ADDIS ABABA UNIVERSITY

COLLEGE OF NATURAL SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

Information Extraction from Amharic language Text: Knowledge-poor Approach

Bekele Worku Agajyelew

A THESIS SUBMITTED TO

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

Addis Ababa, Ethiopia

June, 2015

ADDIS ABABA UNIVERSITY

COLLEGE OF NATURAL SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

Information Extraction from Amharic language Text: Knowledge-poor Approach

By:

Bekele Worku Agajyelew

Advisor: Yaregal Assabie (PhD)

Signature of the Board of Examiners for Approval

Name	**Signature**
1. Yaregal Assabie (PhD), Advisor	_____
2. Sebsibe Hailemariam (PhD)	_____
3. Mesfin Kifle (PhD)	_____

Dedication

To my Mother, Abozenech G/Tsadik

Acknowledgement

First of all, I would like to thank God the Almighty and St. Mary for giving me the wisdom, strength, support and knowledge in exploring things.

Next, I would like to express my gratitude to my advisor Dr. Yaregal Assabie, who was always there during the process of this thesis work for giving me support, encouragement and continuous advice, without his invaluable help and support, the research wouldn't have been completed.

Finally, I would express my special thanks to my mother and father who instilled in me the values I strive to live by. Then, I would like to thank all of my family members and friends for giving me support and encouragements.

Table of Contents

List of Tables

List of Figures

Abbreviations

AIE	Amharic Information Extractor
ANNIE	A Nearly-New Information Extraction
DARPA	Defense Advanced Research Projects Agency
ENA	Ethiopia News Agency
GATE	General Architecture for Text Engineering
HMM	Hidden Makrov Model
HTML	Hyper Text Markup Language
IE	Information Extraction
IR	Information Retrieval
JAPE	Java Annotation Pattern Engine
MUC	Message Understanding Conference
NER	Named Entity Recognition
NLP	Natural Language Processing
POS	Part Of Speech
WIC	Walta Information Center

Abstract

During the last two decades with the accelerated Internet development a great amount of data have been being accumulated and stored on the Web. We are drowns with much data at office, home either in printable or electronic form. Then finding the relevant information from this mass data is critical. At this end, information extraction is a technology which creates the structured representation of unstructured texts by extracting relevant entities from them, thereby, making the data analysis realizable.

This work focuses on developing information extraction system from Amharic language text. The proposed system developed using GATE (General Architecture for Text Engineering) text processing environment using knowledge-poor approach on infrastructure domain. By knowledge-poor approach we mean we are using simple rules and gazetteer list for entity identification. Our proposed Amharic text information extractor consists of three phase's namely preprocessing, extraction and post processing. The preprocessing phases used for handling language specific issues and setting the environment ready for extraction process. The second phase is the main unit in our model. It basically performs named entity recognition, coreference resolution and relation extraction and extract relevant text. The post processing step annotates the selected data and presents the extracted information in a structured form.

Various evaluation techniques, which are used to evaluate the performance of our proposed model were used. The usual precision, recall and F-measure were used to measure the efficiency of the proposed work. We have used 24760 instances for training and testing our model. Our evaluation was conducted on name entity recognition component separately and the overall system as information extraction component. Accordingly, the system achieves the F-measure of 89.1 % on the named entity recognition and in the overall it achieves the F-measure of 89.8%.

Key words: Information Extraction, Amharic Text Information Extraction, Coreference Resolution, Relation Extraction, GATE

CHAPTER ONE: INTRODUCTION

1.1 General Background

In the current era of information technology huge amount of information is available in machine readable format. The information overload we have drowns with is a reality. Reports showed that 5 Exabyte of new information in all formats were produced in 2002 of which over 1,600 terabytes represent textual information [1]. As increasing storage capacity of computers and decreasing price, the amount of information retained by businesses and institutions is likely to increase. Searching that information and deriving useful facts, however, will become more awkward unless new techniques are developed to automatically manage the data. This phenomenon will lead to the situation that relevant information will get buried since it is never revealed. From these points, it becomes obvious that it is increasingly difficult to keep up with new information in a domain or find a piece of information which was produced in the past. This problem can be alleviated by using computers, which can deal with information in a much faster manner than humans, using automatic tasks such as summarization, retrieval or extraction.

Among the automatic text processing tasks, both summarization and retrieval present unstructured data to the user. The term unstructured refers to free texts or a document which does not have a format that can be processed by computer. But, information extraction will display structured data. Information extraction is the next step up from information retrieval in fulfilling information processing needs. Information extraction is a technology that is futuristic from the user's point of view in the current information-driven world. Rather than indicating which documents need to be read by a user, it extracts pieces of information that are salient to the user's needs. The aim of information extraction is presenting tabular data by ignoring the irrelevant ones. Information extraction picks up the information of interest which is relevant to the specification. Therefore, information extraction system is one of the most important mechanisms that we can use to display only the relevant text from large collection of documents.

The development of information extraction is largely due to the influence of the Defense Advanced Research Projects Agency (DARPA) sponsored Message Understanding Conferences

(MUC) program started in late 1980's [5]. Message Understanding Conferences were initiated and financed by DARPA to encourage the development of new and better methods of information extraction. It was the first large scale effort to boost research into automatic information extraction and it would define the research field for the decades to come. Considerable support came from DARPA, the US defense agency, who wished to automate routine tasks performed by government analysts, such as scanning newspapers for possible links to terrorism.

Information Extraction (IE) in the sense of the MUC [2] has been defined as the extraction of information from a text in the form of text strings and processed text strings which are placed into slots labeled to indicate the kind of information that can fill them. From this definition, for example, a slot labeled NAME would contain a name string taken directly out of the text. From this we can understand that the input to information extraction is a set of texts, usually unclassified newswire articles, and the output is a set of filled slots. Therefore; in IE there has to be a pre-defined template/slot to be filled out.

Thus, information extraction is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. In most of the cases this activity concerns processing human language texts by means of Natural Language Processing (NLP). Recent activities on information extraction is not bounded in text data only, it include multimedia document processing like automatic annotation and content extraction out of images, audio and video. Information extraction is the identification of specific information from unstructured data sources and making the information more suitable for further information processing tasks. The mere goal of information extraction is to allow computation to be done on unstructured data to come up with structured one. Structured data is semantically well-defined data from a chosen target domain, interpreted with respect to category and context. On the other hand unstructured data have no any model and does not processed by computer which includes data on the web, office data and etc.

Information extraction aims creating structured format of data with database. Information extraction is processing and selecting structures and combines data, and the final output of the

extraction process will populate some type of database [3]. Un Yong Nahm [4] defines information extraction as a form of shallow text understanding that locates specific pieces of data in natural language documents, transforming unstructured text into a structured database. Unstructured data cannot be stored in database. Those who work on feeding the database manually consume their time and cannot be accurate. Information extraction makes it simple since the extracted data can be stored for further processing. Given a free text an IE system will extract the specific information from the text and put them in the database so that they can easily be retrieved and managed. The stored data can also be used for further analysis.

The General Architecture for Text Engineering (GATE) defined information extraction as a system which analyses unstructured text in order to extract information about pre specified types of events, entities or relationships [6]. The work by Riloff and Lorenzen [7] present the domain specific nature of information extraction. Information extraction systems extract domain-specific information from natural language text. The domain and types of information to be extracted must be defined in advance. Thus, we can understand that in the case of information extraction one should handle both domain and language specific issue.

The numbers of Amharic documents on the web and in other machine readable forms are also increasing from time to time. As a result of this growth, the huge amount of text which contains different valuable information which can be used in education, business, security and other many areas are hidden under the unstructured representation of the textual data. This shows that getting the right information for decision making from existing abundant unstructured text is a big challenge. The non-availability of tools for extracting and exploiting the valuable information which is effective enough to satisfy the users need have been a major problem for years. Since extraction of information is language specific, information extraction system developed for English or any other language cannot work for Amharic language. Getasew [8] was done information extraction system for Amharic language text. Getasew's work was bounded in extracting named and numeric entities only using gazetteer. Thus, this study attempted to fill the gaps identified and develop robust information extractor for Amharic language text using different techniques, and algorithm proposed so as to come up with a better extraction.

1.2 Motivation

Amharic is serving as an official language of Federal Democratic Republic of Ethiopia, Southern nation nationalities and Regional State of Amhara. Being an official language, it is used as medium of instruction for primary and junior secondary schools. It is also a field of specialization at Diploma, Bachelor Degree, and Master's Degree levels at various universities in Ethiopia. Beside this, a number of literature works, newspapers, magazines, education resources, official credentials and religious documents are published and available in the language. Hence, above all these facts initiate us to conduct this work.

On the other case a lot of knowledge is available in unstructured text. News articles, messages, research paper may be machine accessible, but they cannot be used directly because the data in these texts is unstructured. But it follows some rules, they may be semi-structured in web pages or tables, but even natural language text follows grammar rules and some repeating patterns. The idea behind information extraction is that by exploiting these rules and pattern the data from these texts can be extracted and feed into a structured database for further use. This is the other pushing factor to come across information extraction.

On the other hand, in comparable with foreign languages, Amharic is one of the most resource scarce languages in context of NLP. Today the improvement in modern technology raises the availability of digital information on the Internet, which is written by the Amharic language. Identifying relevant information from a given text manually is time consuming and tiresome task. Even though Amharic lacks NLP resources and we are drowns with much data, only Getasew [8] and Ibrahim [9] tried to conduct research on information extraction of Amharic language. But information extraction in Amharic language is still the hottest topic. Finally, the knowledge gap found in Getasew and Ibrahim work also initiate us.

In general, lack of active research on the automatic information extraction and a dramatic growth of electronic Amharic document from time to time are a motivating factor for this work to come up with modules that can alleviate or minimize these problems.

1.3 Statement of the problem

Currently there are more text data in electronic form than ever before, but much of it is not used because no human can read, understand, and synthesize megabytes of text on an everyday basis. Missed information and lost opportunities has spurred researchers to explore various information management strategies to establish order in the text wilderness. Due to this information overload, extracting relevant information has become tiresome and time consuming. Hence, there should be effective and efficient system which extracts structured information.

A lot of valuable information is being produced in Ethiopia, most of them written in Amharic either in hard copy or digital format. The documents contain information related to: research in many fields; particularly agriculture and water resource development; information on the development of the tourist and business sectors; government policies; and bulk of information produced by offices in day to day work. Informative bulletins, magazines and newsletters are also regularly produced by most government ministries, UN agencies, and NGOs. Specially, digitally information, it is available in abundance and in a myriad of forms to an extent of making it near impossible to search manually, sift and choose relevant information. Therefore, this information must instead be filtered, and extracted to avoid drowning in it.

Different researchers propose different methods that make information extraction possible. The information extraction system developed for English or any other language and for some specific domain cannot work for other languages of the same domain. This is due to the reason that information extraction system has to be trained about the different nature of the language and the domain for which they are developed for. Amharic is one the widely used language in Ethiopia which has its own phonetic and grammar. In this regard, building efficient information extraction system for Amharic language is an essential task. Although small in number, there are two works which are done on information extraction from Amharic text. The first is by Ibrahim [9] which used Hidden Makrov Model to extract information and the other is Getasew [8] which used classification based extraction.

Ibrahim [9] proposes information extraction from Amharic text using Hidden Makrov Model. The proposed Information extraction tool is developed for extracting information from a single

sentence. It uses the slots subject, object, action and reporter and tries to extract information from the news text which contain the above listed four slots on a single sentence. If one of these components does not exist in the sentence the system does not extract the information. Therefore, this work is limited to extracted information from single sentence and a single sentence cannot contain the full details of some action or entity.

Getasew [8] proposes information extraction from Amharic news text which is based on classification machine learning method. He employs categorization on collection of news to identify the required category. In this work much emphasis is given to categorization. The success of extraction is depending on text categorization sub component. If the categorization subcomponent failed to categorize correctly, extraction result will not be as expected. On the other hand the work is not addressing the necessary information extraction subtasks like relation extraction, co-reference and anaphoric resolution. For named entity recognition he used gazetteer which contains list of names collected from different sources.

The following gaps were identified particularly from Getasew's [8] works:

- Candidate element selection was based on gazetteer list and it cannot work out of the identified lists
- He employs categorization on the selected candidate element which is not information extraction.
- He used different data sources for categorization and he used economy news data a source for information extraction. Accordingly, his candidate element extraction component depends on the success of his categorization subcomponent.

In general Ibrahim and Getasew tried to develop a model for information extraction from Amharic text. But the system they developed was not sufficient to address what information extraction demands. The mentioned limitations decrease the quality of the extraction result. Hence, this work would explore previous work in detail, as well as examine the performance of automatic Amharic information extraction system by incorporating co-reference resolution and relation extraction.

Therefore, we set the following research question to examine the problem in Amharic language text information extraction.

- What type of data is suitable for information extraction?
- What type of information is relevant for the end user from a news article?
- What type of method is best in person name identification from Amharic news articles?
- What kinds of additional techniques will increase the performance of information extraction system?

1.4 Objectives

Objective of the research are stated as general and specific as follows:

General Objective

The general objective of this thesis is to investigate the way of designing and developing automatic information extraction system for Amharic text using knowledge-poor approach.

Specific Objective

To achieve the general objective, specific objectives given below are identified:

- Conduct literature review on information extraction to better understanding of the state-of-the-art information extraction
- Identify and collect a corpus of Amharic text
- Analyze the general grammatical structure of Amharic language for the purpose of identifying its characteristics for information extraction.
- Identify different representation of objects in Amharic language
- Design a model for automatic Amharic text information extraction
- Develop a prototype using a designed model
- Test the model with sample data
- Infer conclusion and recommendation based on experimental results.

1.5 Methodology

Methodology is a step by step process by which to systematically solve the research problem [10]. This research will be conducted in order to figure out challenges of implementing Amharic information Extraction system. Towards achieving the main objective the following step by step procedures were followed.

Literature Review

To understand the gap created in previous works and to have full point of view on information extraction different review on literatures was conducted. In this study in order to understand the problem domain and for conceptual understanding literatures which was directly related to the study especially on Amharic language got emphasized.

Corpus collection and Data preparation

Relevant Amharic text data was collected from different data sources in order to present the data to the experiment. Different facts about Amharic language like the grammatical structure and number representation was conducted in order to understand the nature of the Amharic language with respect to information extraction.

Development Tool

In order to develop a prototype system, different appropriate tools were selected and used. The GATE framework and Java programming language is used to implement the different language specific algorithms developed.

Evaluation techniques

The study involves designing Amharic text information extraction model and implementing the prototype of the model. So the performance of the system has to be evaluated. To this end, corpus was prepared, queries were constructed and relevance judgment was made for evaluating effectiveness of the work to measure the effectiveness of the system.

1.6 Scope and Limitation

The focus of the study is extracting structured information from unstructured Amharic language text data. The scope of the study covered extraction of named entities and numbers, extraction of relation between entities and extraction of co-reference relation.

Other data types such as video, audio and graphic were not the focus of the study. Building information extraction system that can extract from large document is a time consuming and a challenging task. So, the study was bounded with limited corpus, in order to evaluating the performance of the system developed.

1.7 Application of Results

Being the member of the information age, people are puzzled with the problem of finding relevant information efficiently and effectively. With the rise of data finding relevant information needs scientific and computerized techniques. Applying information extraction on text is linked to the problem of text simplification in order to create a structured view of the information present in free text.

The Amharic text extraction model which is the core of this study will benefit a number of users. The data at hand will be unstructured and that will consume time and energy to dig out the relevant information. The newly designed Amharic information extraction system will display structured data from unstructured or semi-structured text which make it easy for the different users to access it within a short period of time. In short relevant and structured information will be presented for the user so that the user does not need to read unnecessary detail of certain document. On the other hand the user may need the extracted structured data for further processing. But unstructured data cannot be stored in database. This problem can be solved by the extraction system that is to be developed since it can store extracted information into database. Therefore, user can use stored extracted data for further analysis.

1.8 Thesis Organization

The thesis is presented in six chapters. The current chapter discussed about the general background of the research work, problem description, and objective of the study, methodology as well as scope and limitation of the study. Chapter two present reviews made on different literatures regarding Information extraction together with its approaches and different machine learning techniques and discuss the Amharic language. Chapter three discusses related work done on information extraction using different approaches and different domains on different languages. The design and implementation of information extraction model for Amharic text was discussed under the fourth chapter. Chapter five present the details of experimentation of the developed model. Finally the last chapter discusses conclusion and recommendation.

CHAPTER TWO: LITERATURE REVIEW

This chapter presents the general overview of information extraction in detail. In the sections below different approaches for information extraction, the major subtasks in information extraction, evaluation matrices and overview on Amharic language were reviewed. The section also covers natural language processing fields which are related with information extraction in order to understand the extent of the work to be done.

2.1 Information Extraction

In this information age we are drowns with much data around us. Data is a raw fact which has to be processed in order to be information. With a huge amount of data available on the Web, in offices as well as personal documents, it is important to have some technologies and tools to analyze it, derive information and gain knowledge from it which can be used later for any other purposes. Thus, information extraction is one of those technologies which allow obtaining useful information from data presented in any unstructured or semi-structured textual form. With large amounts of potentially useful information in hand, an information extraction system can then transform the raw material, refining and reducing it to a germ of the original text. The goal of IE is to transform the data from unstructured form into structured representation by finding relevant information while ignoring extraneous and irrelevant ones.

A pioneer of information extraction was the field of text understanding. Text understanding requires understanding full natural language texts. But information extraction narrowed it by focusing on specific features than full text. The field of information extraction was also played by the MUC series of conferences. The conferences were organized between 1987 and 1998 by the support of DARPA [5]. The conferences were a medium for reporting the work and results obtained in associated competitions for IE systems. Table 2.1 was taken from [5] briefs the goals the series of MUC conferences.

Table 2.1: A brief overview of the main developments during MUC conferences

MUC conference	Goal
MUC-1 (1987)	• Was mostly exploratory each group was using its own representation model • There was no formal evaluation at the end.
MUC-2 (1989)	• Defined the task as a template filling one (10 slots) • Introduced the evaluation measures of recall and precision. • Both MUC-1 and MUC- 2 used military messages about naval sightings and engagements as input texts.
MUC-3 (1991) and MUC-4 (1992)	• The texts used were reports of terrorist events in Central and South America • The templates had 18/24 slots.
MUC-5 (1993)	• Two types of text were used: international joint ventures and electronic circuit fabrication, in two languages: English and Japanese. • The joint venture task required 11 templates with a total of 47 slots for the output.
MUC-6 (1995)	Its stated goals were: • To identify components that are task-independent, can be performed automatically with reasonable accuracy and have practical uses; • To focus on portability in the task, defined as the ability to rapidly re-target a system to extract information about a different class of events; • To encourage deeper understanding by requiring the systems to perform co-reference resolution, word sense disambiguation and Predicate-argument structure extraction.
MUC-7 (1998)	▪ The task definition was similar to that of MUC-6, the only addition being the Template Relation sub-task.

