
- news:FootballTeam
- news:Foreign_Conflicts
- news:Foreign_Relations
- news:Foreign_SubCategory
- news:Forestry
- news:Free_Education
- news:Military_Training
- news:Mines_and_Energy

- news:Goal
- news:Hand
- news:Health
- news:Health_Institution_Construction
- news:Health_Professionals
- news:Health_Service
- news:Health_SubCategory
- news:Health_Tools
- news:Higher_Education
- news:History
- news:Human_Donation
- news:Humanitarian_and_Democracy
- news:Industrial_Development
- news:Information_and_Communication
- news:Infrastructure_Development
- news:International Agency
- news:International_Politics
- news:Investment
- news:Judiciary_Bodies
- news:Justice_Affairs
- news:Kindergarten
- news:Law_and_Justice
- news:Law_SubCategory
- news:League
- news:Learning_Methods
- news:Learning_Tools
- news:Linesman
- news:Local Agency
- news:Magazine
- news:Magazine Article
- news:Man_made_Disaster
- news:Match
- news:Mediceanes_and_Narcotic_Drug
- news:Medium
- news:Micro_Enterprise
- news:Military_Mission
- news:Runner
- news:Running
- news:Science_and_Technology
- news:ScienceTech_SubCategory
- news:Secondary
- news:Secondary_School
- news:Sex_Discipline
- news:Nation_and_Nationality
- news:Nation_Stability
- news:National_Politics
- news:Natural_Disaster
- news:News
- news:Newspaper
- news:Newspaper Article
- news:Not-Regular_Education
- news:Opponent
- news:Other_Classes
- news:Other_Diseases
- news:Other_Modern_Sport
- news:Others_SubCategory
- news:Overall_Economy_Growth
- news:Participator
- news:Peace_and_Stability
- news:Physical_Disability
- news:Play
- news:Player
- news:PlayingArea
- news:PlayingField
- news:Political_Application
- news:Political_Partis
- news:Politics
- news:Politics_SubCategory
- news:Professional
- news:Professional_and_Social_Assistance
- news:Radio
- news:Radio Program
- news:Referee
- news:Regular
- news:Religious_and_National_Holiday
- news:Religious_Assembly
- news:Research_and_Study
- news:Rugbyball
- news:Rule
- news:Television
- news:Television Program
- news:Terrorism
- news:Test
- news:Tourists
- news:Track
- news:Trade

- news:Soccer
- news:Social_SubCategory
- news:SocialAffairs
- news:Sport
- news:Sports_SubCategory
- news:Stadium
- news:SubCategory
- news:Synonym
- news:Synonym_Sport
- news:TB_and_HIV_AIDS
- news:Teachers_and_Students_Affair
- news:Teammember
- news:Technical_Education
- news:Traditional_Sport
- news:Truism_Development
- news:Urgent
- news:Water_Resource
- news:Weather_Forcast
- news:Web Article
- news:Website
- news:Wedding_and_Divorce
- news:Wild_Animal_Protection
- news:Woman_and_Education
- news:Women_Affair
- news:World_Wide_Continent_Activities
- news:Youth_and_Young_Affairs

Annex B: Data type Properties

- news:description
- news:editor
- news:name
- news:reporter
- news:title
- news:Website

Annex C: Object Properties

- news:ComponentOf
- news:has
- news:Has Keyword Group
- news:Has Medium
- news:hasAgency
- news:hasCategory
- news:hasPart
- news:hasProfessional
- news:hasRule
- news:hasSubCategory
- news:hasTeam
- news:IsA
- news:KindOf
- news:PartOf
- news:source
- news:Use
- owl:differentFrom
- owl:disjointWith

