ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
SCHOOL OF INFORMATION SCIENCE

Amharic Question Answering for Factoid and List Questions using Machine-learning Approaches

By

Medhanit Getachew

February 2019

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
SCHOOL OF INFORMATION SCIENCE

Amharic Question Answering for Factoid and List Questions using Machine-learning Approaches

Medhanit Getachew

A Thesis Submitted to the School of Information Science of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Science and Systems (Language Technology)

February 2019

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
SCHOOL OF INFORMATION SCIENCE

Amharic Question Answering for Factoid and List Questions using Machine-learning Approaches

Medhanit Getachew

Name and signature of members of the examining Board

Name	Title	Signature	Date
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

Declaration

This thesis has not previously been accepted for any degree and is not being concurrently submitted in candidature for any degree in any university.

I declare that the thesis is a result of my own investigation, except where otherwise stated. I have undertaken the study independently with the guidance and support of my research advisor. Other sources are acknowledged by citations giving explicit references. A list of references is appended.

Signature: _____

Medhanit Getachew

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

Advisor's Signature: _____

Ato Ermias Abebe

Acknowledgment

Primarily, I would like to thank God, who paves the way for me and makes everything possible with love. My gratitude is then to my advisor Ato Ermiyas Abebe for his beneficial and constructive advice and for his patience through this research.

Special thanks also goes to my family mainly for my dad Getachew Kebede and my brother Mekdim Getachew, my friends who have been behind me in supporting and encouraging me through difficult times.

In addition, thanks to my LT track classmates and teachers for being so supportive.

Table ofContents

List of Figures

List of Tables

List of Acronyms

HMM	Hidden Markov Model
IR	Information Retrieval
POS	Part Of Speech
QA	Question Answering
ML	Machine Learning
NER	Named Entity Recognizer
NLP	Natural Language Processing
SVM	Support Vector Machine
TREC	Text Retrieval Conference

Abstract

Question answering is a system that allows users to ask questions about some topic in natural language and give exact answers by retrieving answers from collection of documents. Its main aim is to assist human to get exact answers to questions they ask. In addition, it avoids going through many documents to find a single answer to their questions. There are two types of questions in QA namely factoid and non-factoid questions. The first one comprises of what, where, when, who questions and the second one deals with list, definition, acronym, how questions. The focuses of this study are factoid and list questions.

There are some researches conducted previously on question answering. Most of the researches used only SVM algorithm for question classification and any of them did not make use of named entity recognizer for answer extraction. In this study an attempt is made to design a list and factoid question answering using machine learning approach and an answer extraction that makes use of NER.

This research is a closed domain QA for Amharic that focuses on Ethiopian history. It has three components. Question classification for identifying the types of questions which is done using two algorithms; HMM and SVM, passage retrieval that is performed by selecting the relevant sentences using sentence-level retrieval and answer extraction component selects answers from the top ranked sentences using a NER which is developed for this research. Factoid questions are answered by using key words matching and extraction using the NER from the question and the list questions are answered by using co-occurrence of answer types and candidate answers in a text.

The study achieved an F-measure of 73% using the SVM classifier for question classification and an F-measure of 65% was achieved using the HMM classifier for question classification. From the result we achieved, we realized that question classification using SVM has a better answer extraction performance than the HMM. In addition, the use of NER tool helped answer extraction in getting exact answers.

Key words: Amharic Question Answering, Question Classification, Answer Extraction, Factoid Questions, List Questions

Chapter 1

1 Introduction

1.1 Background

Natural language processing is a field related to the area of computer science, artificial intelligence, linguistics and human computer interactions by means of which computational mechanisms are investigated and formulated[1]. These mechanisms allow the development of systems that is capable of understanding the knowledge expressed in texts of a given language. Natural Language Processing is a computational technique for representing and analyzing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications [1].

The most common applications in NLP include machine translation, information retrieval, information extraction, question answering, recognition of entities, classification of documents, generation of summaries, etc.

Question answering is an automated technique to retrieve correct and short answers to the questions asked by human in natural language. The main aim of QA system is to assist in the interaction of human with machines[2]. There are two types of question answering systems. Closed domain question answering deals with questions in a specific domain and open domain question answering deals with questions almost about everything. The types of questions used in a question answering are factoid questions (who, when, what, where) and non-factoid question such as definition, list and acronyms.

The state of the art in QA technology combines machine learning with linguistic information encoded by human experts in the form of rules in most of the QA systems' components. By developing fast and efficient algorithms and models for real-time processing of data, machine learning is able to produce accurate results and analysis[3].

A typical architecture of a QA system consists of three components linked sequentially. The first one question analysis identifies the type of the input question. Retrieval, which is the second component, extracts number of relevant documents that are likely to contain the answers and this component might incorporate paragraph/sentence retrieval depending on the needs and techniques used in the QA systems. Finally, answer extraction extracts and ranks exact answers from the previously retrieved passages [4].