The work presented at [11] define information extraction as "a process of scanning text for information relevant to some interest, including extracting entities, relations, and, most challenging, events or who did what to whom when and where." It demands better analysis than key word searches, but it is much limited than text understanding. IE is all about extracting

structural factual data mostly from unstructured text (web pages, text documents, office documents, presentations, and so on). IE usually uses NLP tools, lexical resources and semantic constraints for better efficiency.

Information extraction is a term which has come to be applied to the activity of automatically extracting pre-specified sorts of information from short, natural language texts typically, but by no means exclusively, newswire articles. For instance, one might scan business newswire texts for announcements of management succession events (retirements, appointments, promotions, etc.), extract the names of the participating companies and individuals, the post involved, the vacancy reason, and so on. In another way, IE may be seen as the activity of populating a structured information source (or database) from an unstructured or free text, information source. This structured database is then used for some other purpose: for searching or analysis using conventional database queries or data-mining techniques; for generating a summary; for constructing indices into the source texts.

Information extraction starts with a collection of texts, and then transforms them into information that is more readily digested and analyzed. The input to information extraction can be unstructured documents written in natural language or semi-structured documents like web pages, tables or itemized and enumerated lists and the output can be entities (Person Entity, Organization Entity), Terms, Relations, Properties, and Events. The output can be represented in filled slots. The set of filled slots may represent an entity with its attributes, a relationship between two or more entities, or an event with various entities playing roles and/or being in certain relationships. Entities with their attributes are extracted in the template element task; relationships between two or more entities are extracted in the template relation task; and events with various entities playing roles and/or being in certain relationships are extracted in the scenario template task. Thus, give unstructured or semi-structured document the role of IE is extracting features such as name or location, and relationships among the features in a natural language text. Table 2.2 shows an example extracted features and feature values.

Table 2.2: Extracted Features and Feature Values

Feature	**Feature Value**
Name	ALEXANDER F. TOMLD
Height	SIX FEET TALL
Weight	WEIGHING 170 TO 180 POUND
Eye Color	BLUE
Hair Color	BROWN
Race	WHITE

As we have seen above different scholars define information extraction differently. From these contexts we can generalize information extraction as an automated process which takes texts as input and produces fixed-format, unambiguous data as output. The output data may be used directly for display to users for their quick information need, or may be stored in a database or spreadsheet for later analysis. In general the goal of information extraction is to transform text into a structured format and thereby reducing the information in a document to a tabular structure. Specified information can then be extracted from different documents with a heterogeneous representation and be summarized and presented in a uniform way.

2.2 Architecture of Information Extraction System

Information extraction is a domain specific task [7, 44]. This means, an IE system developed and working effectively for one domain cannot perform with the same accuracy and efficiency on another domain or may not work at all. For example, IE developed for terrorist event cannot work for housing advertisement domain. Information extraction can also be developed using either rule based or machine learning approach. However, there are core elements that are shared by nearly every extraction system, regardless of whether it is designed according to the rule-based or machine-learning paradigm.

The architecture given by Applet and Israel [12] consists of the four primary modules that every information extraction system has, namely a tokenizer, some sort of lexical and or morphological

processing, some sort of syntactic analysis, and some sort of domain-specific module that identifies the information being sought in that particular application. Figure 2.1 illustrates the components shared by IE systems by Applet and Israel [12].

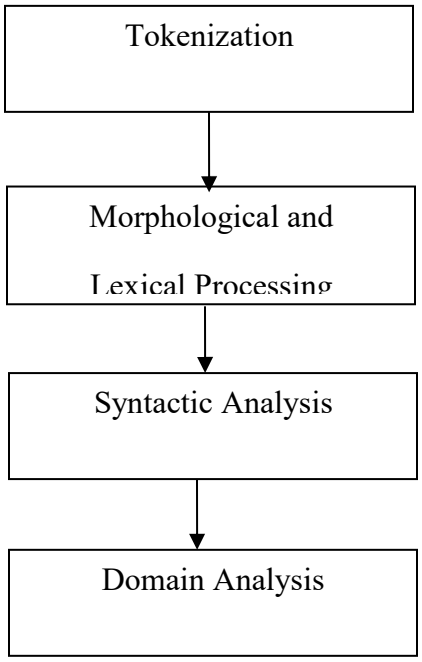

Figure 2.1: Typical Architecture of Information Extraction System

The above illustration is a typical and bare-bone architecture. Depending on the requirements of a particular application, it is likely to be desirable to add additional modules to the bare-bones system illustrated above.

2.3 Related NLP Fields

2.3.1 Natural Language Understanding

Natural language understanding is crucial for most information extraction tasks because the desired information can only be identified by recognizing conceptual roles. We use the term conceptual role to refer to semantic relationships that are defined by the role that an item plays in context. For example, extracting noun phrases that refer to people can be done without regard to context by searching for person names, titles, and personal pronouns, such as “Mary,” “John,”

“Smith,” “Mr.,” “she,” and “him” [13]. Natural language understanding requires a computer to understand the entire language text. Information extraction is a more limited task than natural language understanding. To mitigate the complexity, the scope of natural text understanding is narrowed down to information extraction because IE interested in specific features rather than full text understanding.

2.3.2 Information Retrieval

The paper presented in [52] discuss information extraction is different from the more mature technology of information retrieval. Rather than to extract information the objective of IR is to select a relevant subset of documents from a larger collection based on a user query. The user must then browse the returned documents to get the desired information.

Information retrieval is about returning the information that is relevant for a specific query or field of interest. Note that this information could also be in the form of general documents, sure enough search engines are a notable example of such task. It is possible to say that the most important entities recognizable for information retrieval are the initial set of documents/information and the query that specify "what to search for". On the other hand information extraction is more about extracting (or inferring) general knowledge (or relations) from a set of documents or information. Information extraction is about structuring unstructured information - given some sources all of the (relevant) information is structured in a form that will be easy for processing.

The contrast between the aims of IE and IR systems can be stated as follows; IR retrieves relevant documents from collections while IE extracts relevant information from documents. Hence the two techniques are complementary and used in combination they can provide powerful tools for text processing [14]. The work presented in [52] discusses the complexity comparison between IR and IE. Their compassion conclude that information extraction is often more difficult than information retrieval because IE requires more detailed knowledge about a document such as its organization, person, location, time, etc. Furthermore, IE systems are often required to establish relationships between features.

The basic operations of typical information retrieval systems can be grouped into two main categories: indexing and matching (or retrieval). The purpose of the indexing process is to identify the most descriptive words existing in a text. The aim of the query matching process is to derive a list of documents ranked in decreasing order of relevance to a given query [15]. An information retrieval process begins when a user enters a query into the system. Queries are formal statements of information needs, for example search strings in web search engines. Google is one of the best information retrieval systems for the web. Just like Google, the output of an IR system is a subset of documents that are relevant to a user's query. However, the goal of IE systems is to extract pre-specified features from documents rather than the documents themselves. The extracted features are usually entered into a database automatically. In short, IR is document retrieval while IE is feature retrieval. Figures 2.2 and 2.3 taken from [51] depict the difference between IR and IE.

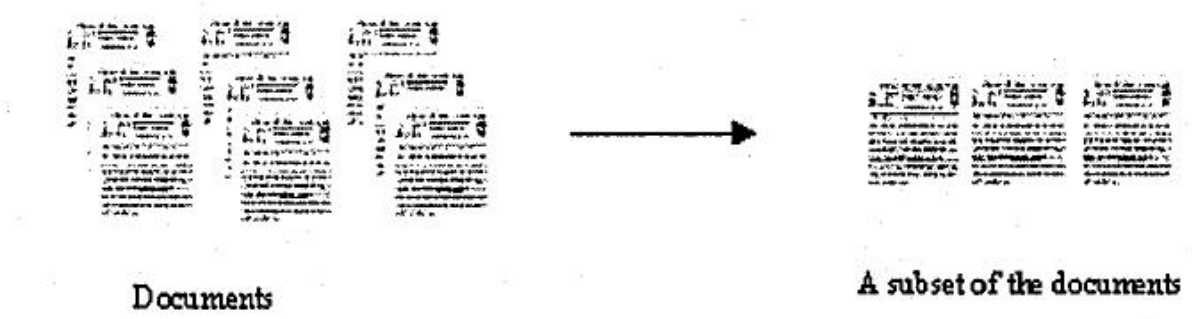

Figure 2.2: Information retrieval

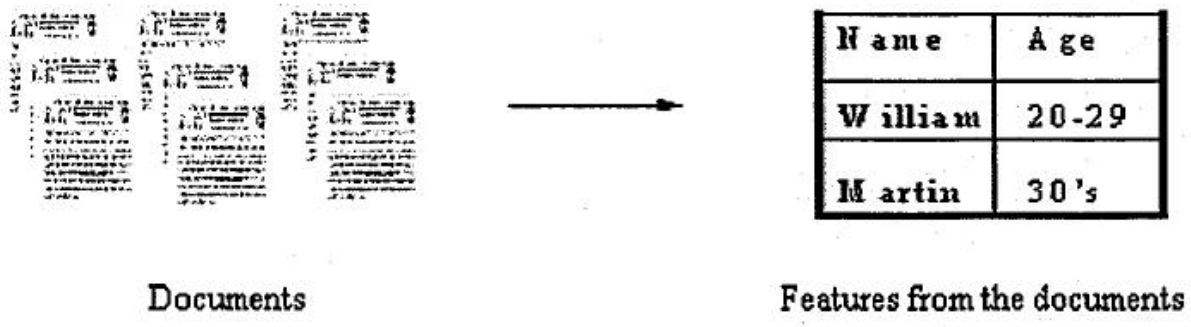

Figure 2.3: Information extraction

2.3.3 Text summarization

In many situations, users would and do prefer to look at abstracts rather than at the whole text, before they decide whether they are going to read through the entire text or not. The abstract that show the concrete idea of the document is summary. Automatic text summarization is the process by which computer program creates a shortened version of text. The product of the process contains the most important points from the original text. In text summarization, the summarized text must convey the meaning of the original document. Search engines such as Google use automatic summarization to produce key phrase extractions in search results.

The work presented at [16] defines a summary as “a text that is produced from one or more texts, that convey important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that". This simple definition captures three important aspects that characterize research on automatic summarization:

- Summaries may be produced from a single document or multiple documents,
- Summaries should preserve important information and
- Summaries should be short

The paper presented in [17] evaluates the initial version of RIPTIDES which stands for Rapidly Portable Trans lingual Information extraction and interactive multi Document Summarization, a system that combines information extraction, extraction-based summarization, and natural language generation to support user directed multi document summarization. In this work the authors hypothesize that information extraction-supported summarization will enable the generation of more accurate and targeted summaries in specific domains than is possible with current domain-independent techniques. From the authors work we can notice that involving information extraction in summarization will produce domain specific summary which much shorter than the original document.

In general the goal of a summary is to give the reader an accurate and complete idea of the contents of the source document in a concise form. It is obvious that the text summarization minimize the large amount of text by extracting the most relevant sentence from larger text. But

the summarized text requires the user involvement to read the summary and extract the specific information needs and the data cannot directly used by other computer applications.

2.3.4 Question and answer

Question Answering is a computer science discipline within the fields of information retrieval and natural language processing, which is concerned with building systems that automatically answer questions posed by humans in a natural language. More commonly, question answering systems can pull answers from an unstructured collection of natural language documents. A question answer system accepts questions in natural language form, searches for answers over a collection of documents and extracts and formulates concise answers. Moldovan and Surdeanu [18] discuss the three essential modules in almost all question answering systems. These are question processing, document retrieval, and answer extraction and formulation. Question answering is the process of extracting answers to natural language questions but IE systems extracts the information of interest provided the domain of extraction is well defined. In IE systems, the information of interest is in the form of slot fillers of some predefined templates.

A question answering system that is based exclusively on information extraction patterns is described in [19]. The paper discussed an information extraction system, Textract, in natural language question answering and examines the role of information extraction in question answer application. The authors conclude that named entity tagging is an important component for question answering, natural language shallow parser provides a structural basis for questions, and high-level domain independent information extraction can result in a question answer breakthrough.

In general those related tasks of NLP listed above tried to solve the information overload issues facing in the current world. They all used for solving huge text data problems and making the text more usable for the end user. Information extraction unlike text summarization, information retrieval and question and answering system its output is in structured form which can be easily managed and accessed as it is stored in the database.

2.4 Sub Tasks of Information Extraction

Different IE systems for different languages and different domains using different approaches are developed so far and are still on development but they all use the different task breakdown for IE. By the time that it ended in 1998 (Which is the end of MUC-7) the MUC program had arrived at a definition of IE split into five tasks.

These are:

- Named Entity Recognition
- Coreference Resolution
- Template Element Construction
- Template Relation Construction
- Scenario Template Production

2.4.1 Named Entity Recognition

The simplest and most reliable IE technology is Named Entity Recognition (NER). It is one of the most often extracted types of tokens during extracting information from documents. NER systems identify all the names of people, places, organizations, dates, amounts of money, etc. In short expression named entity recognition is about finding entities. The term "Named Entity", widely used in information extraction, question answering or other NLP applications, was born in the MUC which influenced IE research in the U.S. in the 1990's [20]. To be precise, it was introduced for MUC-6 in 1995. Throughout the MUC series, the term named entity came to include seven categories; persons, organizations, locations (usually referred to as ENAMEX), temporal expressions, dates (TIMEX), percentages, and monetary expressions (NUMEX).

Named entities are one of the most often extracted types of tokens during extracting information from documents. Named entities play a central role in conveying important domain specific information in text, and good named entity recognizers are often required in building practical information extraction systems. There are different types of named entities. MUC played a crucial role to the emergence and fostering of researches in the area of named entity recognition.

These contests had defined the named entity types for their respective domain and the corresponding named entity types for MUC in the table 2.3 shown below.

Table 2.3: Named entity types as defined by MUC

Named Entity	**Example**
PERSON	Smith, Obama, Clinton
ORGANIZATION	IBM, LIFAN Motors
LOCATION	Texas, Cape Town
DATE	25/02/2010, January 15
TIME	8:30 AM
PERCENTAGE	10%
MONETARY AMOUNT	$120.00, €250

To have a clear view in named entity recognition, let us see Figure 2.4 which is taken from IE software distributed with the GATE system Cunningham [21]. This named entity recognition is for English texts.

Figure 2.4: Output of GATE Named Entity Recognizer for English Text

As discussed in [12] the difficulty of the name recognition task depends on the type of text one is analyzing. In English, upper and lower case make it relatively easy to recognize the fact that a sequence of words is a proper name. The key problem is determining what kind of a name it is. Since there is no such kind of lower case and upper case distinctions, named entity recognition in Amharic language is difficult.

In general, Because of the presence of named entities in almost any text that one would wish to analyze, and because of the importance of named entities in the identification of objects of interest, almost every information extraction system implements some kind of named entity identifier. Thus, named entities are one of the basic IE backbones and their successful identification gives us a substantial part of the information that we wish to extract.

2.4.2 Coreference Resolution

Any given entity in a text can be referred to several times and every time it might be referred differently. In order to identify all the ways used to name that entity throughout the document coreference resolution is performed. Coreference or anaphora resolution is the stage when for noun phrases it is determined if they refer to the same entity or not. There are several types of coreference, but the most common types are pronominal and proper names coreference, when a noun is replaced by a pronoun in the first case and by another noun or a noun phrase in the second one [12].

Coreference resolution involves identifying relations between entities in texts. Besides entities identified by named entity recognition, this may also include anaphoric references to those entities. It is concerned with entities and references (such as pronouns) that refer to the same thing. Coreference resolution enables the association of descriptive information scattered across texts with the entities to which it refers. Figure 2.5 shows the result of Coreference resolution for the example text from Figures 2.4. In the Figure 2.5 the entity “Department Homeland Security” is expressed as “Homeland Security” or “Security” in the other paragraph. But these two entities are referring the same thing and coreference resolution will handle such kind identity relationship between entities.

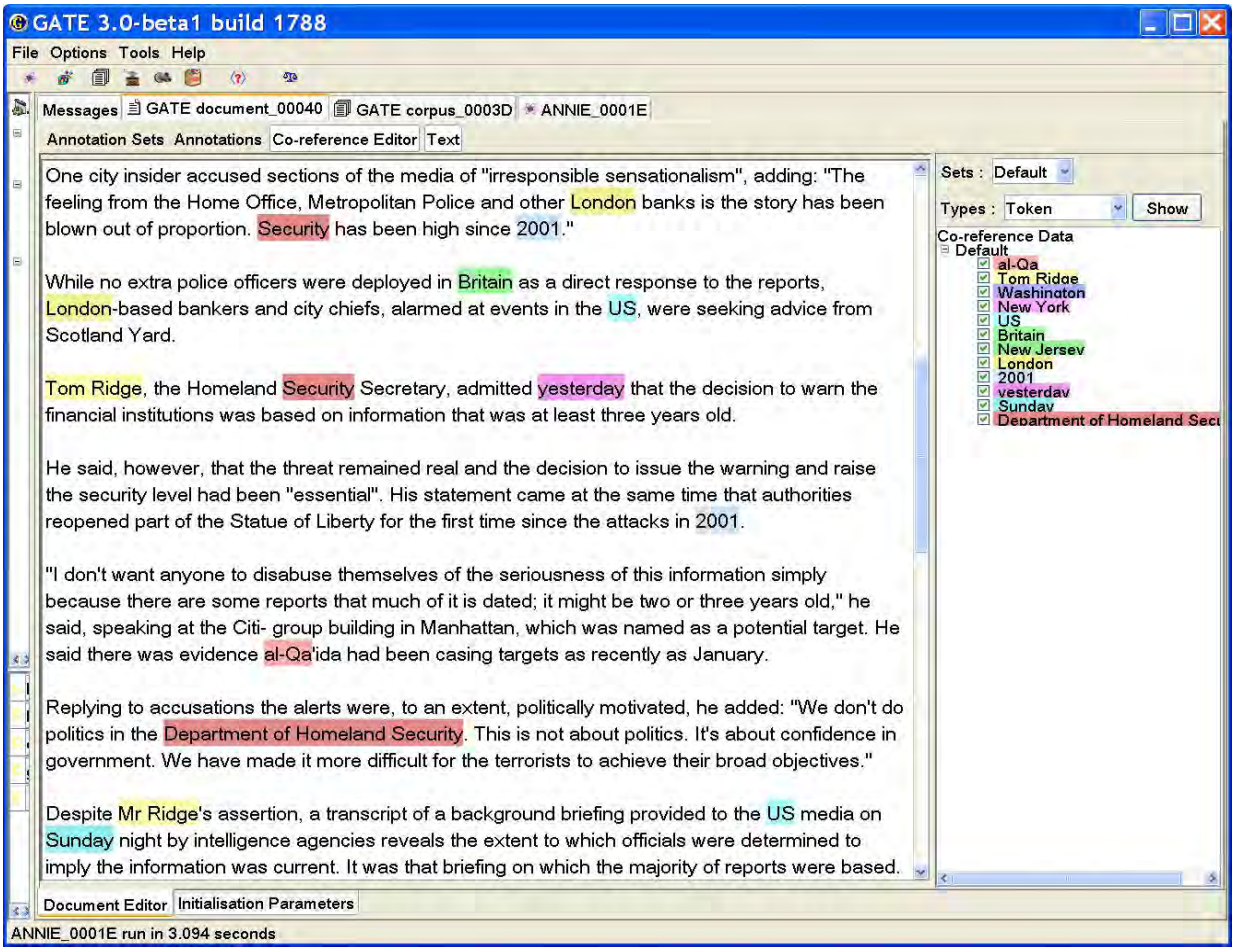

Figure 2.5: Coreference resolution for English text

2.4.3 Template Element Construction

Template element construction task builds on named entity recognition and coreference resolution. Its role is to associate descriptive information with the entities. It is all about what attributes entities have. The different recognized named entities will have different attributes for template element construction. Template element construction is domain dependent, as the types of information that are relevant depend on the types of entities that are important to the application domain. For example, relevant information about an organization includes whether it is private or public, if it is for profit or a charity. Figure 2.6 shows Template Element Construction for the above given example text in Figure 2.4. From Figure 2.6 the entity Organization-001 has three attributes. These attributes are Type: “government”, Name: “Bush administration” and Aliases: “Bush administration” and “government officials”.