owl:equivalentClass

owl:equivalentProperty

owl:inverseOf

owl:sameAs

Annex D: Individuals

news:Addis Admas

news:Additionaltime

news:Adere

news:Angle

news:area

news:Arm

news:Attack

news:Ball

news:BBC-News

news:Berhane

news:Break

news:Broad

news:Bronze

news:challenger

news:Champion

news:championship

news:Championship

news:Club

news:Competition

news:Confederation

news:Corner

news:Cornerkick

news:Crisscross

news:Cross

news:Cross-Country

news:Derartu

news:Dribble

news:ENA

news:ENA Website

news:Endlines

news:E Reporter Amharic Website

news:Ethiopian Reporter English Website

news:Extratime

news:FACap

news:Federation

news:Feet

news:field

news:Final

news:Find Keyword Group

news:Flag1

news:Flag2

news:Foot

news:Forfeit

news:Foul

news:FreeKick

news:Gebrselassie

news:GoalKick

news:Gold

news:Group

news:Haile

news:Half

news:Head

news:ITAR-TASS

news:Janmeda

news:Kenenisa

news:Kilometer

news:KirayBete

news:Maraton

news:Medal

news:Medalist

news:Menilik

news:Meter

news:Mile

news:Moged

news:Money.CNN.Com

news:Net

news:Newcastle

news:Offense

news:Offside

news:Pass

- news:Penality
- news:PenalityArea
- news:Point
- news:Police
- news:Primerleague
- news:Prize
- news:Record
- news:Reporter-Amharic
- news:Reporter-English
- news:Scorer
- news:ShortDistance
- news:Shot
- news:Sidelines
- news:Silver
- news:Speed
- news:Sports Illustrated.Cnn.Com
- news:Score
- news:Strike
- news:Striker
- news:score
- news:Result
- news:Reuters news
- news:Round
- news:Thrill
- news:Tobia
- news:Trapping
- news:Trophy
- news:Tulu
- news:Walta
- news:Walta Information Center
- news:Winner
- news:WorldCup
- news:Www.Addis Admas.Com
- news:WWW.CNN.Com

Annex E: Sample document

በተለያዩ አካባቢዎች የተለያዩ ስፖርቶች ተካሒዱ ስፖርት በድሬዳዋ እስታዲየም ትናንት በተካሔደው የብሔራዊ ሊግ የእግር ኳስ ውድድር ድሬዳዋ ጨርቃጨርቅ ቡድን የደብረብርሀን ብርድልብስ ቡደንን 1 ለ0 አሸነፈ የሁለቱ ቡድኖች ጨዋታ በመጀመሪያው 45 ደቂቃ በአብዛኛው በመሐልሜዳ የተወሰነ ቢሆንም ከእረፍት መልስ በሁለቱም ቡድኖች ላይ የድካም ስሜት ተንጸባርቋል የድሬዳዋ ጨርቃጨርቅ አጥቂ በግብ ክልል በተፈጸመበት ጥፋት የፍጹም ቅጣት ምት በማግኘቱ በአራት ቁጥሩ አህመድ እድሪስ ብቸኛውን ግብ በማግባት አሸናፊ ሆኖአል የደብረብርሀን ብርድልብስ ፋብሪካ ተጫዋቾች የተሰጠው ቅጣት ምት አግባብነት የለውም በማለት ከዳኛው ጋር ሲከራከሩ የታዩ ሲሆን በተለይ አስራ አምስት ቁጥሩ ዳኛውን በመዝለፉ በቀይ ካርድ ከሜዳ ወጥቷል በሌላም በኩል በድሬዳዋ አስተዳደር ካውንስል በሰባት ቡድኖች መካከል ሲካሔድ የሰነበተው የጥሎ ማለፍ ውድድር በዋበሪ ቡድን አሸናፊነት ከትናንት በስቲያ ተጠናቋል እንዲሁም በባህርዳር ከተማ በአምስት ክለቦች መካከል ለአንድ ሳምንት ሲካሔድ የሰነበተው የባህርዳር ልዩ ዞን የክለቦች እግር ኳስ የጥሎ ማለፍ የዋንጫ ውድድር ትናንት በውሀ አገልግሎት ቡድን አሸናፊነት ተጠናቋል ለዋንጫ ሽሚያ በተደረገው ውድድር የውሀ አገልግሎት የኦሳን ቡድን 1 ለዜሮ በማሸነፉ የዋንጫ ባለቤት ሆኖአል በተመሳሳይ ሁኔታ በደቡብ ጎንደር ዞን በደራ ወረዳ አስተናጋጅነት በሐሙሲት ከተማ የአትሌቲክስ ውድድር መካሔዱን የዞኑ ወጣቶች ባህልና ስፖርት ጉዳይ ጽህፈት ቤት አስታውቋል በጽህፈት ቤቱ የውድድር ፕሮግራሞች ቁጥጥርና ክትትል አስተባባሪ አቶ ኮከብ እንዳየሁ እንደገለጹት በስድስት በስምንትና በአስራ ሁለት ኪሎ ሜትር በተካሔደው የጎዳና ላይ የሩጫ ውድድር ከሰባት ወረዳዎች የተውጣጡ 53 ወንድና ሴት ስፖርተኞች ተሳትፈውበታል የውድድሩ አላማ በዞኑ ባሉት ወረዳዎች ውስጥ የአትሌቲክስ ስፖርት እንዲዘወተርና በደብረብርሀን ከተማ በሚካሔደው ክልል አቀፍ የሩጫ ውድድር ዞኑን ወክለው የሚሳተፉ ስፖርተኞችን ለመምረጥ ታስቦ መሆኑን አመልክተዋል በ8 እና 12 ኪሎ ሜትር ወጣትና አዋቂ ወንዶች የደብረታቦር ከተማ ወረዳ እንዲሁም በ6 እና 8 ኪሎ ሜትር ሴቶች የእብናት ወረዳ አሸናፊ በመሆን የተዘጋጀላቸውን ዋንጫ ወስደዋል