This study attempts to design a factoid and list question answering for Amharic language using machine learning approachs in the question classification components and other NLP tools are used for answer extraction.

1.2 Statement of the Problem

Humans ask many questions throughout their lives in order to know the world they are living in. When someone asks a question, that person may want a specific answer to that particular question. Seeking an answer from the web, a person might crawl through many links and pages suggested by the web but still might not get any answer for that particular question. While crawling the web, that person gets a bunch of documents that could hold the answers for the questions raised but require reading one or more of the documents to get the answer. A question answering system reduces the processes of going through number of documents to get a single answer.

In the meantime, if a person inputs a question in Amharic to search through the web, that person might not get any response or might get some documents that hold similar words in the question or documents that might hold an answer to the question. However, the person will not get an exact answer. An Amharic QA, just like any other QA systems, should return exact answers to users.

Linguists put the formation of Amharic questions and answers differently. Accordingly, one can raise questions about some action or condition, about the performer of an action, about the agent who performed the action or time and place, about the cause of the action or aim of the action, how the action is performed or techniques used to perform the action, and so on[6]. The basic

challenge in Amharic is the interrogative words which are multipurpose, in which a word used to ask about a place might also ask about a person. For example, in the question, “የ ውቅዲ ውልዶ የ ተፈረመው በማን እና በማን መካከል ነበር?”.

There is a research work done on Amharic question answering for factoid questions[6]. Hand crafted rules and regular expressions were used for identification of question types and also a gazetteer was used to get named entities. Other works such as a web based Amharic question answering for factoid questions[7], Amharic question answering for list questions[8] and Amharic definitive question answering[9] applied machine learning approach and used SVM for question classification task.

The research works mentioned above did not make use of POS tagging and NER tools, which make the performance of question answering better and help in making the answer extraction and retrieval easier. Instead, some of them used gazetteer based entity recognition. Even though there had been researches undertaken to develop such tools (POS and NER), there are no systems available to integrate to question answering systems since those studies were made as MSc researches. For this study, we develop and use these tools to make the retrieval process easier and to get a better performance.

The classification algorithm the previous works have used is only SVM but we investigate both SVM and HMM and compare the performance of the two algorithms. Furthermore, since there was no previous work done by combining both list and factoid questions we investigated a multi-type question answering that comprises factoid and list questions. Moreover, in this study we introduced a new design that is a multi-type question answering with a better answer extraction using an NER tool and different classification algorithms.

1.3 Research Questions

- How can we integrate factoid and list questions and come up with a multi-type question answering for Amharic?
- To what extent will the development and use of NER and POS tagger help gain better performance?

1.4 Objective

1.4.1 General Objective

The general objective of this study is to design Amharic Question Answering system for factoid and list questions.

1.4.2 Specific Objectives

To achieve the main objectives, the study has the following specific objectives:

a) To study the general features of Amharic factoid and list questions.

b) To construct a general architecture of Amharic Question Answering system.

c) To develop a prototype for the new system.

d) To evaluate the designed system and check performance differences.

1.5 Significance of the Study

This research has a great significance to the development of question answering system. It can be an input for other researchers working on Amharic factoid and list questions and for others who work on a full-fledged QA that integrates factoid and other non-factoid questions.

1.6 Scope and Limitation of the Study

The scope of the study is limited to Amharic language as the title indicates because the researcher is familiar to the language. The question types considered are only factoid questions and list questions to come up with an integrated question answering. Since the study focuses on a specific domain, it will only be about Ethiopian history covering topics from the ancient to modern ages. We selected this specific domain because the researcher believes that in this domain, many questions can be raised and used in QA. Taking the tools used for the study in to consideration, only NER and POS tagger is developed and used from the NLP tools we need. In addition, algorithms used in this study for question classification are only SVM and HMM.

1.7 Methodology

The research methodology applied for this particular research is a design science approach. Design science approach involves a strict process to design artifacts to solve observed problems, to make research contributions, to evaluate the designs, and to communicate the results to the appropriate audiences[10].

For better understanding of the work and the approaches that are available, techniques and algorithms used and also to find the gaps from previous works, literature review is used. The

dataset used for this research is an Amharic corpus of Ethiopian history, collected from different books, newspapers, electronically available documents and the Internet. In addition, the possible questions that people could ask about Ethiopian history is collected using different techniques. The corpus gathered from different sources first pass through some pre-processing tasks and then the appropriate machine learning approach is applied to go to the next steps, which are question analysis, document retrieval and answer extraction. For the development of the question answering, python programming language is used. Some NLP tools such as part of speech tagger and named entity recognizer are developed and used, as both tools are very important for the designed system. The experiment is evaluated and tested at each phases of the research and performances are compared with previous works. The methodology we used is explained well in chapter four of this research paper.