Figure 2.6: Template Element Construction

2.4.4 Template Relation Construction

Before MUC-7, relations between entities were part of the scenario-specific template outputs of IE evaluations. In order to capture more widely useful relations, MUC-7 introduced the template relation task. The template relation task requires the identification of a small number of possible relations between the template elements identified in the template element task. This might be, for example, an employee relationship between a person and a company, a family relationship between two persons, or a subsidiary relationship between two companies. Extraction of relations among entities is a central feature of almost any information extraction task, although the possibilities in real-world extraction tasks are endless [12]. It finds relations between template element entities. It is all about what relationships between entities there are.

The line between template entity and template relation is somewhat indistinct as both identify information relating to entities found by named entity recognition. What separates them is the domain of the application: Template relation needs relations between entities, with both the relation and entity types being relevant to the application domain. Template element needs additional information about entities, which may involve other entities but this data is mainly used to enrich the description of the entity. Figure 2.7 shows Template Element Relation from Figure 2.4.

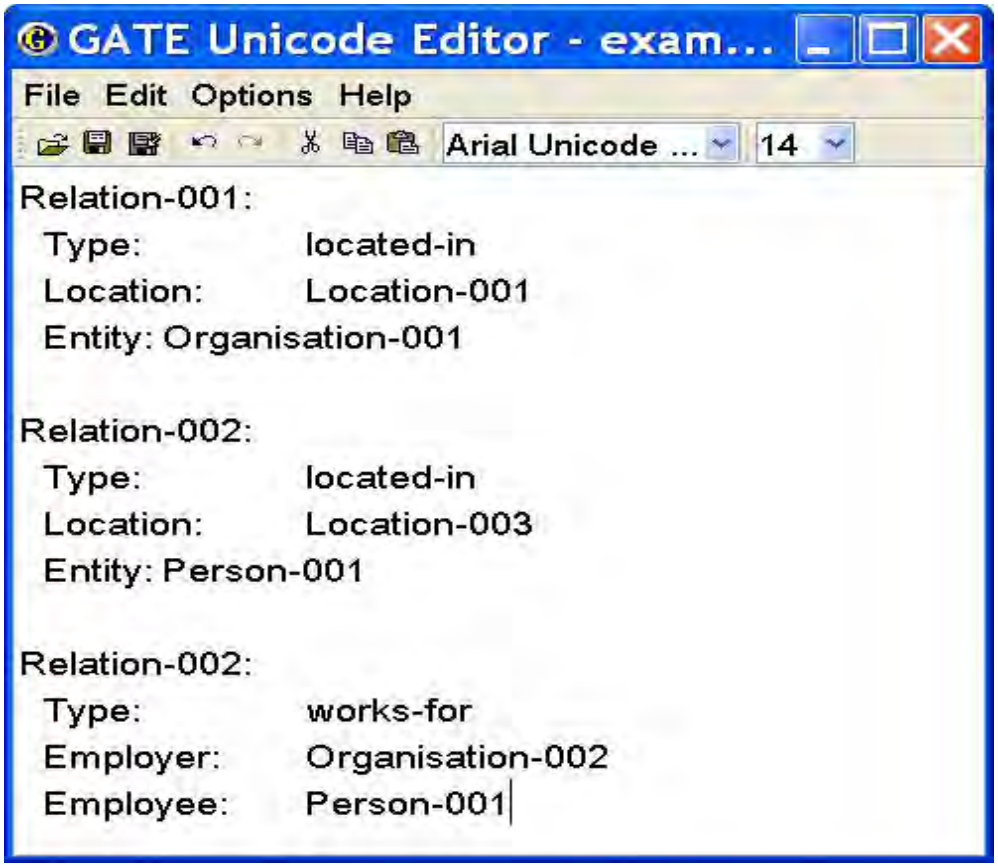

Figure 2.7: Template Relation Construction

2.4.5 Scenario Template Production

Scenario templates are the prototypical outputs of IE systems, being the original task for which the term was used. They tie together template element construction entities and template relation construction relations into event descriptions. Scenario template is a difficult IE task; the best

MUC systems score around 60%. The human score can be as low as around 80%, which illustrates the complexity involved [12].

The scenario template production task is both domain dependent and, by definition, tied to the scenarios of interest to the users. Note however that the results of named entity, template relation and template element feed into scenario template.

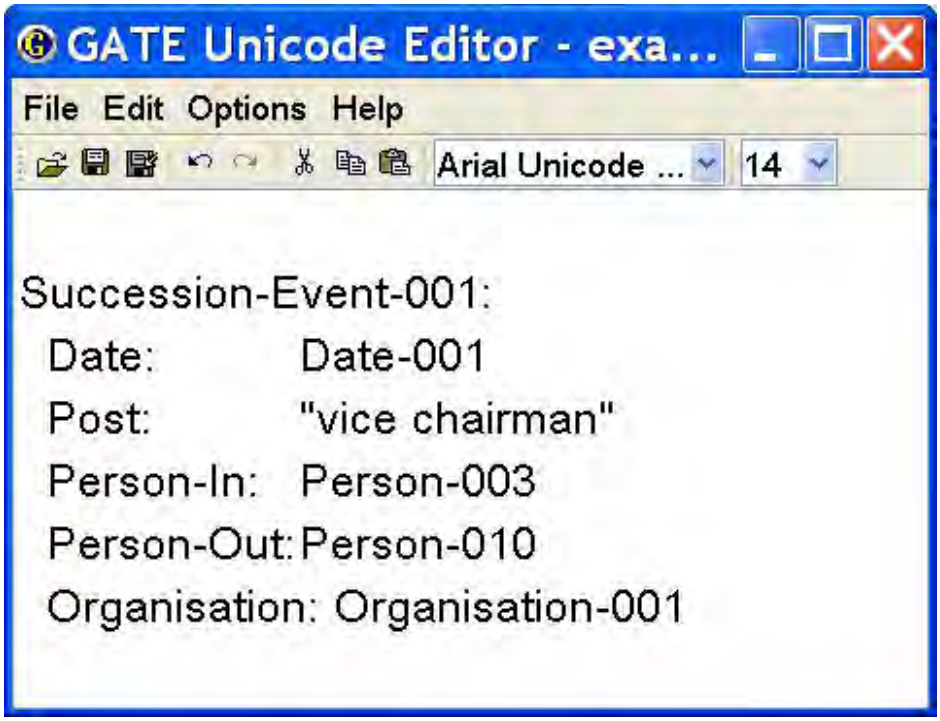

Figure 2.8: Scenario Template Construction

2.5 Data Sources

Information extraction system needs sufficient data to perform analysis. If the data source is not large enough or not representative of the domain, IE systems cannot achieve good performance. Usually, IE systems use different methods on different kinds of data. For the better performance of information extraction system, the accuracy of the training data also critical. According to Eikvil [23], the source of data in information extraction can be categorized in to three. These are free text (unstructured), semi- structured and structured.

2.5.1 Free or Unstructured Text

Free text (Unstructured text) is just narrative text without explicit formatting. The sources may include newswire reports, newspapers, journal articles, electronic mail, etc. Originally the aim of information extraction was to develop practical systems which could take short natural language texts and extract a limited range of key pieces of information from them. For example the texts might be news articles about terrorist attacks and the key information might be the perpetrators their affiliation the location of the victims etc. [22].

Since there is no well-defined structured, managing free text is very complex. An IE system for free text has generally used natural language techniques and the extraction rules are typically based on patterns involving syntactic relations between words or semantic classes of words. Several steps are required including syntactic analysis, semantic tagging, recognizers for domain objects such as person and company names and extraction rules. The rules or patterns can be hand-coded or generated from training examples annotated with the right label by a human expert. The state-of-the-art in information extraction from free text is not comparable to human capability but still provides useful results. This is true whether the rules are hand coded or automatically learned [22]. Compared to hand coded results, the performance of automatic IE systems for unstructured text is poor due to the fact that narrative text is often very complex. However, IE systems can still provide useful outcomes on narrative text largely because they depend on specified features that can have regular structure.

2.5.2 Semi structured text

Semi structured data are an intermediate point between unstructured collections of textual documents and fully structured tuples of typed data. Such texts fall between structured and free text and have previously been almost inaccessible to IE systems. Semi structured text is ungrammatical and often telegraphic in style and does not follow any rigid format [22]. Semi structured text sometimes does not contain full sentences either. Semi-structured text has a format in some sense, but the structure of the format is imprecise compared to structured text.

Various NLP techniques were deployed to design rules for extraction of information from free text. However these methods which are appropriate for grammatical text will usually not work for semi structured text which seldom contains full sentences. Hence for semi structured texts the traditional techniques of IE cannot be used and at the same time simple rules used for rigidly structured text will not be sufficient.

2.5.3 Structured Text

Structured text is defined as textual information in a database or file following a predefined and strict format. Such information can easily be correctly extracted using the format description. Usually quite simple techniques are sufficient for extracting information from text provided that the format is known otherwise the format must be learned [22]. Structured texts are presented in a table or database schema; therefore, it is simple to extract the relevant one compared to free text or semi-structured text sources. Since a computer can more easily understand structured text presented in a database schema, IE research involving structured text is not as prevalent as research on unstructured and semi-structured text.

The data we have used for training and testing our proposed model was collected from Walta Information Center. The data was from infrastructural news domain and are declarative type. From the above discussed data source types it will classified as free or unstructured type.

2.6 Approaches to Information Extraction

There are two main approaches to the design of information extraction systems which can be called:

- knowledge engineering approach and
- Machine learning (automatic training) approach

2.6.1 Knowledge engineering approach

In the knowledge engineering approach grammars expressing rules for the system are constructed by hand using knowledge of the application domain. A person who creates such a

type of system, or is responsible for writing those rules (i.e., a knowledge engineer) must be an expert in the knowledge domain chosen for extraction or at least must be closely familiar with it. In addition to requiring skill and detailed knowledge on a particular IE system, the knowledge engineering approach usually requires a lot of labor, a lengthy test-and-debug cycle, and it is dependent on having linguistic resources at hand, such as appropriate lexicons.

Knowledge engineering approach is a time-consuming task to build the rules and the system is also hard to maintain. However, most of the best performing systems are hand crafted. In this approach, the computer system does not learn anything from the data. It only implements what human experts have learned. According to Appelt and Israel [12] knowledge engineering approach is a very important factor in creating a system with a high level of performance. High level performance achieved because of every aspect of the knowledge will be built by knowledge engineer. Building a system using this approach involves iterative process. Firstly, the knowledge engineer writes a particular rule. Then he applies it to the available texts and checks whether it works correctly or not. Modifications are done if needed and the rule is examined again until a desirable result is achieved. In some case it is termed as rule-based approach, since it involves writing rules.

2.6.2 Machine learning approach

In Machine learning approach there is no need of building extraction rules manually. Therefore, a person who is responsible for the information extraction process does not have to know how to write rules and how a system works. A machine learning algorithm implemented in the information extraction system creates those rules. In order to do that the algorithm must have access to a large number of training texts related to the chosen domain. Since machine learning approach learns and works according to training data, for better performance large amount of corpus will be used to train the system. This approach sometimes called automatic training approach. Rather than focusing on producing rules, the automatic training approach focuses on the training data [23].

Developing a system with machine learning approach is relatively faster than the knowledge engineering approach but requires that a sufficient volume of training data must be available. In

this approach the same machine learning algorithm can be applied to different domains as long as corpora of domain-related texts are available. Therefore, unlike knowledge engineering approach, machine learning approach is domain independent. On the basis of analyzing the benefits and drawbacks of both approaches it is possible to conclude with the criteria which determine the choice of one of them. The most important condition to choose the automatic training approach is the presence of a set of suitable texts which can be used to train the algorithm. In the case of the knowledge engineering approach the availability of a person who is experienced in writing extraction rules is the most crucial criterion. Automatic learning systems can be categorized in to three groups: supervised learning systems, semi-supervised learning systems and unsupervised learning systems.

Supervised Learning

Supervised learning uses training data to induce extraction rules. Thereby, almost no knowledge about the domain is necessary, but a large set of training data has to be annotated according to the underlying structure of information to be extracted. The main bottleneck of supervised IE systems is the preparation of the training data. Most systems need a large amount of annotated documents for a particular extraction task, which also leads to the lack of portability of an IE system. However supervised learning is known to be dependent on the availability of a large amount of manually prepared training data. Even though supervised learning saves human expert time, it is still a time-consuming task to prepare the training data manually.

Semi-supervised Learning

To deal with the still heavy reliance on human expertise in supervised learning, another method termed semi-supervised learning is employed in IE systems. The work presented at [24] describes semi-supervised learning as follows:

"Using semi-supervised learning, a system learns from a mixture of labeled (annotated) and unlabeled data. In many applications, there is a small labeled data set together with a huge unlabeled data set. It is not good to use only the small labeled data set to train the system because it is well known that when the ratio of the number of training samples to the number of feature measurements is small, the training result is not accurate. Therefore, the system needs to

combine labeled and unlabeled data during training to improve performance. The unlabeled data can be used for density estimation or preprocessing of the labeled data, such as detecting inherent structure in the domain. In other words, the system extracts patterns from the annotated data, and labels the un annotated data automatically using the patterns. As a result, all data are labeled for the training."

Semi-supervised learning saves human effort while the performance can be as good as the performance of a supervised learning technique.

Unsupervised Learning

Unsupervised Learning systems reduce the burden of the user to require only a statement of the required information. No extraction patterns are given in advance by the user. In this learning an annotated corpus is not used to improve the system's level of performance. The main challenge is to realize the user's need into a set of the extraction patterns. The systems are based on bootstrapping methods, expanding an initial small set of extraction patterns.

In general, it is not necessary to create all the components of an information extraction system using only one particular approach. It is quite possible to interchange these two approaches while building different components of the system. One of the reasons of having such a possibility is that one can never say objectively which approach is better. Both of them have their advantages and disadvantages. As Appelt and Israel [12] stated, the systems which use a knowledge engineering approach show a higher performance compared to automatic training approach. However, they require a lot of effort and time and depend on the knowledge engineer's skills and experience and availability of linguistic resources. The very important advantage of a machine learning based system is that it can be transferred to a different domain easily as long as specific texts and a person who can annotate them are available. But sometimes those texts are problematic or expensive to obtain or there is a lack of useful documents on which an algorithm can learn, and manual (or even machine-aided) annotation on the scale needed to provide reasonable levels of performance may be expensive. So, choosing the right approach for certain extraction task will be depend on suitable conditions like knowledge engineers or enough training data.

2.7 Knowledge-poor Approach

Knowledge-poor approach falls under the category of knowledge engineering approaches which do not rely extensively on linguistic and domain knowledge. By knowledge-poor, it means that the specifications for new extractors are reasonably small and easy to write [53].

For the development of our proposed work we follow the knowledge-poor technique of knowledge engineering approach. In our case we have used rules for entities which have consistent pattern like person name which proceed by title. But these rules do not consider extensive analysis on the language and rather depend on low level linguistic knowledge. On the other hand we have used list of gazetteers for entities which does not have such kind of consistent pattern. In general we have used both rules and gazetteers for our development and we mentioned the approach as knowledge-poor approach.

2.8 Evaluation Matrices

The necessity for evaluation metrics for the information extraction problem came with the message understanding conferences. Although the event is called a "conference", it can be described with other words like "competition" between information extraction research groups or "evaluation" of their systems' performances. Thus, the major aim of these conferences was the evaluation of the state-of-the-art in the information extraction area, discovery and promotion of the new approaches in information extraction. The starting points for the development of these metrics were the standard IR metrics of recall and precision. Information extraction systems evaluation can also expressed on the notion of true and false positives and true and false negatives. It can be said that correctly extracted entities are true positives, whereas false positives are wrongly extracted information. Similarly, false negatives are relevant but not extracted information which is left in the text; true negatives are the information which is not extracted and not relevant to the task [23].

Precision (*P*) is the proportion of correctly extracted entities (*Ncorrect*) to the total number of extracted entities (*Nresponse*) (the ratio between number of needles in a hand and number of needles and straws in the hand). It refers to the reliability of the extracted information.

$$\mathbf{P} = \frac{\mathbf{N}_{\text{correct}}}{\mathbf{N}_{\text{response}}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.1)}$$

Recall (*R*) is the proportion of correctly extracted entities (*Ncorrect*) to the total number of entities which are extracted manually (*Nkey*) (the ratio between number of needles in the hand and total number of them in the haystack). In short it shows how much of the information that was correctly extracted.

$$\mathbf{R} = \frac{\mathbf{N}_{\text{correct}}}{\mathbf{N}_{\text{key}}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.2)}$$

When comparing the performance of different systems both precision and recall must be considered. However as it is not straightforward to compare the two parameters at the same time various combination methods have been proposed. Buckland and Gey [25] studied the relationship between precision and recall in the information retrieval field. However, their findings can be applied to the information extraction area as well. As they stated, recall can be described as the measure of extraction effectiveness, whereas precision is the measure of extraction purity. *F* measure combines precision P and recall R in a single measurement

F-measure: the F-measure has been defined as a weighted combination of Precision and Recall. It is the harmonic mean of two metrics which allows comparing and assessing different information extraction systems using one common base.

$$\mathbf{F} = \frac{(\beta^2+1)\,PR}{\beta^2 P+R} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.3)}$$

The parameter ß determines how much to favors recall over precision. When ß equals one, the weight of precision and recall are the same [22]. If ß is greater than one, then precision becomes more important than recall. On the other hand, if ß is less than one, then recall becomes more important than precision. Using the F-measure the relative performance of systems reporting different values for recall and precision can be easily compared.