Annex F: Rules

```
#Rule file
@prefix news: <http://cs.aau.edu.et/ontologies/News.owl#>
@prefix owl: <http://www.w3.org/2002/07/owl#>
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
# ======================================
# Content: News Ontology Text Rules
# ======================================
# ======================================
#              Football
# ======================================
[FootballTeam:
       (news:Football rdfs:subClassOf news:Sport)
       (news:FootballTeam rdfs:subClassOf news:Team)
->     (news:FootballTeam rdfs:subClassOf news:Football)
]
[FRule1:
       (news:Player rdfs:subClassOf news:Professional)
->     (news:opponent rdfs:sameAs news:Player)
]
[FRule2:
       (?x rdf:type news:Play)
       (news:Play rdfs:subClassOf news:Activity)
->     (?x rdfs:subClassOf news:Football)
]
[FRule3:
       (?x rdfs:subClassOf news:Activity)
       (news:Play rdfs:subClassOf news:Activity)
->     (news:Play rdfs:subClassOf news:Football)
]
[FRule4:
       (?x rdf:type news:Skill)
       (news:Skill rdfs:subClassOf news:Player)
->     (?x rdfs:subClassOf news:Football)
]
[FRule5:
       (?x rdf:type news:Skill)
       (news:Skill rdfs:subClassOf news:Player)
->     (news:Player rdfs:subClassOf news:Football)
]
[FRule6:
       (?x rdf:type news:Field)
       (news:Field rdfs:subClassOf news:Stadium)
->     (?x rdfs:subClassOf news:Football)
]
[FRule7:
       (?x rdf:type news:FootballTeam)
       (news:FootballTeam rdfs:subClassOf news:Team)
->     (?x rdfs:subClassOf news:Football)
]
[ARule8:
       ( ?x rdf:type news:Hand)
       ( news:Hand rdfs:subClassOf news:FootballRule)
->     ( ?x rdfs:subClassOf news:Football)
]
[ARule9:
       ( ?x rdf:type news:Duration)
       ( news:Duration rdfs:subClassOf news:FootballRule)
->     ( ?x rdfs:subClassOf news:Football)
]
[ARule10:
       ( ?x rdf:type news:Body)
       ( news:Body rdfs:subClassOf news:FootballRule)
->     ( ?x rdfs:subClassOf news:Football)
]
[ARule11:
       (?x rdf:type news:Football)
       (news:Football rdfs:subClassOf news:Sport)
```

```
->     (?x rdfs:subClassOf news:Football)
]
[ARule12:
       (news:Player rdfs:subClassOf news:Professional)
->     (news:Player rdfs:subClassOf news:Football)
]
# ======================================
#              Athletics
# ======================================
[AthleticsTeam:
       (news:Athletics rdfs:subClassOf news:Sport)
       (news:AthleticsTeam rdfs:subClassOf news:Team)
->     (news:AthleticsTeam rdfs:subClassOf news:Athletics)
]
[ARule1:
       (?x rdf:type news:Athlete)
       (news:Athlete rdfs:subClassOf news:Professional)
->     (?x rdfs:subClassOf news:Athletics)
]
[ARule2:
       (?x rdf:type news:AthleticsTeam)
       (news:AthleticsTeam rdfs:subClassOf news:Team)
->     (?x rdfs:subClassOf news:Athletics)
]
[ARule3:
       (?x rdf:type news:Athletics)
       (news:Athletics rdfs:subClassOf news:Sport)
->     (?x rdfs:subClassOf news:Athletics)
]
[ARule4:
       (news:Athlete rdfs:subClassOf news:Professional)
->     (news:Athlete rdfs:subClassOf news:Athletics)
]
# ======================================
#              Boxing
# ======================================
[BRule1:
       (?x rdf:type news:Boxing)
       (news:Boxing rdfs:subClassOf news:Activity)