2.9 The Amharic Language

Ethiopia is the second most populous African country and harbors more than 80 different languages. Three of these are dominant: Oromo, a Cushitic language spoken in the South and Central parts of the country and written using the Latin alphabet; Tigrinya, spoken in the North and in neighboring Eritrea; and Amharic, spoken in most parts of the country, but predominantly in the Eastern, Northern, and Central regions. Amharic and Tigrinya are Semitic and about as close to each other as are Spanish and Portuguese [26]. Despite its similarity with Tigrigna, Amharic also strongly influenced by the Cushitic languages, especially Oromo and the Agaw languages [27]. Both languages (Amharic and Tigrigna) are written using their own unique script, horizontally and left-to-right in contrast to many other Semitic languages.

Amharic is the second-most spoken Semitic language in the world, after Arabic and the official working language of the Federal Democratic Republic of Ethiopia and thus has official status nationwide. Following the Constitution drafted in 1993, Ethiopia is divided into nine independent regions, each with its own regional language. Then, Amharic become the official or working language of several of the states/regions within the federal system, including Amhara and the multi-ethnic Southern Nations, Nationalities and Peoples region. Outside Ethiopia, Amharic is the language of millions of emigrants (notably in Egypt, USA, Israel, and Sweden), and is spoken in Eritrea [28]. Despite its wide speaker population, Atelach and Asker [29] claim that the computational linguistic resources for Amharic are very limited and almost nonexistent.

2.9.1 The Amharic Writing

Amharic is written using a writing system called fidel - ፊደል ("alphabet", "letter", or "character") adapted from Ge'ez (the liturgical language of the Ethiopian Orthodox Church) language. There are 33 basic characters, each of which has seven forms called orders depending on which vowel is to be pronounced in the syllable. The seven orders were represent seven vowel sounds. Therefore, these 33 basic characters with their seven forms will give 7*33 syllable patterns (syllographs), or fidels. Amharic characters were represented by computer using Unicode. Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language [30]. Ethiopic characters (fidel - ፊደል) have more

than 380 Unicode representations including punctuations and special characters (U+1200-U+137F).

Two of the base forms (ኧ and ዐ) represent vowels in isolation, but the rest are for consonants (or semi-vowels classed as consonants) and thus correspond to Consonant Vowel (CV) pairs, with the first order being the base symbol with no explicit vowel indicator. The writing system also includes four (incomplete, five character) orders of labialized velars and 24 additional labialized consonants. In total, there are 275 fidels, but not all the letters of the Amharic script are strictly necessary for the pronunciation patterns of the spoken language; some were simply inherited from Ge'ez without having any semantic or phonetic distinction in modern Amharic. There are many cases where numerous symbols are used to denote a single phoneme, as well as words that have extremely different orthographic form and slightly distinct phonetics, but with the same meaning. Table 2.4 shows the orders of ሀ (h) and መ (m) in their Latin alphabets.

Table 2.4: The orders of ሀ(hä) and መ(mä)

Orders	1	2	3	4	5	6	7
V / C	/ä/	/u/	/i/	/a/	/e/	/ɨ/	/o/
H	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
M	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ

From the given table above, it is clear that the characters are formed having the seven vowel sounds. The seven vowels are አ፣ ኡ፣ ኢ፣ ኣ፣ ኤ፣ እ፣ ኦ or ዐ፣ዑ፣ ዒ ፣ዓ ፣ዔ፣ ዕ፣ ዖ which are different in character and similar sound.

Amharic took the whole Geez alphabet and uses it in the Amharic writing system. The Amharic alphabet does not have capital and lower case distinctions. It then added some more symbols for some other sounds that it has and that could not be represented by the symbols of the Geez alphabet. A list of the Amharic alphabets (called fidel - ፊደል) with its orders is shown in appendix A.

2.9.2 Grammatical Arrangement

According to Baye [31], the Amharic language has been declared to have word categories as ስም (noun), ግስ (verb), ቅፅል (adjective), ተውሳከ ግስ (Adverb), and መስተዋድድ (preposition).

Noun: are words that are used to name or identify any of categories of things, people, animal, places or ideas or a particular of one of these entities. A word will be categorized as a noun, if it can be pluralized by adding the suffix **ኦች/ዎች** and used as nominating something like person, animal, and so on [31]. In Amharic sentences noun is used as to indicate subject of a sentence. Pronouns were considered as separate word category in the earlier works of linguists now considered as nouns. Pronoun is a word that is used instead of a noun or noun phrase. They are characterized based on number, gender and possessiveness. Some of pronouns for deictic specifier such as **ይ ይህ እሰዋ እኔ አንቺ እሱ እነሱ** ..... ; Quantity specifiers such as **አንዳንድ ጥቂት ብዙ** .... and possession specifier such as **የእሱ የእኔ የእነሱ** ..... .

Verb: any word which can be placed at the end of a sentence and which can accept suffixes as /ህ/,/ሁ/,/ሽ/, etc. which is used to indicate masculine, feminine, and plurality is classified as a verb. Verb expresses accomplishment of an action and used to close the sentence. For example in a sentence "**ተሰማ ከባህርዳር መጣ**" the word "**መጣ**" is verb since it appears at end of the sentence and closes the meaning of sentence.

Adjective: Adjectives in a sentence modify nouns to denote quality of a thing; that is, it specifies to what extent a thing is as distinct from something else. It will come before a noun to qualify a noun with some form of size, kind and behavior. For example in the sentence "**ትልቅ ፈረስ**" the word "**ትልቅ**" used to qualify the size of the noun **ፈረስ**.

Adverb: it will be used to qualify a verb by adding extra idea on the sentence. Adverbs usually precede the verbs they modify or describe. The followings are adverbs Amharic **ትናንት፣ ገና፣ ዛሬ፥ ቶሎ፣ ምንኛ፣ ክፉኛ፣ ጅልኛ፣ ግምኛ** and **አደገኛ**.

Preposition: preposition is a word which can be placed before a noun and perform adverbial operations related to place, time, cause and so on; which can't accept any suffix or prefix; and which is never used to create a new word. Prepositions could not have any meaning alone but

they will have meaning only when they are attached or used together with other words such as nouns and pronouns. For example in the sentence **"ያዕቆብ ከልጆቹ ጋር ወደ ግብፅ ሄደ"** words **ከ፣ጋር፣ወደ** are prepositions. Some of prepositions include **ከ፣ ለ፣ ወደ፣ ስለ፣ እንደ፣ጋር**….

2.9.3 Amharic Punctuation marks and Numerals

In Amharic, there are different punctuation marks used for different purposes. In the old scripture, a colon (two dots ፡) has been used to separate two words. These days the two dots are replaced with whitespace. An end of a statement is marked with four dots (**አ ራት ነ ጥብ ።**) while **ነጠላ ሰ ረ ዝ (፣ or ፥)** is used to separate lists or ideas just like the comma in English and **(ድርብ ሰረዝ ፤)** is used as a semicolon in English. The question and exclamation marks have recently been included in Amharic writing system.

The Amharic number system consists of twenty single characters which are Geez numbers. They represent numbers one to ten, multiples of ten (twenty to ninety), hundred, and thousand. These characters are derived from Greek letters and in order to make them look like the Amharic characters the symbols are modified by adding a horizontal stroke above and below. The system has no place value and there is not symbol representing the number zero (0). In addition, the number system does not use commas or decimal points. These situations make arithmetic computation using this system very complicated.

2.9.4 Amharic Sentences

A sentence, in every language, is a group(s) of word(s) that comply with the grammatical arrangement of the language and capable of conveying meaningful message to the audience. Baye [31] categorizes Amharic sentences into simple and complex. A simple sentence is a complete structure that can convey a complete idea. Complex sentences are used to convey complete ideas like simple sentences but unlike simple sentences they are composed of complex phrases. Complex phrases are phrases that have complements and/or modifiers that are sentences themselves.

A sentence in Amharic can be a statement which is used to declare, explain, or discuss an issue. The combination of phrases to create another phrase that can express a full idea on something is

a sentence. When Amharic sentence is viewed from grammatical structure point of view it is a combination of noun phrase and verb phrase. The noun phrase comes first and then the verb phrase. The noun phrase and the verb phrase further will be divided to different particles such as other sub noun phrase and verb phrase, noun, adjectives, specifier and so on. Amharic sentences are in the order of Subject- Object -Verb.

There are different types of sentences such as interrogative sentences which are used for questioning, exclamatory sentence which is used for emphasis and emotion. Most of the sentences that we will use for extracting information will be declarative sentences. Declarative sentence is a sentence that is used to express the physical, psychological, imaginary or real events. It is main objective is description of some issue. The news articles use the declarative sentences for expressing the details different issues.

2.9.5 Problems in the Amharic Writing System

One of the problems in Amharic writing is the redundancy of symbols used with the same pronunciation. Although in the Ge'ez language, these different symbols give each word different meanings, in the Amharic language they have been used interchangeably. Since there is clear distinction in Amharic characters with the same sound, it makes the Amharic writing system more difficult. Table 2.5 shows an example of the character redundancy where more than one symbol is used for same sound.

Table 2.5: Amharic characters with the same sound

Consonants	Other symbols with the same sound
ሀ (hä)	ሃ ሐ ሓ and ኀ
ሰ (sä)	ሠ
አ (ä)	ኣ ዐ and ዓ
ጸ (tsä)	ፀ

These redundancies of characters will affect the extraction process. For example if we the word "አለም" it can be represented as ዐለም፣ ኣለም or ዓለም. Therefore, during the pre-processing stage of Amharic documents for this research, the different forms of a character that have the same sound are changed to one common form.

The other inconsistency will be observed regarding the representation of compound words. Some compound words are used as a single word in some instances (either by fusing the two words or by inserting a hyphen between them) and as two separate words at other instances. For instance, if we take "ቤተ-ክርስቲያን" which means church can be written as ቤተ ክርስቲያን or ቤተክርስቲያን. This happened to be inconsistent in Amharic texts and should be considered in the extraction process.

The other issues are related with word loan form other languages and abbreviations. When loan words are translated to Amharic it may written differently. The cause of the difference in the Amharic spellings of these foreign language words seems to be the difference in the pronunciations of these words. For example, the word "Sport" may be translated as ስፖርት or እስፖርት. On the other hand Amharic writings with abbreviations also create inconsistency in automatic extraction process. For instance, the phrase AD can be written in the text as ዓ.ም or ዓመተ ምህረት. However, in text processing tasks such words should come into one common form since they are representing the same meaning.

Summary

This chapter briefs the general concepts of IE and different approaches and evaluation metrics. The focus of IE is extracting relevant set of a document in a structured representation using predefined templates. At this end it uses either machine learning mechanism or rule based approach. Accordingly we choose the knowledge-poor approach which is knowledge poor technique of rule based approach. The most important task during information extraction is named entity recognition. Named entity recognition employs the recognition of person name, organization, numeric value, location name and etc. Amharic is one of widely used language in Ethiopia. Information extraction in Amharic language involves declarative sentences since it describe different entities about a given news.

CHAPTER THREE: RELATED WORK

In this chapter, we present some works done so far on information extraction. Among them we have chosen the most relevant ones which are related our works that are done on different languages.

3.1 Information Extraction from English Text

The research work presented in [32] proposes a method for protein name extraction from biological texts. Their method exploits handcrafted rules based on heuristics and a set of protein names using dictionary. Their method used 1,745 annotated protein names, most protein name fragments are nouns (85%), and thus Part Of Speech (POS) taggers are unlikely to be helpful to distinguish protein names from numerous nouns. Therefore, in this method unlike previous works they avoid the use of NLP tools such as POS taggers and syntactic parsers, which are computationally costly. Additionally, they complementarily make use of a protein name dictionary to raise the coverage.

In their core part they implement hand crafted rules and a dictionary of protein names. In the hand crafted rule they design extracting rules based on the surface clues associated with protein names. But all proteins may be extracted based on the rule, so protein name dictionary takes the advantage of protein names extraction which the rules cannot identify. They evaluate their system in terms of accuracy, generalizability and processing speed. As common to other extraction system accuracy is measured in terms of precision, recall and F-measure, generalizability was used to detect whether the system achieve constant performance even on different corpora or not. In measuring processing speed the result showed that since it was not incorporate NLP tools, such as POS taggers and syntactic parsers it was comparably faster. The authors from the result conclude that their system produces outcome comparable to the state-of-the-art protein name extraction system on multiple corpora.

The other work presented in [33] uses hierarchical Hidden Makrov Model (HMM) to extract information from scientific literature. Hierarchical HMMs have multiple levels of states which

describe input sequences at different levels of granularity. In this model, the top level of the HMMs represent sentences at the level of phrases, and the lower level of the HMMs represent sentences at the level of individual words. In this model the input representation for all sentences being processed is hierarchical, their models represent the shallow phrase structure of sentences, they focus on learning to extract relations rather than entities, they used null models to represent sentences that do not describe relations of interest, and they used a discriminative training procedure.

The proposed work is based on the hypothesis that incorporating sentence structure into the learned models will provide better extraction accuracy. Their approach is based on using syntactic parses of all sentences to be processed. In this case they conduct the shallow parser and construct parse tree. Then the resultant tree will be used in constructing the input representation of hierarchical HMMs. They have evaluated their approach in the context of learning information extraction models to extract instances of three biomedical relations from the abstracts of scientific articles. These experiments demonstrate that incorporating a hierarchical representation of grammatical structure improves extraction accuracy in HMMs. In their conclusion they raised that using an approach that takes advantage of grammatical information represented at multiple scales and their approach generalizes to additional levels of description of the input text as main contribution.

3.2 Information Extraction from Thai Text

In the work presented [34], they made an effort to extract information about plant-disease from Thai agricultural document. Most of elements in the document are name entity (e.g. the names of plant, disease, pathogen and chemicals etc.), so Thai named entity recognition was a crucial module in the extraction process. They also implement co-reference resolution and discourse analysis. The data source they have used was semi-structured style, a document describes about one plant and it has plant name in title.

In the main focus of the extraction task of their work they have defined two types of information that they want to extract. The first one is Entity Information which consists of plant, disease and cause; which are relevant named entity elements recognized by named entity recognition. The

other one is Explanation Information which extracts symptom and treatment. Explanation Information is information that cannot be explained by only name entity, so they have to extract a set of sentences that explain about the topic and use discourse analysis technique to combine them together to be one unit.

In their implementation they resolve co-reference resolution by dividing the module into three. The first module was name entity co-reference resolution and solved by matching name entities that reference to the same entity. The second module was noun phrase co-reference resolution which was solved by matching noun phrase that refer to name entity or reference to same entity and the last one was Zero pronouns referent resolution that match each zero pronoun to appropriate entities. Then, they perform selection of a sentence which contains relevant topic of interest. They identify a paragraph as relevant paragraph if a paragraph contains relevant sentences of interested topic. They were extract relevant elements by using information from document context and document structure. They were used semi-structured documents as a data source; the paragraphs were classified by topic. Each paragraph can have one topic or more, whereas one topic may has many paragraphs that scatter the entire document. In such cases they used discourse analysis technique to catch the salience from paragraphs that have same topic and combine them to be one unit for fill in concern slot.

The researchers conclude solving co-reference resolution using discourse analysis technique to combine the scattered sentences to be one unit of explanation information makes extracted information to be interpreted easier and time consumption is reduced for user to consume all information. They also plan to extend explanation information to be represented in script based format, in order to have more systematic interpretation of extracted information.

3.3 Information Extraction from Chinese Text

The work presented as [35] extract information from Chinese free text based on automatic pattern rule learning in combined with heuristic information. In this work the researchers present two main contributions. First they proposed a methodology of automatic pattern rule learning for Chinese free text. Second a new approach to employ heuristic information based on context

information is presented. And the work they presented combines extensive use of extraction patterns with heuristic information.

In the development of the extraction system they have passed several steps as information extraction required. In their document preprocessing step; first, input document is broken into sections or sentences based on layout cues. Second, they perform word segmentation by looking up dictionary because Chinese sentence is composed of a sequence of characters without any natural delimiters such as spaces between words. After preprocessing they conduct syntactic analysis which is used to categorize un-categorized words during preprocessing step using HMM model probability approach. Then they identify noun phrase and verb phrase. Then they resolved pronominal anaphora resolution in order to understand whether the extracted contents in some instances that pronouns are contained without their antecedents or not. They used both rule based and statistical approach in resolving pronominal anaphora resolution. For their extraction task they used information extraction engine which was operated on semantic pattern rule. But pattern rule cannot solve all types of system. So, they employ heuristic rule. The heuristic information is utilized to make one complex text into simple form from the given sentence. It will be simplify the structure of the text so that the pattern rules can be used to extract the information. Thus, during extraction procedure, instead of using one complex rule that extracts all template slots from a text, they took heuristic information which minimizes complexity.

Their work was used 50 articles that are selected from web pages were used for the pattern extraction rule learning and other 100 articles from the same pages were used as test data for the information extraction. The evaluation is based on the template whose slots are person name, organization name, old position, and new position. During their test they apply method 1and method 2. Here method 1 is based on pattern matching without heuristic information; method 2 is based on pattern matching with heuristic information. Precision and recall value of their evaluation results given is below.

Table 3.1: Precision and Recall of different slot

Slot & result		**Method 1**	**Method 2**
Person Name	Recall	64.1%	78.8%
	Precision	89.2%	87.2%
Organization	Recall	62.3%	76.5%
	Precision	92.1%	89.3%
Old Position	Recall	64.5%	77.4%
	Precision	86.3%	84.6%
New Position	Recall	68.3%	83.3%
	Precision	84.5%	81.3%

The researchers finally conclude that integrating heuristics information and pattern rule learning improves the extraction result and increases recall.

3.4 Information Extraction from Portuguese Text

The work presented in [36] deals on information extraction from Portuguese medical discharge letters. On this work they used 915 free text discharge letters written in Portuguese from the Infante D. Pedro Hospital in Aveiro, Portug for training, validation and evaluation of their system (MedAlert). The medical information extractor called MedAlert they have developed was based on rule based and machine-learning mechanism.

The MedAlert have components like Ingestion, general natural language processing and name entity recognition module. Ingestion is a component which reads the patient discharge letters XML files and converts them into plain text while keeping information about the document's structure. General natural language processing is the one used for NLP related tasks like sentence discovery, tokenization, part-of-speech tagging and shallow parsing. The other component is named entity recognizer which identifies the concepts defined in the MedAlert discharge letters representation model based on specified terminology. The evaluation result show MedAlert's precision value (0.69), which they consider to be the most important characteristic of systems, intended to be used in clinical domains.

The other work presented in [37] used machine learning approach to extract temporal information from Portuguese text. Temporal and event processing is a task of recognizing

temporal expressions and analyzing events. During their extraction they annotate raw text with temporal information with event terms, temporal expressions and temporal relations. They trained the extractor with subset contains 68,351 words, 6,790 events, 1,244 temporal expressions and 5,781 temporal relations. In this work there are two important aspects which are event identification and temporal expression identification. For event identification they used a classifier that automatically classify as a given word denotes an event or not. But they claim that this strategy is not very efficient, since some very frequent words cannot possibly denote events. The other aspect is in temporal expression identification. In these cases they trained the classifier to understand time related values.

3.5 Information Extraction from French Text

Kamel Nebhi [38] presented a rule based approach in automatic ontology based information extraction from French news text articles. Their system establishes relation between named entities in a text, the ontological standardized semantic content of the DBpedia ontology and the DBpedia databank. Ontology based information extraction is different from traditional IE because it finds type of extracted entity by linking it to its semantic description in the formal ontology. The DBpedia ontology is a shallow, cross-domain ontology, which has been manually created based on the Wikipedia projects. The result of evaluation for person, organization and location entity categories achieved 91% of traditional F-measure and 94% of an augmented F-measure.

The other researchers presented in [39] proposed medication extraction system from French Clinical texts. The authors here presented the implementation of a medication extraction system which extracts drugs and related information from French clinical texts, on the basis of an approach initially designed for English. The corpora they used in this experiment consist of a total of 17,412 French data from the cardiology unit of a French University Hospital out of each 50 of them was used as test data. They used rule based learning approach in the medication extraction. They prepared a set of lexicons to define the relevant vocabulary, and a set of extraction rules encode the grammar of medication expressions.

The system start its operation by segmenting the text into sentences based on typographical clues. Then since the drugs are in the list of lexicons it will operate lexicon lookup to extract and recognize drug names. The extraction rules they used in this system is almost the exact copy of that of English rules. They transposed their program from English to French using the same definition of the items to be extracted, namely the following six types of information: drug name, dosage, mode, frequency, duration and reason. Their system relies on specialized lexicons and a set of extraction rules. They evaluate the system with 50 annotated texts and obtain 86.7% F-measure, a level higher than the original English system. Finally they conclude that the same rule based approach can be applied to English and French languages, with a similar level of performance.

3.6 Information Extraction from Spanish Text

The research work done on extracting information from natural disaster news reports from Spanish text presented in [40] used machine learning approach. The researchers followed the linear separator approach for learning patterns in the sentence of the text. The linear separator approach is mainly based on the hypothesis that looking at the word combinations around the relevant data is enough for learning the required extraction patterns. Its main advantage, compared to other methods from the same approach, is that it does not require applying a deep linguistic analysis of texts in order to generate the classification features.

The system has two modules namely text filtering and fact extraction. The purpose of text filtering module is to select the documents about natural disasters from a set of news reports. In particular, this module considers the classification of news reports in six different categories, one for each type of natural disaster (hurricanes, forest fires, inundations, droughts, and earthquakes) and other one for non-relevant documents. The purpose of fact extraction module is to extract the relevant data from the selected news reports. The design of fact extraction module is supported on the idea that looking at the word combinations around the interesting data is enough to decide about its category. This module considers two processes. The first one is identification candidate text which is responsible for extracting all text segments that have some possibility for being part of the extraction template such as proper names, quantities and dates. The other is selection of

relevant information which is used to extract the text segments relevant for a predefined output template.

The authors evaluate their work using a corpus of Spanish news reports about natural disasters. The experimental results on a collection of Spanish news show the effectiveness of the proposed system for detecting relevant documents about natural disasters was reaching an F-measure of 98% and for extracting relevant facts to be inserted into a given database was reaching an F-measure of 76%. From their work the authors conclude that although their method applies traditional machine learning techniques, it differs from other previous information extraction systems in that it does not depend on sophisticated resources for natural language processing.

3.7 Information Extraction from Amharic Text

The other thesis work by Ibrahim presented in [9] uses Hidden Makrov Model to extract information from Amharic text. The proposed information extraction system was developed for extracting information from a single sentence. It uses the slots subject, object, action and reporter and tries to extract information from the news text which contain the above listed four slots on a single sentence. If one of these components does not exist in the sentence the system does not extract the information. This works only takes a single sentence as input and produce extracted entities. But a single sentence can't convey full information about certain event or action. Therefore, our proposed work will fill the gap by considering multiple sentences.

A research work by Getasew [8] developed information extraction model from Amharic news text using classification machine learning approach. The proposed model has document preprocessing, text categorization, learning and extraction and post processing as its main components. In this section we made extensive review on Getasew's work and each his model components discussed separately as follow.

Document preprocessing

The document preprocessing component handles language specific features like normalization of characters, normalization of numbers, tokenization and number prefix separator. In his tokenization he considers Amharic punctuation and chopping into words and character

normalization involves changing Amharic character with the same sound into one canonical form. The purpose of his number prefix separator is to stem the prefix from the number and to consider the prefix and the number as independent tokens. He also employ number normalization which is used to normalize all the numbers in the Amharic news texts which are represented using digit or Amharic language characters or a combination of both in to their equivalent number representation.

Text Categorization

The text categorization component of his system is used to categorize the news text as investment, infrastructure or others categories. The development process of this involves training data preparation, training a classifier model and using the trained classifier model for text categorization.

The training data that is used for text categorization is economy news category which was obtained from Ethiopian News Agency (ENA). He prepared the training data manually by selecting all the news texts and storing them in the text file format. Then, all the text file format news texts are organized in three different folders which have the same name to the category of the news texts which are investment, infrastructure, and others. For the text categorization subcomponent he adopted the work of Yohannes Afework [54] which was done on text categorization using Weka. In data preparation he employs feature selection to reduce the dimension of the data by selecting features from the original words of the news text. During feature selection he used stop word removal, stemmer, number prefix separator, name prefix separator and name remover.

After categorization data preparation using language specific developed tools, the next step is to preprocess it using Weka to make the data ready for training the classifier and using the model for prediction at later stage. Then he train the classifier model which later will be used to predict the category of unseen news texts. For training a classifier he used Weka open source machine learning algorithm. Among the different Weka algorithms he used Naive Bayes, SMO and Decision tree. At the end he used the generated classifier model for predicting the category of the unseen news text.

Learning and Extraction

The learning and extraction component main task is to extract candidate texts and train and use the classifier model for predicting the category of the extracted candidate texts. In the learning phase the predefined attributes that he set to extract from infrastructure news are infrastructure name, Place where the infrastructure is built, the amount of money used to build the infrastructure, the source of money for the infrastructure development, the number of users which will be benefited from the infrastructure, the person who give the information to the news agency.

After the classifier model is generated using the training data the next step is to use the trained classifier model to work on the unseen news text. To use the classifier model for extraction he identified candidate texts. Accordingly names and numbers were identified as candidate text. The name of a person is a candidate text for Reporter attribute, place name is for place the infrastructure is built on, the number in the news text is considered as a candidate text for number of users and the for the amount of money spent for the infrastructure development, organization names are used as candidate texts for attribute financial source for infrastructure development and names used for infrastructure are the candidate texts. The candidate text selection was done using the Gazetteer list which comprises of the different names under consideration. The gazetteer which consists of names for different places in Ethiopia, different names that can be used for identification of persons, the different infrastructure names, and the different governmental and nongovernmental organization list is used.

Once the candidate texts are selected and tagged, the feature extractors then extracts all the features from the tagged candidate texts and store it in the database for later processing by Weka. After the features are extracted and preprocessed using Weka the trained classifier model is used to predict the category of the candidate text. Among the candidate texts those with the token category of the predefined attributes will be stored in the database and others which are not under the category of the predefined attributes will be discarded.

The data that the author used to train and test their proposed system was obtained from ENA. These Amharic news texts are categorized under 16 main categories manually which are law and justice, health, events directory, international relations, social affairs, culture, politics, agriculture,

defense and security, science and technology, sport, education, economy, accident, weather and other classes. Among the different categories from ENA news the economy category was used as a data source for the training and testing of the proposed system. Then various evaluation techniques, which were used to evaluate the performance of the classifier machine learning algorithms, were used for IE and text categorization. Among the different classifier machine learning algorithms used for text categorization component, the Naive Bayes algorithm performs by correctly classifying 92.83% of the 1200 news texts used as a dataset. On the other hand, 1422 instances are used for training and testing the information extraction component. Different scenarios are used to evaluate the role of the different features in predicting the category for the candidate texts. Among the different scenarios they considered and the different machine learning algorithms they employed the SMO algorithm correctly classified 94.58% of the instances correctly, when all the features are considered which yields higher precision and recall rate for the different attributes considered for extraction.

Critics

We have made extensive review on information extraction systems from Amharic language text particularly Getasew's [8] work. From the review his main objective is categorization of candidate text which he termed as information extraction. In candidate text identification he used list of gazetteer which contain different texts of predefined candidate texts. Then he categorizes those candidate texts using his trained classifier. As a critique, at the first place categorizing candidate text is not information extraction and he used gazetteer for all list of candidate text and his extraction cannot works beyond entities among the list.

Summary

In this chapter we have presented a review on a number of IE works which was done on different languages. In our review we have emphasized on algorithms, feature sets, corpus size, approaches and performance of those works. In the general NLP process rule based approach gives better result than machine learning approach. Rule based approach is preferable for a condition with limited corpus size. We have also critically review on Amharic information extraction systems and identify the gap.

CHAPTER FOUR: DESIGN AND IMPLEMENTATION

4.1 Introduction

This section preset the detail description of design and implementation of the proposed information extraction system. The architecture of this thesis work is similar with most of the automatic information extraction systems. We should note that information extraction can be applied to spoken language i.e. audio files. In this thesis however, we shall focus solely on text processing applications. A text file is simply a data structure consisting of alphanumeric and special characters. As NLP task the system involves preprocessing which are common for any kind of information extraction task by which the proposed architecture also shares it.

We have used infrastructural news data as a domain because of the nature of the data is factual. The news data were taken from Walta Information Center. The proposed system was implemented using GATE (General Architecture for Text Engineering). GATE is a language-engineering environment developed by the University of Sheffield. Since its first release in 1996, this tool was used in a large number of IE applications and gained wide acceptance by scientific community [47, 48]. GATE is distributed with an information extraction component set called ANNIE (A Nearly-New Information Extraction) system. ANNIE is provided as part of GATE [49] which is an architecture, framework and development environment for language processing research and development. ANNIE consists of the following set of modules: tokenizer, sentence splitter, POS tagger, gazetteer, finite state transducer, orthomatcher, and pronominal coreference resolution. ANNIE is suitable for extracting information from English text. But one can use ANNIE for other language as needed by re-write the JAPE (Java Annotation Patterns Engine) grammar and modifying gazetteer and additional language resources. We used ANNIE as a good base to build our proposed system.

4.2 System Architecture

The general architecture of an IE system was defined by Hobbs [43] in MUC-5 as "a cascade of transducers or modules that, at each step, add structure to the documents and, sometimes, filter relevant information, by means of applying rules". Most current systems follow this general architecture, although specific systems are characterized by their own set of modules, and most of the architectures are currently being developed. In general as discussed in [12, 43, 55], the combination of such modules allows some of the following functionalities to a greater or lesser degree:

- Document preprocessing: to handle language specific issues and to make the data ready for extraction
- Syntactic parsing, full or partial: This is to identify word groups which are either nouns groups, or verbs groups and identify level of importance.
- Semantic interpretation: To generate either a logical form or a partial template from a parsed sentence.
- Discourse analysis: This is to link related semantic interpretations among sentences. This is done using anaphora resolution, and other kinds of semantic inferences.
- Output template generation: for the translation of the final interpretations into the desired format.

The architecture given in Figure 4.1 named Amharic Information Extractor (AIE) partially adopted from the generalized information extraction architecture given in [55]. The AIE model contains components which are specific to the Amharic language. In general the AIE model is categorized into three phases. These are preprocessing, extraction and post processing. The general view of the architecture is show below.

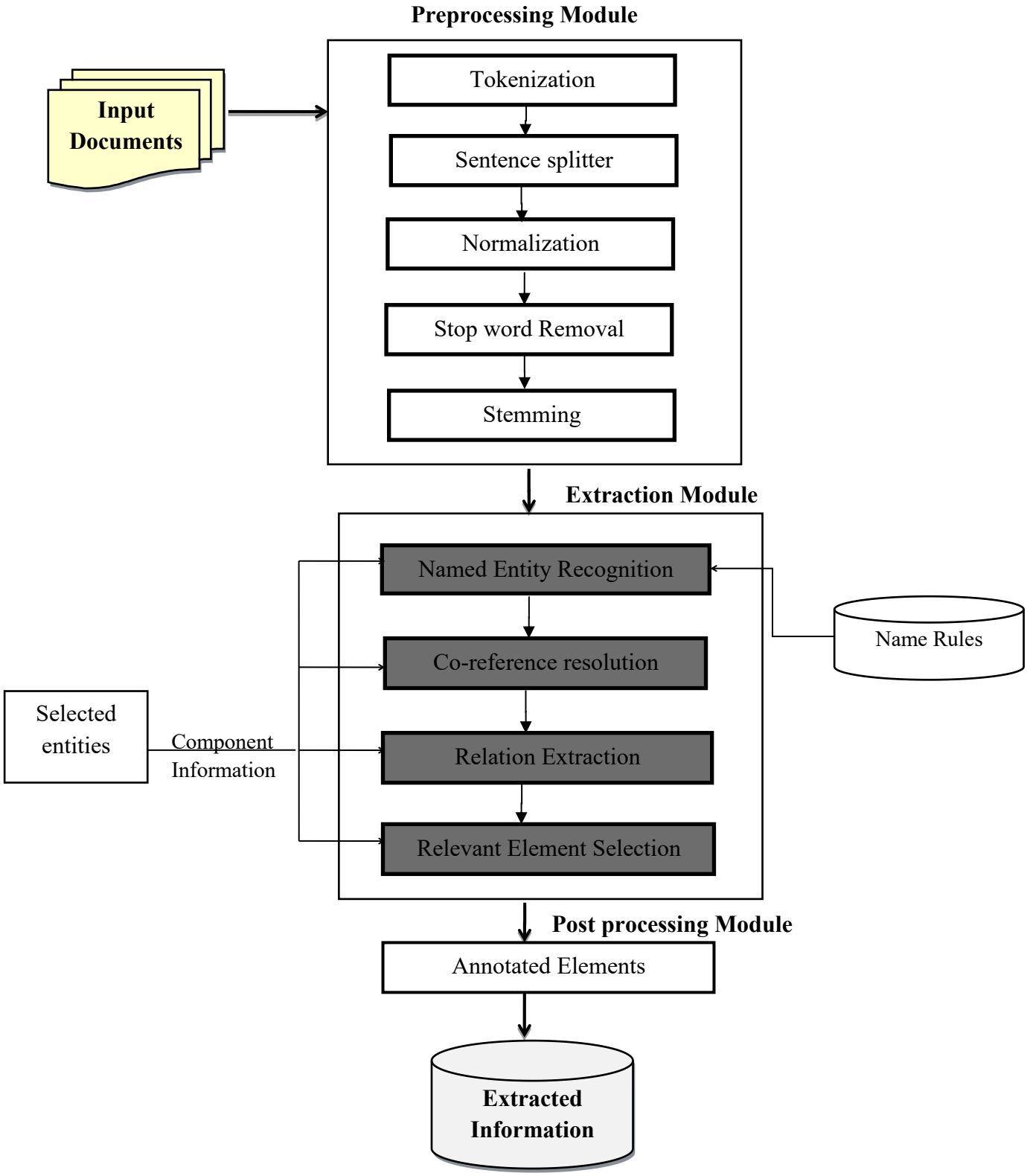

Figure 4.1: Architecture of AIE

4.3 Preprocessing Module

Preprocessing is the task that is used to make the data ready for further processing. It is about data cleaning and preparation for extraction and it is the primary step of information extraction system. The main role of preprocessing is formatting or converting the input documents, so that later extracting can be performed easily. The document preprocessing component handles the different language specific issues that are imposed by the nature of the language to make the data ready for remaining phases. In order to get good results, language dependent text preprocessing should be performed before automatic extraction is implemented. Text or document preprocessing is the step by which the text is made comfortable to the learning algorithm. The preprocessing step is simply a removal of non-informative words or characters from the text. Thus, this section presents the detail how each module was developed.

Tokenization

Tokenization is a task of splitting a text into pieces called tokens, which are disjoint and meaning full texts. It manipulates the text on the level of individual words. Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation. A token is an instance of a sequence of characters in some particular document that are grouped together as a useful semantic unit for processing. As described in [41], tokenizing is often considered to be part of the text preprocessing, which also includes removal of markup tags and excessive whitespace characters.

In Amharic a very simple and obvious way to split a text into tokens is to split at certain tokens, like whitespace or other punctuation characters. The challenge here in tokenization process is when there are compound words and special characters. For example if we take the word **ቤተ-እስራኤል** it has to take as one word as it is. Tokenization is based upon a set of rules: that is on the left hand side we have the pattern to be matched and on the right hand side we have the actions to be taken up. This activity reads a sequence of characters as a string and tokenizes them using predefined list of delimiters such as new lines and space.

In this thesis the tokenization component uses the ። (**አራት ነጥብ**) the Amharic full stop and ፣ (**ነጠላ ሰረዝ**) the Amharic comma as the most commonly used punctuation mark in the news texts. The ። (Amharic full stop) is used for identifying the sentence demarcation and ፣ is used to separate different text segments which mostly are related. In the old scripture two dots **: (ሁለት ነጥብ)** were used to separate words and now it is replaced by white space. Thus, the tokenization process in our system uses Amharic punctuation marks and white spaces for token identification. It also considers abbreviation and concatenated words. The sentence in the Figure 4.2 uses space and tokenizes the Amharic sentence as given below.

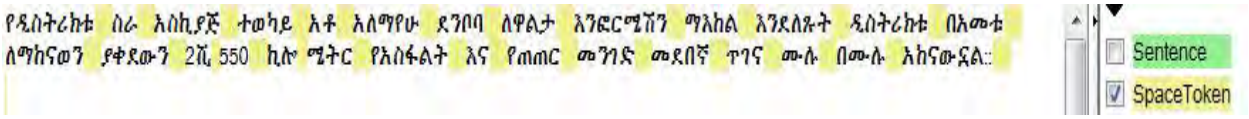

Figure 4.2: Tokenizer processing result

Start

Input text document

detect punctuation symbol and white space ()

If the word is abbreviated or concatenated

Ignore the split

else

Split the document in to words()

Create array of a words()

Output

list of words

Stop

Figure 4.3: Tokenization Algorithm

Sentence splitter

A sentence splitter divides a spawn of text into sentences. In Amharic a question mark and Amharic full stop (**::**) are used to end a sentence. Sentence splitting involves the identification of sentences and words of the document to be extracted. The extraction mechanism demands sentence identification since a single sentence could convey a message. Therefore, we have used Amharic full stop (**አራት ነጥብ** ።) and the usual question mark as sentence demarcation. Figure 5.4 show a demarcated sentence using Amharic full stop.

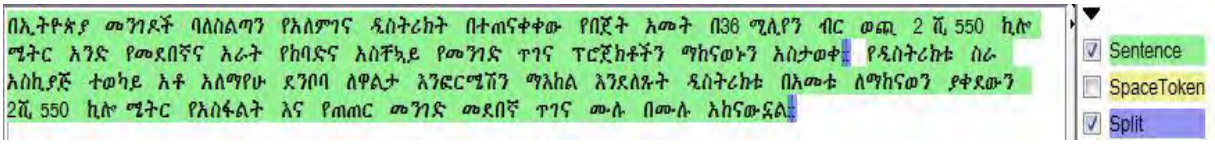

Figure 4.4: Sentence splitter processing result

Normalization

Normalization is the process of transforming text into a single canonical form that it might not have had before. Normalizing text before storing or processing it allows for separation of concerns, since input is guaranteed to be consistent before operations are performed on it. Text normalization requires being aware of what type of text is to be normalized and how it is to be processed afterwards; there is no all-purpose normalization procedure.