->     (?x rdfs:subClassOf news:Box)
]
[BRule2:
       ( ?x rdf:type news:Boxing )
       ( news:Boxing rdfs:subClassOf news:Sport)
->     ( ?x rdfs:subClassOf news:Box)
]
# ======================================
#              Cycling
# ======================================
[CRule1:
       (?x rdf:type news:BicycleTypes)
       (news:BicycleTypes rdfs:subClassOf news:Bicycle)
->     (?x rdfs:subClassOf news:Cycling)
]
[CRule2:
       (?x rdf:type news:BicycleComponents)
       (news:BicycleComponents rdfs:subClassOf news:Bicycle)
->     (?x rdfs:subClassOf news:Cycling)
]
[CRule3:
       (?x rdf:type news:Cycling)
       (news:Cycling rdfs:subClassOf news:Sport)
->     (?x rdfs:subClassOf news:Cycling)
]
[CRule4:
       (?x rdf:type news:CyclingTeam)
       (news:CyclingTeam rdfs:subClassOf news:Team)
->     (?x rdfs:subClassOf news:Cycling)
]
[CRule5:
       (?x rdf:type news:Bicycle)
```

```
       (news:Bicycle rdfs:subClassOf  news:Cycling)
->     (?x rdfs:subClassOf news:Cycling)
]
[CRule6:
       (news:Cyclist rdfs:subClassOf news:Professional)
->     (news:Cyclist rdfs:subClassOf news:Cycling)
]
# ======================================
#              Chess
# ======================================
[ChRule1:
       (?x rdf:type news:ChessType)
       (news:ChessType  rdfs:subClassOf  news:Chess)
->     (?x rdfs:subClassOf news:Chess)
]
[ChRule2:
       (?x rdf:type news:ChessPart)
       (news:ChessPart  rdfs:subClassOf news:Chess)
->     (?x rdfs:subClassOf news:Chess)
]
[ChRule3:
       (?x rdf:type news:Chess)
       (news:Chess  rdfs:subClassOf news:Sport)
->     (?x rdfs:subClassOf news:Chess)
]
# ======================================
#              Economy
# ======================================
[EcRule1:
       (?x rdf:type news:IndustrialDevelopment)
       (news:IndustrialDevelopment  rdfs:subClassOf  news:Economy)
->     (?x rdfs:subClassOf news:IndustrialDevelopment)
]
[EcRule2:
       (?x rdf:type news:GoodsManufacturers)
       (news:GoodsManufacturers rdfs:subClassOf news:IndustrialDevelopment)
->     (?x rdfs:subClassOf news:IndustrialDevelopment)
]
[EcRule3:
       (?x rdf:type news:ServiceProvider)
       (news:ServiceProvider  rdfs:subClassOf  news:IndustrialDevelopment)
->     (?x rdfs:subClassOf news:IndustrialDevelopment)
]
[EcRule4:
       (?x rdf:type news:InfrastructureDevelopment)
       (news:InfrastructureDevelopment  rdfs:subClassOf  news:Economy)
->     (?x rdfs:subClassOf news:InfrastructureDevelopment)
]
[EcRule5:
       (?x rdf:type news:Bond)
       (news:Bond  rdfs:subClassOf  news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule6:
       (?x rdf:type news:Stock)
       (news:Stock  rdfs:subClassOf  news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule7:
       (?x rdf:type news:OthersInvestment)
       (news:OthersInvestment rdfs:subClassOf  news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule8:
       (?x rdf:type news:RuralDevelopment)
       (news:RuralDevelopment rdfs:subClassOf
news:AgricultureandRuralDevelopment)
->     (?x rdfs:subClassOf news:AgricultureandRuralDevelopment)
]
[EcRule9:
       (?x rdf:type news:OrganicAgriculture)
```

```
       (news:OrganicAgriculture rdfs:subClassOf news:Agriculture)