In Amharic writing system there are characters with the same pronunciation but different symbols which are called homophones. The letters such as **አ**, **ኣ**, **ዐ** and **ዓ**; **ሠ** and **ሰ**; **ሀ**, **ኀ**, **ሃ**, **ኻ**, **ሓ**, **ኃ** and **ሐ**, **ጸ** and **ፀ** are examples of characters with the same meaning and pronunciation but different symbol. For example if we take the word "**Habtamu**" **ሀብታሙ** it could have different forms like **ሐብታሙ**, **ሃብታሙ**, **ሓብታሙ** **ኀብታሙ** and **ኃብታሙ**. Therefore, all the above different forms must be normalized into **ሀብታሙ** by changing the first character of a word. Therefore, these characters should be normalized. The other normalization issue is related with short hand representation of words like **አ/አ**, **ት/ሚ** and **ጽ/ቤት**. Therefore, these characters should be converted into their expanded long form. Appendix C present list of abbreviations and their expanded form taken from Getasew's [8] work.

In this thesis the normalization component of the preprocessing step handles the problem of Amharic text writing and considers abbreviation writing. The main function of the component is to replace the alphabets that have the same pronunciation and use with one of the alphabets. For example, if the character was one of ሃ ኅ ኃ ሐ or ሓ then it was converted to ሀ. Algorithm for character normalization is given below.

Start
From a document read a character
If the character is one of ሃ፣ ኅ፣ ኃ፣ ሐ ፣ ሓ and their orders
    replace them with ሀ
If the character is ሠ and its orders
    replace it with ሰ
If the character is one ዐ and its orders
    replace it with አ
If the character is ፀ and its orders
    replace it with ጸ
If the character is ፁ
    replace it with ፉ
If the character is ዉ
    replace it with ው
End

Figure 4.5: Normalization algorithm

Stop word Removal

Since information extraction is used to extract relevant words from large collection of document, words like ነው and ነበር does not affect extraction process and have no role on result of extraction. 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.

Most news specific stop-words such as **አመልክቷል, አስታውቋል, ገልጸዋል,** … and **ታውቋል,** are considered.

Hence, news specific common words of this type were used as a stop word list. In this thesis, we have used news specific stop-words which are taken from Melese's work [46]. See Appendix B for the list of stop-words.

Start

Input document

For each sentence in the document

If a word in each sentence is stop word (any occurrence of **ነበሩ፣ ናቸው ሆኗል** ..,.)

Remove the word from the sentence

End for

Output list of sentences without stop word

End

Figure 4.6: Stop word Removal algorithm

Stemming

In linguistic morphology and information retrieval, stemming is the process for reducing inflected (or sometimes derived) words to their stem, base or root form, generally a written word form. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root. Stemming reduces words to their root word so that different variations of the root word will be matched to the root word during retrieving relevant word for the extraction process. In general stemming is the process of reducing morphological variants of a word into a common form.

Amharic is morphologically rich language. For morphologically less complex languages like English, this usually involves removal of suffixes. For languages like Amharic that have a much

richer morphology, this process also involves dealing with prefixes, infixes and derivatives in addition to the suffixes [29]. Therefore, the stemmer is responsible for changing the word variation into its root form. For example **ፍቅራችን, በፍቅራችን** and **ስለፍቅር** will change to their stem word **ፍቅር.** Normally proper names, dates, and numbers should not be subjected to stemming since they will not be reduced to root words. In this thesis, we employ the stemming algorithm developed in [45]. The stemming algorithm developed in [45] removes those affixes that are usually used for changing the tense, number, gender and case of a word.

4.4 Extraction Module

This component is the main unit of the Amharic text information extraction system. This component used to identify entities and categorize them. Entity recognition is based on selected ontologies. Selected ontologies are constructed elements that determine the type of entity (person name, organization, currency and location) to be extracted. The module of information extraction component is discussed below.

4.4.1 Named Entity Recognition

It is an information extraction task aimed at identifying and classifying words of a sentence, a paragraph or a document into predefined categories of named entities. The idea of Named Entity Recognition (NER) is identifying named entities like people, place, date, number and etc. NER task can additionally include extracting descriptive information from the text about the detected entities through filling of a small-scale template. For example, in the case of persons, it may include extracting the title, position, nationality, sex, and other attributes of the person.

In this thesis we have developed rules and gazetteers that can identify the different named entity classes. After identifying named entities, the next task will be tagging the appropriate named entity (person, organization, time etc.). Generally we have identified four named entity groups based on our domain. These are PERSON which refers name of a person who gives the news to the reporter, ORGANIZATION which refers news agency organization and source of budget organization, LOCATION which refers the place where the construction takes place and NUMERIC which refers monetary values for budget and length value for constructed road. We

have developed rules and gazetteers for the identification of these named entities. We have used infrastructure news as a domain. From this news we have selected road construction news and the following entities will be particularly extracted. Let's define what these entities are as given below:

- Source of news [**የዜና ምንጭ**]: This refers the name of the organization (media organization)
- Who gives the news [**ዜናውን የሰጠው ሰው**]: This refers name of the person who gives the information for the media journalist
- Infrastructure type [**መሰረተ ልማቱ**]: refers the type of infrastructure constructed.
- Constructed road (in km) [**የተሰራው መንገድ**]: This about how many kilometer road is being constructed.
- Place [**የተሰራበት ቦታ**]: refers place name where the construction takes place.
- Buget [**በጀት**]: This is refers currency allocated for the construction
- Source of Budget [**የበጀት ምንጭ**]: refers either governmental of non- governmental organization who allocates the budget for the project.

Thus, the purpose of our named entity recognition is annotating those seven entity types given above. In the extraction we use rules for entities which can have patterns and we developed gazetteers for entities which does not have any pattern and context to recognize.

Rules for Person Name Recognition

Person name in the Amharic language is very difficult to recognize and extracting from the text as the proper names do not start with a capital letter as is the case in many other languages. This makes Amharic named entity recognition difficult. The paper in [56] used rule based approach for Arabic proper name identification and develop rules based on a keyword set and located proper name by referencing the keyword. The authors keyword set includes titles, stop words and other key verbs. In this work we adopt the idea in Amharic person name identification. In most articles person names may appear in any position within the sentence and they are written next to a title. Hence we have identified a set of rules to deal with all these cases.

Table 4.1: An example show position of person name next to title

Amharic sentence fragment	English translation
ፕሬዝዳንት ሙላቱ ተሾመ እንደገለፀት	President Mulatu Teshome Expresses

Hence, in this case the development of the title list plays a central role in the development of the rules and is added to the GATE system. Rules are based on the position of title word. Based on this positional rule the algorithm for person named identification is given as follow. List of person titles taken from Getasew's [8] work were presented in Appendix D.

Start

Read a word w from a sentence

IF w belongs to Title list

THEN Tag the next word as PERSON

ELSE Do nothing

Output PERSON tagged sentence

End

Figure 4.7: Algorithm Person name identification

Organization, Infrastructure Type and Location Recognition

Since those entities cannot have any fixed pattern, recognition of such named entities will be identified based on the given list which contains list of organization, infrastructure type and location. Accordingly we have collected names of organizations which include news agencies, governmental organizations and non-governmental organizations. Infrastructural lists also contain a list of maintained and constructed roads. In the location list we have identified the different place names of all regions, cities, towns and woredas. Therefore, the recognition will be by inspecting those identified list.

Numeric Value Recognition

In our case numeric expressions, which are subdivided into simple numeric value expressions for measuring the coverage of the constructed road in KM (**250 ኪ.ሜ, 250 km**) and money expressions (**25 ሚሊዮን ብር, 25 million Birr**).

በኢትዮጵያ መንገዶች ባለስልጣን የአለምገና ዲስትሪክት በተጠናቀቀው የበጀት አመት በ36 ሚሊየን ብር ወጪ

From the sentence fragment above, numeric value representing money will be expressed using alphabetic and numeric value. The best way to recognize is by using money representing coefficient called (ብር) Birr. The term Birr (ብር) is used to represent Ethiopian currency. In the majority of financial articles every mention about money consists of two parts - amount and currency. Therefore, finding a number and a currency will give us a piece of monetary information.

```
Macro: Number

{Token.string == "ሺህ"} |

{Token.string == "ሚሊየን"}|

{Token.string == "ቢሊየን"}

Rule: MyCurrencyCatagory

( {Number}({SpaceToken})*)

{Lookup.majorType==”currency_unit”}

):label

:label.MyCurrency={type=”Money”, value=:label.Number.value}
```

Figure 4.8: Currency Jape Rule

በኢትዮጵያ መንገዶች ባለስልጣን የአለምገና ዲስትሪክት በተጠናቀቀው የበጀት አመት በ36 ሚሊየን ብር ወጪ 2 ሺ 550 ኪሎ ሜትር አንድ የመደበኛና አራት የከባድና አስቸኳይ የመንገድ ጥገና ፕሮጀክቶችን ማከናወኑን አስታወቀ ::

The processing result of the above sentence also has the same pattern with the previous one in representing numeric value. The keyword we have used to identify the coverage of constructed road is (ኪሎ ሜትር) kilo meter.

4.4.2 Coreference Resolution

Any given entity in a text can be referred to several times and every time it might be referred differently. In order to identify all the ways used to name that entity throughout the document anaphora and coreference resolution is performed. In this task application relevant entities will be referred to in many different ways throughout a given text and thus, success on the IE task was, to a least some extent, conditional on success at determining when one noun phrase referred to the very same entity as another noun phrase. In what is perhaps the simplest kind of case, this might mean recognizing full identity of strings. Coreference resolution is the stage when for noun phrases it is determined if they refer to the same entity or not. It is about the identification of multiple (co-referring) mentions of the same entity in the text [44, 12].

Coreference resolution was performed in order to identify whether a given noun phrase refers to the same entity or not. There are several types of coreference, but the most common types are pronominal and proper names coreference, when a noun is replaced by a pronoun in the first case and by another noun or a noun phrase in the second one. The coreference resolution task tries to identify equivalence between the entities that were recognized in the named entity recognition phase. In this work we implement coreference resolution using orthographic matching of strings by the GATE component orthomatcher. This module adds identity relations between named entities. It does not find new named entities such, but it may assign a type to an unclassified proper name, using the type of a matching name. The matching rules are only invoked if the names being compared are both of the same type. For example let's see the following sentence in Figure 4.11.

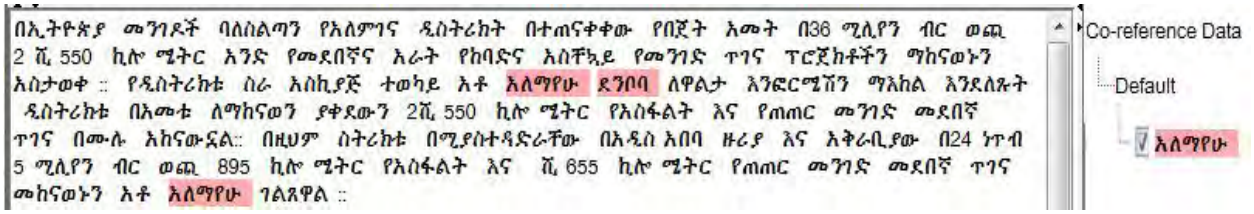

Figure 4.9: Orthomatcher processing result

The named entity **አለማየሁ** mentioned two times as shown above. Therefore, the orthomatcher display it as a coreference data.

4.4.3 Relation Extraction

This process is realized by creating and applying extraction rules which specify different patterns. The text is matched against those patterns and if a match is found the element of the text is labeled and later extracted. The formalism of writing those extraction rules differs from one information extraction system to another. Relation extraction is the task of detecting and classifying predefined relationships between entities identified in text and ideally identifying ho did what to whom, when, where, through what methods (instruments), and why [42]. In this module we want to extract surface-level relations that hold between the extracted named entities. In a sentence, two named entities must be related. Therefore, this phase will identify the relation between two pair of named entities in a sentence. The following figure shows relation between named entities.

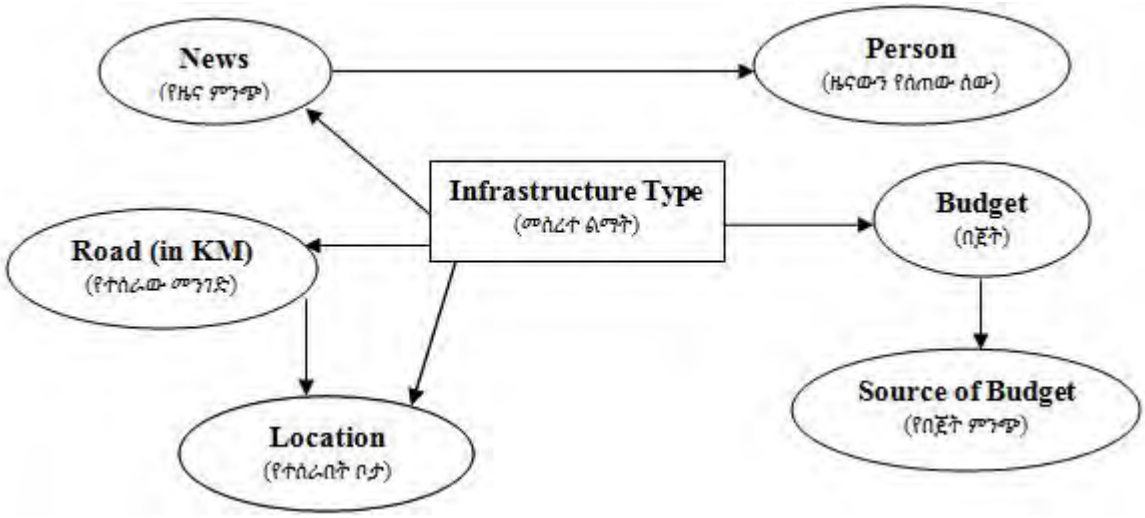

Figure 4.10: Relation between named entities

```
Start

Read named entities
```

```
Put all named entities in the array
For each named entity n
For each named entity m
If there is relation(n,m) or relation(m,n)
Add relation (n,m)
Else
Continue with m
End
```

Figure 4.11: Relation extraction algorithm

4.4.4 Relevant Element Selection

This component used to select relevant entities to be extracted based. This is the final step of the extraction process. In this module we have presented the identified entities using annotations. Generally person name, news source, the type of infrastructure, constructed road, location, budget and the source of budget were identified and annotated as relevant elements.

4.5 Post-processing Module

The last stage in AIE model is post-processing component. In this stage the different relevant entities that are extracted will be formatted and presented for the user. The main function of the post processing component is to arrange the format of the extracted data so that it will be flexible for data mining or any other application which want to use the data.

In the final stage of information extraction there is a task called template filling. This stage is fairly automatic, but may actually absorb a significant degree of effort to ensure that the correct formats are produced and that the strings from the original text are used. Extracted entities will be merged based on the constructed template. But our implementation tool provides a mechanism of representing extracted entities using annotations.

Hence, in this module we have presented the identified entities using annotations. Generally person name, news source, the type of infrastructure, constructed road, location, budget and the source of budget were identified and annotated as relevant elements. Therefore, we present entities using GATE annotation set. Accordingly, the following figure shows the extracted entities using color codes.

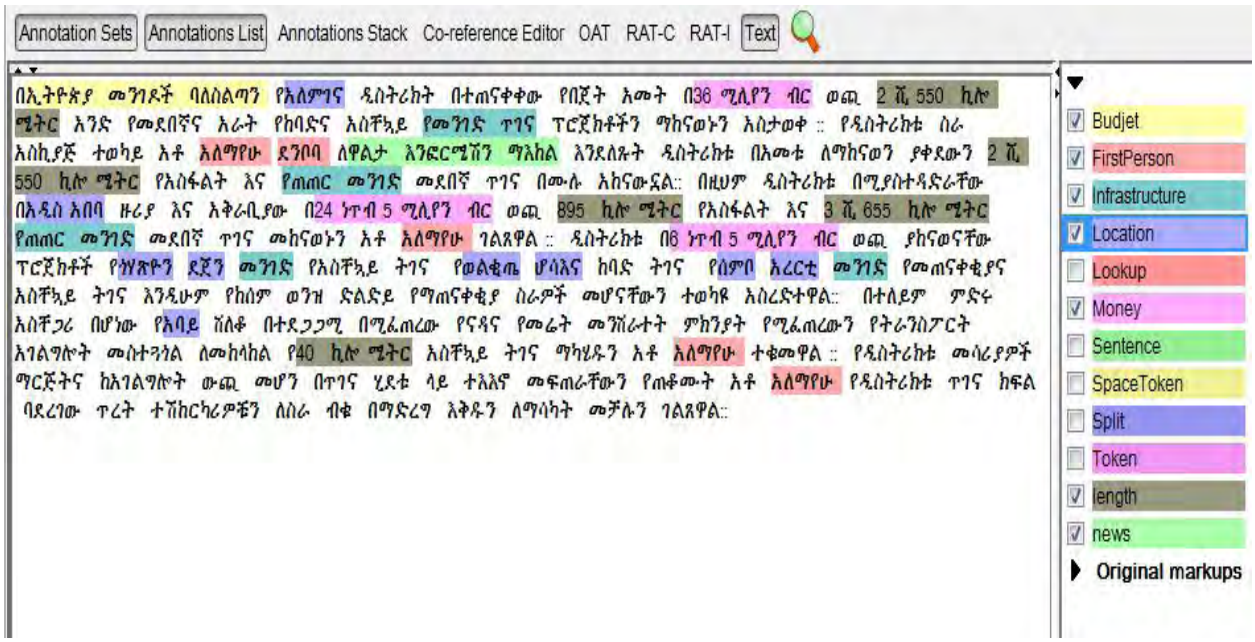

Figure 4.12: Result of Information Extraction using Annotation Sets

Summary

In this chapter design and implementation of AIE model was presented. The design and implementation consists of three main phases. The first phase discusses the document preprocessing which is vital for better extraction. The second phase deals on the main tasks of the model. This phase is extraction phase which is responsible for extraction entities and relation extraction. Then at the end selected entities were represented using annotated set.

CHAPTER FIVE: EXPERIMENT

This chapter discusses the experiment details of the developed system. In this section, we describe the data set, the evaluation metrics, and the results of the extraction system. Though evaluating the performance of the extraction system is an important part of the study, this was not a straightforward task. In the subsequent pages of this thesis, we describe the set of procedures that we used to conduct the experiment and the evaluation and its outcome is explained in detail.

5.1 Experimental Procedure

To evaluate information extraction systems, manually annotated documents have to be created. For domain-specific information extraction systems, the annotated documents have to come from target domain. Our proposed AIE system is used to extract infrastructure specific text only. Therefore, the following tasks are carried out in order to evaluate the extraction system we proposed.