->     (?x rdfs:subClassOf news:AgricultureandRuralDevelopment)
]
[EcRule10:
       (?x rdf:type news:IndustrialAgriculture)
       (news:IndustrialAgriculture rdfs:subClassOf news:Agriculture)
->     (?x rdfs:subClassOf news:AgricultureandRuralDevelopment)
]
[EcRule11:
       (?x rdf:type news:Agriculture)
       (news:Agriculture rdfs:subClassOf news:AgricultureandRuralDevelopment)
->     (?x rdfs:subClassOf news:AgricultureandRuralDevelopment)
]

[EcRule12:
       (?x rdf:type news:InsuranceIndustry)
       (news:InsuranceIndustry rdfs:subClassOf news:BankandInsurance)
->     (?x rdfs:subClassOf news:BankandInsurance)
]
[EcRule13:
       (?x rdf:type news:Commercial)
       (news:Commercial rdfs:subClassOf news:BankingIndustry)
->     (?x rdfs:subClassOf news:BankandInsurance)
]
[EcRule14:
       (?x rdf:type news:Central)
       (news:Central rdfs:subClassOf news:BankingIndustry)
->     (?x rdfs:subClassOf news:BankandInsurance)
]
[EcRule15:
       (?x rdf:type news:BankingIndustry)
       (news:BankingIndustry rdfs:subClassOf news:BankandInsurance)
->     (?x rdfs:subClassOf news:BankandInsurance)
]
[EcRule16:
       (?x rdf:type news:BankandInsurance)
       (news:BankandInsurance rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:BankandInsurance)
]
[EcRule17:
       (?x rdf:type news:Projects)
       (news:Projects rdfs:subClassOf news:Donor)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule18:
       (?x rdf:type news:Program)
       (news:Program rdfs:subClassOf news:Donee)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule19:
       (?x rdf:type news:Donation)
       (news:Donation rdfs:subClassOf news:DonationandDevelopment)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule20:
       (?x rdf:type news:Development)
       (news:Development rdfs:subClassOf news:DonationandDevelopment)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule21:
       (?x rdf:type news:Social)
       (news:Social rdfs:subClassOf news:Development)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule22:
       (?x rdf:type news:economic)
       (news:economic rdfs:subClassOf news:Development)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule23:
       (?x rdf:type news:DonationandDevelopment)
```

```
       (news:DonationandDevelopment rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:DonationandDevelopment)
]
[EcRule24:
       (?x rdf:type news:ServiceProvider)
       (news:ServiceProvider rdfs:subClassOf news:IndustrialDevelopment)
->     (?x rdfs:subClassOf news:IndustrialDevelopment)
]
[EcRule25:
       (?x rdf:type news:GoodsManufacturers)
       (news:GoodsManufacturers rdfs:subClassOf news:IndustrialDevelopment)
->     (?x rdfs:subClassOf news:IndustrialDevelopment)
]
[EcRule26:
       (?x rdf:type news:InfrastructureDevelopment)
       (news:InfrastructureDevelopment rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:InfrastructureDevelopment)
]
[EcRule27:
       (?x rdf:type news:Stock)
       (news:Stock rdfs:subClassOf news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule28:
       (?x rdf:type news:OthersInvestment)
       (news:OthersInvestment rdfs:subClassOf news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule29:
       (?x rdf:type news:Bond)
       (news:Bond rdfs:subClassOf news:Investment)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule30:
       (?x rdf:type news:Investment)
       (news:Investment rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:Investment)
]
[EcRule31:
       (?x rdf:type news:MicroEnterprise)
       (news:MicroEnterprise rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:MicroEnterprise)
]
[EcRule32:
       (?x rdf:type news:MinesandEnergy)
       (news:MinesandEnergy rdfs:subClassOf news:Economy)
->     (?x rdfs:subClassOf news:MinesandEnergy)
]
[EcRule33:
       (?x rdf:type news:NonMetalliferousMinerals)
       (news:NonMetalliferousMinerals rdfs:subClassOf news:MinesandEnergy)
->     (?x rdfs:subClassOf news:MinesandEnergy)
]
[EcRule34:
       (?x rdf:type news:NaturalGas)
       (news:NaturalGas rdfs:subClassOf news:MinesandEnergy)
->     (?x rdfs:subClassOf news:MinesandEnergy)
]
[EcRule35:
       (?x rdf:type news:MetalliferousOres)
       (news:MetalliferousOres rdfs:subClassOf news:MinesandEnergy)
->     (?x rdfs:subClassOf news:MinesandEnergy)
]
[EcRule36:
       (?x rdf:type news:BuildingOrnaments)
       (news:BuildingOrnaments rdfs:subClassOf news:MinesandEnergy)
->     (?x rdfs:subClassOf news:MinesandEnergy)
]
[EcRule37:
       (?x rdf:type news:OverallEconomyGrowth)
       (news:OverallEconomyGrowth rdfs:subClassOf news:Economy)
```

```
->      (?x rdfs:subClassOf news:OverallEconomyGrowth)
]
[EcRule38:
        (?x rdf:type news:Trade)
        (news:Trade rdfs:subClassOf news:Economy)
->      (?x rdfs:subClassOf news:Trade)
]
[EcRule39:
        (?x rdf:type news:Overseas)
        (news:Overseas rdfs:subClassOf news:Trade)
->      (?x rdfs:subClassOf news:Trade)
]
[EcRule40:
        (?x rdf:type news:Inland)
        (news:Inland rdfs:subClassOf news:Trade)
->      (?x rdfs:subClassOf news:Trade)
]
[EcRule41:
        (?x rdf:type news:WaterResource)
        (news:WaterResource rdfs:subClassOf news:Economy)
->      (?x rdfs:subClassOf news:WaterResource)
]
# ======================================
#              Accidents
# ======================================
[AccRule1:
        (?x rdf:type news:NuclearTerrorism)
        (news:NuclearTerrorism  rdfs:subClassOf  news:Terrorisms)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule2:
        (?x rdf:type news:EcoTerrorism)
        (news:EcoTerrorism  rdfs:subClassOf  news:Terrorisms)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule3:
        (?x rdf:type news:CyberTerrorism)
        (news:CyberTerrorism  rdfs:subClassOf  news:Terrorisms)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule4:
        (?x rdf:type news:BioTerrorism)
        (news:BioTerrorism  rdfs:subClassOf  news:Terrorisms)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule5:
        (?x rdf:type news:Terrorisms)
        (news:Terrorisms  rdfs:subClassOf  news:Intentional)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule6:
        (?x rdf:type news:War)
        (news:War  rdfs:subClassOf  news:HumanMade)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule7:
        (?x rdf:type news:Carelessness)
        (news:Carelessness  rdfs:subClassOf  news:ManMadeDisaster)
->      (?x rdfs:subClassOf news:ManMadeDisaster)
]
[AccRule8:
        (?x rdf:type news:DisasterPrevention)