For evaluation purpose, we have used The GATE Annie Diff tool. GATE Annie Diff provides a distinct evaluator to evaluate each task in the GATE. The evaluator requires a test data that is tagged in similar fashion to the training data. The general principle followed by the evaluator during performance evaluation is that, a manually tagged test data is supplied to the evaluator. Then the evaluator stores the tag of each token and the tokens are supplied to system. The tokens are then tagged by the system. When the tagging completes, the output is supplied back to the evaluator. The evaluator crosschecks the output generated by the system against the corresponding manual tags that are previously kept and computes the performance. Accordingly we have prepared test data which have similar manner with the training data and given to the evaluator.

Data Collection and Preparation

The data set we have used for training and testing is collected from Walta Information Center (WIC). WIC is a private media outlet established in 1994. Starting from the date of establishment WIC plays a significant role in minimize the gap of information flow in the country. The media center present Amharic news in different categories like law and justice, health, events directory, international relations, social affairs, culture, politics, agriculture, defense and security, science and technology, sport, education, infrastructure, accident, weather and other classes.

From this news category we have selected infrastructure news domain as a data source for training and testing of our proposed AIE system. The reason for selecting infrastructural news category, among the other categories is the availability of factual information which can be extracted and stored in the database. Since information extraction is annotation of relevant entities which mostly are names and numeric values taking infrastructural domain as a source is a right choice. From the infrastructure articles we have done inspection on the content selected articles which are not subjective type i.e. we have selected articles with more quantitative value and short which have eight sentences in average. Accordingly about 236 Amharic news articles were collected. In these articles we have about 1888 sentences and about 24760 tokens.

5.2 Evaluation Matrices

As we have discussed in chapter 2 of this paper the main metrics to evaluate rules quality are Precision, Recall and F measure. The first shows the system's accuracy, the second the coverage, and the third is the harmonic mean between the first two. In general terms, Recall is the ratio between textual elements correctly extracted by the system, and textual elements that are manually annotated. Precision is the ratio between items that have been correctly extracted and the total number of extracted items.

Our evaluation tool is Annie Diff and the resultant values displayed are: correct, partially correct, missing and spurious. Correctly extracted items are those whose system's result matches the manually annotated items. Partially correct mean when partial of the elements is recognized and for calculation purpose the coefficient 0.5 will be used [50]. If an element is recognized by the

system but at the same time is not annotated manually it is spurious. Missing values are entities which are found on the annotated but not extracted. The number of manually annotated elements includes all the items that the user wants the system to extract. The total number of the elements, in fact, extracted by the system is comprised of items that have been extracted correctly, incorrectly and spuriously. Hence, based on these values the formula of deriving precision and recall will be:

$$\text{Precision} = \frac{\text{Correct} + \frac{1}{2}\text{partial}}{\text{Correct} + \text{spurious} + \text{partial}} \quad \ldots\ldots\ldots\ldots (5.1)$$

$$\text{Recall} = \frac{\text{Correct} + \frac{1}{2}\text{partial}}{\text{Correct} + \text{missing} + \text{partial}} \quad \ldots\ldots\ldots\ldots\ldots (5.2)$$

F-measure (which is can also referred to as F1-measure) is the balanced harmonic mean of both precision and recall.

$$F1 = \frac{2(PR)}{P+R} \quad \ldots\ldots\ldots\ldots\ldots\ldots (5.3)$$

5.3 Performance Evaluation

In evaluation process we have identified Named entity recognition to be evaluated separately and at last the whole system will be evaluated

5.3.1 Evaluation of Named Entity Recognition

The evaluation was carried out using the Annie Diff Tool of GATE. This tool enables two versions of an annotated corpus to be compared, i.e. the annotations produced by the System, and the human annotations. For each annotation type, Precision, Recall and F-measure are calculated. The output of the tool is written as a table in an HTML file. We have measured the performance using both the training and evaluation data set.

Table 5.1: Performance achieved on the training set

Annotation Type	Correct	Partially Correct	Missing	Spurious	Precision	Recall	F-measure
News	61	3	7	1	0.962	0.880	0.919
Person	1791	20	39	14	0.987	0.974	0.980
Road in KM	1018	15	29	47	0.950	0.966	0.958
location	942	4	34	64	0.935	0.963	0.949
Budget	754	6	115	97	0.883	0.865	0.874
Source of Budget	50	11	9	12	0.760	0.793	0.776
Infrastructure Type	45	2	6	1	0.958	0.868	0.911
Average					0.919	0.901	0.910

Table 5.2: Performance Evaluation on Evaluation set

Annotation Type	Correct	Partially Correct	Missing	Spurious	Precision	Recall	F-measure
News	88	14	5	7	0.872	0.888	0.880
Person	2227	51	13	31	0.976	0.983	0.979
Road in KM	1066	16	31	57	0.943	0.965	0.954
location	1290	4	59	99	0.927	0.955	0.941
Budget	1239	56	217	95	0.912	0.838	0.873
Source of Budget	311	75	59	89	0.734	0.783	0.758
Infrastructure Type	61	12	3	8	0.827	0.882	0.854
Average					0.884	0.899	0.891

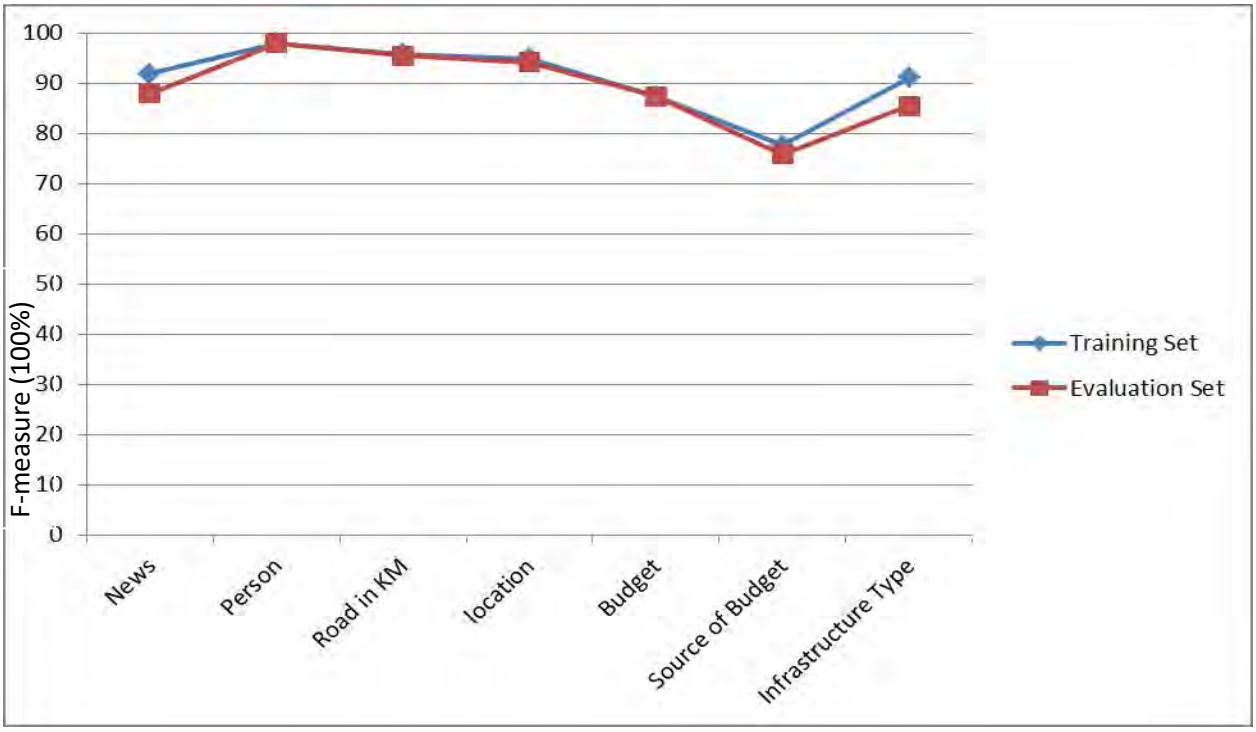

Figure 5.1: F1-measure for each NE type in the training set and evaluation set

5.3.2 Evaluation of Information Extraction

This component was evaluated based on manually annotated data in the same fashion with named entity recognition evaluation.

Table 5.3: Performance Evaluation on training data

Annotation Type	Correct	Partially Correct	Missing	Spurious	Precision	Recall	F-measure
News	79	15	7	1	0.911	0.856	0.883
Person	2791	20	39	14	0.992	0.983	0.987
Road in KM	938	15	29	47	0.946	0.963	0.954
Location	1183	4	34	64	0.947	0.971	0.959
Budget	861	34	15	59	0.920	0.965	0.942
Source of Budget	89	1	9	32	0.734	0.904	0.810
Infrastructure Type	65	5	4	2	0.938	0.912	0.925
Average					0.912	0.936	0.923

Table 5.4: Performance Evaluation on test data

Annotation Type	Correct	Partially Correct	Missing	Spurious	Precision	Recall	F-measure
News	58	19	5	7	0.804	0.823	0.813
Person	1987	69	13	21	0.973	0.977	0.975
Road in KM	419	32	13	7	0.950	0.938	0.944
location	1930	147	89	99	0.921	0.925	0.923
Budget	1279	57	33	72	0.929	0.955	0.942
Source of Budget	431	159	92	14	0.845	0.749	0.794
Infrastructure Type	25	1	3	2	0.911	0.879	0.895
Average					0.905	0.892	0.898

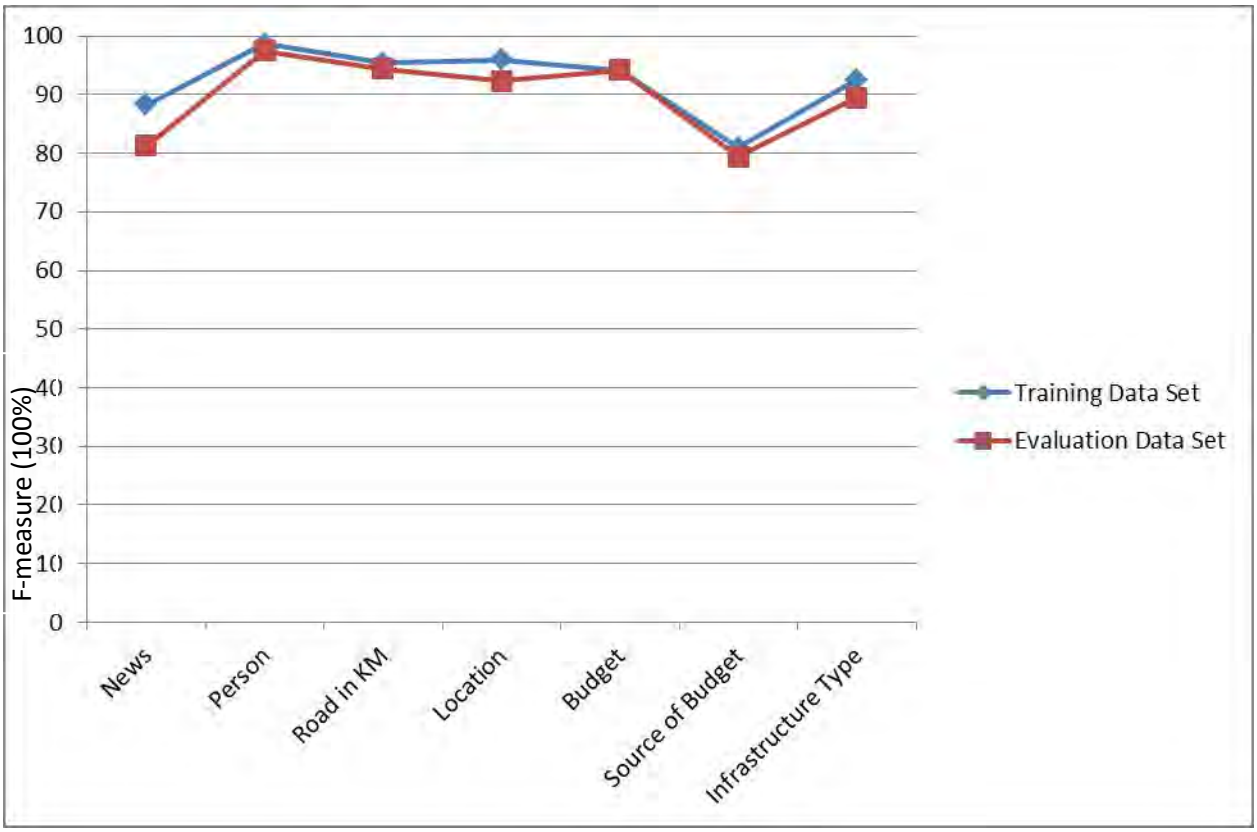

Figure 5.2: F1-measure for each extracted entities in the training set and evaluation set

Discussion on the Experiment

From the above tables the result was expressed as correct, partially correct, missing and spurious. Correct entities were extracted for annotation A of the response, if there exists an annotation B from the key set such that type A and type B equals. In the other case if the entities to be extracted fulfills the rule of extraction and it is relevant it will be extracted. Partially correct entities will be extracted if there are two annotations set which share the given entity and since the entity will be ambiguous there will be partial extraction. Partially correct responses are normally allocated a half weight. Missing entities extracted if the entity is not properly identified. We have used both rules and gazetteers in entity identification. In the case of rules missing will be occurred if the rule cannot address the entity expression. For example name of a person will be identified by title and if the data is without title missing will be occurred. In the case of gazetteer list if the entity is not found in list there will be missing. The other extracted result is spurious entities. They are false positives which are extracted but not relevant. False positives will be crated mostly due to dirty data and some ambiguous data. For example there is a word "ግድም" in our location gazetteer which is a place name around North Shoa and when evaluating the annotator annotates it as a location name but the word does not in the sense entity in the news article.

On the other hand the above tables and graphs show named entity recognition and information extraction results on the training and evaluation set. The result shows that the recognition of named entities whose grammars rely on gazetteer lists mostly becomes worse on the evaluation set. For example, the F-measure on the named entities "News" drops from 91.9% to 88%. On the other hand, named entities defined using regular expressions have the F-measure nearly constant (Person, Budget, and Road (in KM)). The result from the named entity recognition were influenced the information extraction component and the result is in the same fashion with the named entity recognition component.

5.4 Comparison with previous work

As discussed in the earlier sections, the work of Getasew [8] was carried out information extraction from Amharic news text based on categorization of selected candidate elements. His work also played on infrastructure news domain. Since both of the work has been done on the same language and the same domain, hence it is necessary to see the performance gap created among our system and the previous system.

In this comparison we evaluate only the domain aspect because we have different entities which make the comparison complex. On the other hand he test his work based on categorization method using Weka algorithms such as SMO, Naïve Bayes and Decision tree algorithms. In his evaluation the main focus of the evaluation of his IE component is to see the role of the different features that are considered and their effect in efficiently categorizing the candidate tokens as one of the predefined attributes. From his evaluation we observe that he conduct categorization of relevant entities and this is not the right information extraction evaluation. Therefore, in this comparison we focus only comparing the methods of extraction of two works based on the F-measure values given in Table 5.5.

Table 5.5: F-measure for different Annotations of the current work and the previous one.

Annotation	Our proposed work	Annotation	Previous work
	F-measure		F-measure
News	0.813		
Person	0.975	Reporter	0.953
Infrastructure Type	0.895	Infrastructure Name	0.939
Road in KM	0.944	Number of users	0.908
location	0.923	Place	0.951
Budget	0.942	Money Spent	0.981
Source of Budget	0.794	Financial Source	0.851

The above table shows the F-measure value of each system. This comparison is not conducted equally on two systems using the same data set. The F-measure value of on his annotation was directly taken from his paper. When look into each class; for example the annotation type “Person” and “Reporter” refers person name entity. In this we may say that our rule on person name identification better than the previous one. But we can’t generalize this concept. But our system handles the two important components of information tasks where the previous system does not have. These components are coreference resolution and relation extraction. Therefore, we can conclude that our work takes over the knowledge gap created on previous work by incorporating coreference resolution and relation extraction components.

CHAPTER SIX: CONCLUSION AND RECOMMENDATION

Today there are numerous electronic documents on different issues which can help the day to day life of individuals on our handheld devices as well as on the Internet. The increasing amount of text available online makes it necessary to find efficient ways to index and process texts automatically. Humans cannot process this mass of information and searching does not reduce its amount. Therefore, it is needed to extract the main information and summarize the available content. However, the availability of huge amount of information makes it difficult to manually search and acquire the required information from the ocean of unstructured data. The text data in local languages is also increasing from time to time. This is also true for Amharic as there is a growth in development and use of different online newspapers and contents. To alleviate this problem different research works have been conducted to extract relevant information automatically.

The existence of NLP discipline has enabled computers understand human languages and process them. Information extraction is one of the disciplines of NLP. Information extraction in general is a complex task. To extract the important information, it is necessary to identify and combine it with similar data found in other sources. The relationships between entities found in the text are analyzed and used to understand the key issues of the content. As a result of the huge information overload, information extraction becomes the focus of many researchers. Information extraction is the identification of instances of a particular class of events or relationships in a natural language text, and the extraction of the relevant ones. Hence, users will not be flooded with huge information, rather specific information (in the form of text, or sentence, or paragraph) will be returned. The document will be further chopped down in to pieces of factual information where that piece of information by itself is meaningful and capable of representing the document.

6.1 Conclusion

In this research work we have developed information extractor for Amharic language text. The system has three basic components developed using GATE. The first component is the preprocessing module, which resolves language specific issues and makes the data ready for extraction. The second one is the main unit of the extractor system which is called information extraction module. The extraction component is used to identify entity categories and will select the relevant one. Then the extracted entities were presented using annotations.

When testing our system, the system evaluation shows a promising performance. We have used 24760 tokens for training and testing and obtained 90.5% Recall, 89.2% Precision and 89.8% F1-measure. In general, given different constraints our algorithm obtained good performance compared with resource rich languages like English.

6.2 Contribution

The main contributions of the study are outlined below:

- The general architecture for information extractor from Amharic language text
- Algorithms are developed for language specific issues which handle normalization and tokenization.
- Rules for person name identification and coreference resolution
- Algorithm for relation extraction
- Implement Amharic language text extractor for infrastructure news
- Conduct experiment and come up with promising result.

6.3 Recommendation

Information extraction is probably a new study for Amharic language. The task is very complex for such under resourced languages. The developed Amharic information extraction system has portions that require further improvements that we want to recommend them as future works. The following are our recommendations for future works:

- The size of the training and test collection used in this research is too small. However, one can increase the data collection and improve performance.
- In this work only orthographic coreference were considered. Therefore, in the future nominal co-referencing will increase the performance of extraction and are highly recommended.
- Incorporating POS tagger and syntax parser will increase the performance of the extraction process. It is highly recommended incorporating POS tagger and syntax parser.
- For recognizing names the rules which depend on sentence pattern used. But sentence cannot always be in the same pattern. Therefore, using an automatic named entity recognition in later stages might minimize the burden of selecting the named entities
- Incorporating Amharic spell checkers to minimize the spelling problems which mostly happen in the news text might also have an impact as we manually modify the spelling errors as they have impact for named entity recognition.
- Incorporating Amharic word net for understanding the sense of words so that extraction will be better.