        (news:DisasterPrevention  rdfs:subClassOf news:Accidents)
->      (?x rdfs:subClassOf news:DisasterPrevention)
]
[AccRule9:
        (?x rdf:type news:EmergencyResponse)
        (news:EmergencyResponse  rdfs:subClassOf news:DisasterPrevention)
->      (?x rdfs:subClassOf news:DisasterPrevention)
]
[AccRule10:
```

```
       (?x rdf:type news:Preparedness)
       (news:Preparedness  rdfs:subClassOf news:DisasterPrevention)
->     (?x rdfs:subClassOf news:DisasterPrevention)
]
[AccRule11:
       (?x rdf:type news:Prevention)
       (news:Prevention  rdfs:subClassOf news:DisasterPrevention)
->     (?x rdfs:subClassOf news:DisasterPrevention)
]
[AccRule12:
       (?x rdf:type news:Accidents)
       (news:Accidents rdfs:subClassOf news:Category)
->     (?x rdfs:subClassOf news:Accidents)
]
[AccRule13:
       (?x rdf:type news:Tornadoes)
       (news:Tornadoes rdfs:subClassOf news:Sudden)
->     (?x rdfs:subClassOf news:NaturalDisaster)
]
[AccRule13:
       (?x rdf:type news:Hurricanes)
       (news:Hurricanes rdfs:subClassOf news:Sudden)
->     (?x rdfs:subClassOf news:NaturalDisaster)
]
[AccRule14:
       (?x rdf:type news:EarthQuake)
       (news:EarthQuake rdfs:subClassOf news:Sudden)
->     (?x rdfs:subClassOf news:NaturalDisaster)
]
[AccRule15:
       (?x rdf:type news:Flood)
       (news:Flood rdfs:subClassOf news:Gradual)
->     (?x rdfs:subClassOf news:NaturalDisaster)
]
[AccRule16:
       (?x rdf:type news:Famine)
       (news:Famine rdfs:subClassOf news:Gradual)
->     (?x rdfs:subClassOf news:NaturalDisaster)
]
# ======================================
#       Science andTechnology
# ======================================
[ScT1:
       (?x rdf:type news:Programs)
       (news:Programs  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT2:
       (?x rdf:type news:Institution)
       (news:Institution  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT3:
       (?x rdf:type news:Equipment)
       (news:Equipment  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT4:
       (?x rdf:type news:Crew)
       (news:Crew  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT5:
       (?x rdf:type news:Clients)
       (news:Clients  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT6:
       (?x rdf:type news:Channel)
       (news:Channel  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
```

```
]
[ScT7:
       (?x rdf:type news:AgencyType)
       (news:AgencyType  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT8:
       (?x rdf:type news:AgencyPurpose)
       (news:AgencyPurpose  rdfs:subClassOf  news:BroadcastingAgency)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT9:
       (?x rdf:type news:BroadcastingAgency)
       (news:BroadcastingAgency  rdfs:subClassOf
news:Science_and_Technology)
->     (?x rdfs:subClassOf news:BroadcastingAgency)
]
[ScT10:
       (?x rdf:type news:CreativityWork)
       (news:CreativityWork  rdfs:subClassOf  news:Science_and_Technology)
->     (?x rdfs:subClassOf news:CreativityWork)
]
[ScT11:
       (?x rdf:type news:Informations)
       (news:Informations  rdfs:subClassOf  news:InformationandCommunication)
->     (?x rdfs:subClassOf news:InformationandCommunication)
]
[ScT12:
       (?x rdf:type news:Communication)