REFERENCES

[1] Lyman Peter and Hal Varian, "How much Information", Technical Report, School of Information Management and Systems, University of California at Berkeley, 2003.

[2] Nancy Chinchor, "MUC-7 Information Extraction Task Definition", http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/ie_task.html, last Visited November 5, 2013

[3] Jim Cowie and Yorick Wilks, "Information Extraction", Hand Book of Natural Language Processing, Editors: Robert Dale, Hermann Moisl and Harold Somers, ISBN: 0 – 8247 – 9000 – 6.

[4] Un Yong Nahm and Raymond Mooney, "A Mutually Beneficial Integration of Data Mining and Information Extraction", In the *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pp.627-632, Austin, 2001.

[5] Mihai Valentin Tablan, "Toward Portable Information Extraction", Department of Computer Science, the University of Sheffield, December 2009.

[6] "Information Extraction", http://gate.ac.uk/ie/, last visited October 31, 2013

[7] Riloff Ellen and Jeffrey Lorenzen, "Extraction-based Text Categorization: Generating Domain-specific role Relationships Automatically", *Kluwer Academic Publishers*, Dordrecht, the Netherlands, 1999.

[8] Getasew Tsedalu, "Information Extraction Model from Amharic News Texts", a Thesis Submitted to the School of Graduate Studies of Addis Ababa University in partial fulfillment for the Degree of Masters of Science in Computer Science, 2010.

[9] Ibrahim Yassin Hamid, "Automatic Information Extraction for Amharic Language Text Using Hidden Markov Model", a Thesis Submitted to Graduate School of Telecommunication and Information Technology in partial fulfillment for the Degree of Master of Science in Information Technology, 2009.

[10] C. R. Kothari, "Research Methodology: Methods and Techniques", 2nd edition, PP. 1-2, *New Age International Ltd.*, New Delhi, 2004.

[11] Jerry Hobbs and Ellen Rilo, “Handbook of Natural Language Processing: Information Extraction”, Machine Learning and Pattern Recognition Series, 2nd Edition, 2010.

[12] Douglas Appelt and David Israel, “Introduction to Information Extraction Technology”, a Tutorial Prepared for IJCAI-99, Artificial Intelligence Center, Menlo Park, CA.

[13] Ellen Riloff, “Information Extraction as a Stepping Stone toward Story Understanding”, Department of Computer Science, University of Utah, Salt Lake City.

[14] Sugato Basu, Arindam Banerjee and Raymond Mooney, "Semi supervised Clustering by Seeding", In the *Proceedings of the Nineteenth International Conference on Machine Learning*, 2002.

[15] Fotis Lazarinis, “Combining Information Retrieval with Information Extraction for Efficient Retrieval of Calls for Papers”, Department of Computing Science, University of Glasgow, Glasgow, Scotland.

[16] Dipanjan Das and Andre Martins, “a Survey on Automatic Text Summarization”, Language Technologies Institute, Carnegie Mellon University, November 21, 2007.

[17] Michael White, Tanya Korelsky, Claire Cardie, Vincent Ng, David Pierce, and Kiri Wagstaff, “Multi-document Summarization via Information Extraction”, *Proceedings of the First International Conference on Human Language Technology Research,* pp. 263-269, 2001.

[18] Dan Moldovan and Mihai Surdeanu, “On the Role of Information Retrieval and Information Extraction in Question Answering Systems”, Handbook of Information Extraction in the web Era, Springer edition, ISBN 3-540-40579-8.

[19] Rohini Srihari and Wei Li, “A Question Answering System Supported by Information Extraction”, *Proceedings of the sixth Conference on applied Natural Language Processing*, pp. 166-172, 2000.

[20] Ralph Grishman and Beth Sundheim, “Message Understanding Conference - 6: A Brief History”, *Proceeding of the 16th Conference on Computational Linguistics*, Volume 1, pp. 466-471, 1996.

[21] Hamish Cunningham, “Information Extraction – Automatic”, Department of Computer Science, University of Sheffield, Regent Court, 211, Portobello Street, Sheffield S1 4DP, UK.

[22] Line Eikvil, “Information Extraction from World Wide Web - A Survey”, July 1999.

[23] Madina Ipalakova, “Information Extraction”, A dissertation submitted to the University of Manchester for the Degree of Master of Science in the Faculty of Engineering and Physical Sciences, School of Computer Science, 2010

[24] Robert Gaizauskas and Yorick Wilks, “Information Extraction: Beyond Document Retrieval”, Computational Linguistics and Chinese Language Processing, Department of Computer Science, University of Sheffield, August 1998.

[25] Buckland M. and Gey F., “The Relationship between Recall and Precision”, *Journal of the American Society for Information Science*, 1994.

[26] Thomas Bloor, ”The Ethiopic Writing System: a Profile”, *Journal of the Simplified Spelling Society*, 19(2), pp. 30–36, 1995.

[27] “Amharic Language”, http://www.britannica.com/EBchecked/topic/20500/Amharic-language, last accessed April 28, 2014.

[28] “Amharic Language”, http://www.lonweb.org/link-amharic.htm, last accessed April 28, 2014

[29] Atelach Alemu Argaw and Lars Asker, “An Amharic Stemmer: Reducing Words to their Citation Forms”, *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, pp. 104-110, Prague, Czech Republic, June 28, 2007

[30] “Amharic Unicode”, http://www.unicode.org/standard/WhatIsUnicode.html, last accessed May 7, 2014.

[31] ባየ ይማም ፣ “የአማርኛ ሰዋሰው”፣ ሁለተኛ ዕትም፣ አዲስ አበባ፣ ጥቅምት 2000፣ ISBN 978-99944-999-8-4.

[32] Kazuhiro Seki and Javed Mostafa, “An Approach to Protein Name Extraction using Heuristics and a Dictionary”, *Proceedings of the American Society for Information Science and Technology*, Volume 40, Issue 1, pp. 71–77, October 2003.

[33] Marios Skounakis, Mark Craven and Soumya Ray, “Hierarchical Hidden Markov Models for Information Extraction”, *Proceeding of International Joint Conference on artificial Intellegence*, pp.427-433, 2003.

[34] C. Sirigayon, H. Chanlekha and A. Kawtrakul, “Information Extraction for Agricultural Information Access”, The Specialty Research Unit of Natural Language Processing and Intelligent Information System, Technology Department of Computer Engineering, Kasetsart University, Bangkok, Thailand.

[35] Ying Yu, Xiao-Long Wang and Yi Guan, “Information Extraction for Chinese Free Text Based On Pattern Match Combine with Heuristic Information”, School of Computer Science and Technology, Harbin Institute of Technology, Harbin150006, China.

[36] Liliana Ferreira, Ant´onio Teixeira and Jo˜ao Paulo da Silva Cunha, “Information Extraction from Portuguese Hospital Discharge Letters”, Institute of Electronics and Telematics Engineering of Aveiro Department of Electronics, Telecommunications and Informatics University of Aveiro, Portugal.

[37] Francisco Costa and Ant´onio Branco, “Extracting Temporal Information from Portuguese Texts”, University of Lisbon.

[38] Kamel Nebhi, “Ontology-Based Information Extraction for French Newspaper Articles”, University of Geneva, Language Technology Laboratory, Department of linguistics, Switzerland.

[39] Louise Deléger, Cyril Grouin and Pierre Zweigenbaum, “Extracting Medication Information from French Clinical Texts”, LIMSI-CNRS, Orsay, France.

[40] Alberto Téllez Valero, Manuel Montes y Gómez and Luis Villaseñor Pineda, “Using Machine Learning for Extracting Information from Natural Disaster News Reports”, Laboratorio de Tecnologías del Lenguaje, Coordinación de Ciencias Computacionales, Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Luis Enrique, Tonantzintla, Puebla, México.

[41] Habert, B., Adda, G., Adda-Decker, M., de Mareuil, P., Ferrari, S., Ferret, O. Ferret, G. Illouz, P. Paroubek, "Towards Tokenization Evaluation", In *Proceedings of lrec* (Vol. 98, pp. 427–431), 1998.

[42] Jakub Piskorski and Roman Yangarber, "Information Extraction: Past, Present and Future", *Proceeding of the conference of Association for Computational Linguistics*, pp. 23-49, Sofia, Bulgaria, 2013.

[43] Jerry Robert Hobbs, the Generic Information Extraction System, Artificial Intelligence Center, SRI International, Menlo Park, CA 9402 5, In *Proceedings of the 5th Message Understanding Conference* (MUC-5), 1993.

[44] Ralph Grishman, "Information Extraction: Capabilities and Challenges", Notes prepared for the 2012 International Winter School in Language and Speech Technologies, Rovira I Virgili University Tarragona, Spain, January 21, 2012.

[45] Tessema Mindaye, "Design and Implementation of Amharic Search Engine", Master's thesis Addis Ababa University, Addis Ababa, Ethiopia, 2007.

[46] Melese Tamiru, "Automatic Amharic Text Summarization using Latent Semantic Analysis", A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in partial Fulfillment for the Degree of Masters of Science in Computer Science, 2009.

[47] J.A. Borsje, "Rule based semi-automatic ontology learning", Master's thesis, Erasmus University Rotterdam, July 2007.

[48] Hamish Cunningham, "Software Architecture for Language Engineering", PhD thesis, University of Sheffield, 2000.

[49] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Johann Petrak, and Wim Peters, "Developing Language Processing Components with GATE Version 6.1", http://gate.ac.uk/sale/tao/split.html, Last accessed March 2015.

[50] Turmo, J., Ageno, A., and Catala, N., "Adaptive Information Extraction", *ACM Computing Surveys*, 38(2), pp. 1-47, 2006.

[51] Tianhao Wu, “Theory and Applications in Information Extraction from Unstructured Text” , A Thesis Presented to the Graduate and Research Committee of Lehigh University In Candidacy for the Degree of Master of Science In Computer Science and Engineering, Lehigh University, June 2002.

[52] Robert Gaizauskas and Yorick Wilks, "Information Extraction: Beyond Document Retrieval", *In Proceedings of Computational Linguistics and Chinese Language Processing*, vol. 3, no. 2, pp. 17-60, August 1998

[53] GAIL C. MURPHY and DAVID NOTKIN, “Lightweight Lexical Source Model Extraction”, University of Washington

[54] Yohannes Afework, “Automatic Amharic text categorization”, A Thesis Submitted to the School of Graduate Studies of the Addis Ababa University in partial fulfillment for the Degree of Master of Science in Computer Science, 2007

[55] Philipp Johannes Masche, “Multilingual Information Extraction”, University of Helsinki, Department of Computer Science, Helsinki, 2004.

[56] Ali Elsebai, “A Rule based System for Named Entity recognition in Modern Standard Arabic”, School of Computing, Science and Engineering University of Salford, Submitted in Partial Fulfillment of the requirments of The Degree of Doctor of Philosophy, Salford, UK, 2009.

APPENDICES

Appendix A: The Amharic character set

Order							Labialized				
1st	2nd	3rd	4th	5th	6th	7th					
ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ					
ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ			ሏ		
ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ					
መ	ሙ	ሚ	ማ	ሜ	ም	ሞ			ሟ		
ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ					
ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ			ሯ		
ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ			ሷ		
ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ			ሿ		
ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቈ	ቊ	ቋ	ቌ	ቍ
በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ			ቧ		
ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ			ቷ		
ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ			ቿ		
ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ	ኈ	ኊ	ኋ	ኌ	ኍ
ነ	ኑ	ኒ	ና	ኔ	ን	ኖ			ኗ		
ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ			ኟ		
አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ					
ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ					
ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ					
ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኰ	ኲ	ኳ	ኴ	ኵ
ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ					
ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ			ዟ		
ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ			ዧ		
የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ					
ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ			ዷ		
ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ					
ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጐ	ጒ	ጓ	ጔ	ጕ
ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ			ጧ		
ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ			ጯ		
ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ			ጿ		
ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ					
ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ			ፏ		
ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ					

ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ

Appendix B: List of Stop Words

ሁሉ
ሁሉም
ኋላ
ሁኔታ
ሆነ
ሆኑ
ሆኖም
ሁል
ሁሉንም
ላይ
ሌላ
ሌሎች
ልዩ
መሆኑ
ማለት
ማለቱ
መካከል
የሚገኙ
የሚገኝ
ማድረግ
ማን
ማንም
ሰሞኑን
ሲሆን
ሲል
ሲሉ
ስለ
ቢቢሲ
ቢሆን
ብለዋል
ብቻ

ብዛት
ብዙ
ቦታ
በርካታ
በሰሞኑ
በታች
በኋላ
እባክህ

በኩል
በውስጥ
በጣም
ብቻ
በተለይ
በተመለከተ
በተመሳሳይ
የተለያየ
የተለያዩ
ተባለ
ተገለጸ
ተገልጿል
ተጨማሪ
ተከናውኗል
ችግር
ታች
ትናንት
ነበረች
ነበሩ
ነበረ
ነው
ነይ
ነገር
ነገሮች

ናት
ናቸው
አሁን
አለ
አስታወቀ
አስታውቀዋል
አስታውሰዋል
እስካሁን
አሳስበ
አሳስበዋል
አስፈላጊ
አስገነዘቡ
አስገንዝበዋል
አብራርተዋል

እባክዎ
አንድ
አንጻር
እስኪደርስ
እንኳ
እስከ
እዚሁ
እና
እንደ
እንደገለጹት
እንደተገለጸው
እንደተናገሩት

እንደአስረዱት
እንደገና
ወቅት
እንዲሁም
እንጂ
እዚህ
እዚያ
እያንዳንዱ
እያንዳንዳቸው
እያንዳንዷ
ከ
ከኋላ
ከላይ
ከመካከል
ከሰሞኑ
ከታች
ከውስጥ
ከጋራ
ከፊት
ወዘተ
ወይም
ወደ
ዋና
ወደፊት
ውስጥ
እባክሽ

ውጪ
ያለ
ያሉ
ይገባል
የኋላ
የሰሞኑ

የታች
የውስጥ
የጋራ
ያ
ይታወሳል
ይህ
ደግሞ
ድረስ
ጋራ
ግን
ገልጿል
ገልጸዋል
ግዜ
ጥቂት
ፊት
ደግሞ
ዛሬ
ጋር
ተናግረዋል
የገለጹት
ይገልጻል
ሲሉ
ብለዋል
ስለሆነ
አቶ
ሆኖም
መግለጹን
አመልክተዋል
ይናገራሉ

አበራርተው
አስረድተዋል
እስከ

Appendix C: List of abbreviations and their Expanded form

ት/ቤት	ትምህርት ቤት
ት/ርት	ትምህርት
ት/ክፍል	ትምህርት ክፍል
ሃ/አለቃ	ሀምሳ አለቃ
ሃ/ስላሴ	ሀይለ ስላሴ
ደ/ዘይት	ደብረ ዘይት
ደ/ታቦር	ደብረ ታቦር
መ/ር	መምህር
መ/ቤት	መስሪያ ቤት
መ/አለቃ	መቶ አለቃ
ክ/ከተማ	ክፍለ ከተማ
ክ/ሀገር	ክፍለ ሀገር
ወ/ር	ወታደር
ወ/ሮ	ወይዘሮ
ወ/ሪት	ወይዘሪት
ወ/ስላሴ	ወሌተ ስላሴ
ፍ/ስላሴ	ፍቅረ ስላሴ
ፍ/ቤት	ፍርድ ቤት
ጽ/ቤት	ጽህፈት ቤት
ሲ/ር	ሲስተር
ጠ/ሚኒስትር	ጠቅላይ ሚኒስትር
ዶ/ር	ዶክተር
ገ/ጊዮርጊስ	ገብረ ጊዮርጊስ
ቤ/ክርስትያን	ቤተ ክርስትያን
ም/ስራ	ምክትል ስራ
ም/ቤት	ምክር ቤት
ተ/ሃይማኖት	ተክለ ሃይማኖት
ሚ/ር	ሚኒስትር
ኮ/ል	ኮለኔል
ሜ/ጀነራል	ሜጀር ጀነራል
ብ/ጀነራል	ብርጋዳር ጀነራል
ሌ/ኮለኔል	ሌተናሌ ኮለኔሌ
ሊ/መንበር	ሊቀ መንበር
አ/አ	አድስ አበባ
ር/መምህር	ርእሰ መምህር
ፕ/ት	ፕሬዝዳንት
ዓ.ም	አመተ ምህረት
ዓ.ዓ	አዲስ አበባ
ዶ.ር	ዶክተር
ፕ/ር	ፕሮፌሰር

Appendix D: List of Titles

አቶ
ወ/ሮ
ወ/ሪት
ዶ/ር
ሸህ
ቄስ
ክቡር
ክብርት
ሻምበል
ኮሎኔል
አስር አለቃ
አምሳ አለቃ
ሻለቃ
ጀኔራል
ጀነራል
ፕሮፌሰር
ወታደር
ኢንጅነር
ኰሎኔል
ልኡል
ራስ
አቡነ
መምህር
አለቃ
ብላታ
ሀኪም
ነጋድራስ
ፕ/ት
አፈ ጉባኤ
ማእድንና ኢነርጂ ሚኒስትር
ወይዘሮ
ጠቅላይ ሚኒስትር
ዳይሬክተር
ድያቆን
ባላምበራስ
ብላቴን ጌታ
ፊታውራሪ
ብላታ
አባ
ደጃዝማች
ኰሎኔል
ሜጀር
ጀነራል
በጅሮንድ
መምህር
ግራዝማች
ብላቴን ጌታ
ባላምባራስ
ሊቀ ጠበብት
ዶክተር
ሻንበል
ነጋድራስ
ወ/ሮ
ኢንጂነር
ሰአሊ
ፒያኒስት
ሚ/ር
ጠ/ሚኒስትር
ጠ/ ሚኒስቴር
ፕሬዝዳንት
ፕረዝዳንት
ፕሬዚዳንት
ፕሬዚደንት
ፕረዝደንት
ካፒቴን
ፓትሪያርክ
ኢንስፔክተር
ሀጂ
አርቲስት
አፈ-ጉባኤ
የተከበሩ
አምባሳደር
ኮማንደር
ብርጋድየር ጀኔራል
ሌተናል ኮሌኔል
ሹም
ፕ/ር
አፄ
መቶ አለቃ
ሚስተር
ጠ/ሚ
ሚኒስትር ድኤታ
ብፁእ
ዶክተር
ተመራማሪ
ከንቲባ
ሊቀመንበር
ምክትል
ሳጅን
አ/አለቃ
ከንቲባ
ክቡር
ሎሬት
ሀምሳ አለቃ
አሰልጣኝ
አምበል
ኡስታዝ
ኢንስትራክተር
ሸክ
ዋና ዳይሬክተር

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.

Bekele Worku

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

Yaregal Assabie (PhD)

Addis Ababa, Ethiopia June, 2015