       (news:Communication  rdfs:subClassOf  news:InformationandCommunication)
->     (?x rdfs:subClassOf news:InformationandCommunication)
]
[ScT13:
       (?x rdf:type news:InformationandCommunication)
       (news:InformationandCommunication rdfs:subClassOf
news:Science_and_Technology)
->     (?x rdfs:subClassOf news:InformationandCommunication)
]
[ScT14:
       (?x rdf:type news:Researchtypes )
       (news:Researchtypes rdfs:subClassOf  news:ResearchandStudy)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT15:
       (?x rdf:type news:Researchpurposes)
       (news:Researchpurposes rdfs:subClassOf  news:ResearchandStudy)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT16:
       (?x rdf:type news:Researchmethods)
       (news:Researchmethods rdfs:subClassOf  news:ResearchandStudy)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT17:
       (?x rdf:type news:Publishing)
       (news:Publishing rdfs:subClassOf  news:ResearchandStudy)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT18:
       (?x rdf:type news:Scientificresearch)
       (news:Scientificresearch rdfs:subClassOf  news:KindsofResearch)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT19:
       (?x rdf:type news:Historicalresearch)
       (news:Historicalresearch rdfs:subClassOf  news:KindsofResearch)
->     (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT20:
       (?x rdf:type news:KindsofResearch)
       (news:KindsofResearch rdfs:subClassOf  news:ResearchandStudy)
```

```
->      (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT21:
        (?x rdf:type news:ResearchandStudy)
        (news:ResearchandStudy rdfs:subClassOf  news:Science_and_Technology)
->      (?x rdfs:subClassOf news:ResearchandStudy)
]
[ScT22:
        (?x rdf:type news:LocalAgency)
        (news:LocalAgency rdfs:subClassOf  news:Agency)
->      (?x rdfs:subClassOf news:Agency)
]
[ScT23:
        (?x rdf:type news:InternationalAgency)
        (news:InternationalAgency rdfs:subClassOf  news:Agency)
->      (?x rdfs:subClassOf news:Agency)
]
[ScT24:
        (?x rdf:type news:Website)
        (news:Website rdfs:subClassOf  news:Medium)
->      (?x rdfs:subClassOf news:Medium)
]
[ScT25:
        (?x rdf:type news:Television)
        (news:Television rdfs:subClassOf  news:Medium)
->      (?x rdfs:subClassOf news:Medium)
]
[ScT26:
        (?x rdf:type news:Radio)
        (news:Radio rdfs:subClassOf  news:Medium)
->      (?x rdfs:subClassOf news:Medium)
]
[ScT27:
        (?x rdf:type news:Newspaper)
        (news:Newspaper rdfs:subClassOf  news:Medium)
->      (?x rdfs:subClassOf news:Medium)
]
[ScT28:
        (?x rdf:type news:Magazine)
        (news:Magazine rdfs:subClassOf  news:Medium)
->      (?x rdfs:subClassOf news:Medium)
]
[ScT29:
        (news:Television rdfs:subClassOf  news:Medium)
->      (news:Television rdfs:subClassOf news:BroadcastingAgency)
]
[ScT30:
        (news:Radio rdfs:subClassOf  news:Medium)
->      (news:Radio rdfs:subClassOf news:BroadcastingAgency)
]
```

Declaration

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all sources of materials for the thesis have been duly acknowledged.

MERON SAHLEMARIAM

This thesis has been submitted for examination with my approval as an advisor.

MULUGETA LIBSIE (Ph. D.)

Addis Ababa, Ethiopia

January 2009