1.8 Organization of the Thesis

This thesis is organized into six chapters. The first chapter introduces the background of the study and the statement of the problem that leads to this study. In addition, the objectives of the study, significance, the scope and limitation and methodologies used in the study are presented.

The second chapter covers the literature review on question answering. This chapter discusses the overview of QA, types of QA, the approaches and components of QA. In addition, it covers related works from local and global level. Similarly, the third chapter presents about the Amharic language. Formation of questions in Amharic, construction of sentences and the punctuation marks used in Amharic are included in the chapter.

In the fourth chapter, we present the methodology and design that we used in this study. The steps followed in design science are discussed in accordance with our study including the system architecture and components used in designing QA using factual and list questions.

The fifth chapter, which is the experiment and evaluation, describes about the experiments we performed and results found using the evaluation techniques. Finally, the last chapter is about the conclusion we made after the findings of the experiments and recommendations that we suggested to other researchers.

Chapter 2

2 Literature Review

2.1 What is Questions Answering?

Question answering is one of the tasks performed in natural language processing. It is a system that allows users to ask questions about some topic in natural language and gives exact answers by retrieving answers from collection of documents[5]. It has two interrelated goals: first understand issues in natural language understanding and representation and second to develop a natural language interface to computers. Natural language understanding and representation enables users to address computers as they are addressing human using their native language or natural language but this will not be possible if computers cannot understand the natural language of human. So in order to make these goals happen, natural language processing designing methods that can analyze, understand and generate languages that humans use naturally must be developed in addition an interface to access all of these should be developed.

A question answering system is mainly used to provide short and precise answers to users' questions using natural language. Therefore, users are able to ask questions and retrieve answers using their native language. The system can be used to search over a fixed set of books, to search over the web or for reading comprehension. Companies can use domain specific Question Answering techniques internally for the employees who are searching out answers to provide information for clients. Another application is in education and medical fields, it can also be used for answering questions in fields where there are frequently asked questions that people want to search [11].

Information retrieval is always mentioned when a question answering system is developed. Information retrieval is related to question answering in which users use IR in order to find answers to their questions but what they get on hand is number of documents to look for answers for their questions. IR is used in question answering for two purposes. One on the stage of passage retrieval, to get the relevant passages from the retrieved documents and the other is the IR community has developed a methodology for evaluation, which are the annual Text REtrieval

Conferences (TRECs), run by the US National Institute of Standards and Technology. It is from this methodology and community that the recent question answering evaluation is developed [5].

2.2 Types of Question Answering Systems

Question answering system deals with a wide range of question types including: factoid, list, definition, How, Why, hypothetical and cross-lingual questions. There are however two basic types of question answering systems [12].

2.2.1 Closed-domain Question Answering System

Closed-domain question answering deals with questions under a specific domain. It is a system that accepts only limited types of questions. It has very high precision as it returns exact answers but requires wide language processing and it is restricted to single domain [12]. In close domain system, answers to question have to be stored since they are not available in public domain. we cannot search them using any search engine.

The first step in closed domain system is to understand natural language questions so that solution could be matched to the respective answer in the database or corpus[1]. Understanding natural language will also help to answer questions accurately. Restricted or closed domain can be taken as an efficient approach for improving accuracy of question answering systems since the search will be on a specific area of focus.

2.2.2 Open domain Question Answering System

Open domain systems depend mostly on word knowledge and general ontologies and deals with questions about nearly anything [1]. These systems usually have much more available data from which answers can be extracted. The most important challenge of an open domain system is its database. The efficiency of this system depends on how well the database is arranged and maintained since it tries to answer almost everything.

While considering open domain systems, web based systems are also mentioned most of the time because of the capability of the web to cover almost everything. In addition, majority of web based systems are also focused in open domain. When other search techniques failed to provide short and to the point answers to users' queries, a web-based question answering systems were developed [13]. Web based systems use search engines to get web pages that are expected to

contain the answers to users' questions and then extract the most similar answer from all possible answers returned as a search result. Each search result usually contains title, URL and some strings of the related web document and that is called "snippets" [1]. Since downloading web documents and analyzing them consumes more time, the system makes use of the snippets in the search result.

START can be taken as an example of an open domain and web based QA system. The system works by talking snippets containing possible answers. The possible answers are classified using a vector space model then to find more accurate answer, ranking is also performed. Finally the evaluation of the final answer will be done and returned to the system [1].

2.3 Architecture of a Question Answering System

A typical architecture of QA is known to contain three components; question analysis, document retrieval, and answer extraction as it is depicted in the figure below [14]. In this architecture, the question analyzer which is a sub-component of question processing is the component expected to analyze the question that determines the proper answer type and formulate proper queries for document retrieval. The document retrieval component will retrieve the top related documents according to the formulated question and then will be subjected to passage retrieval later. The final component, answer extraction component is responsible to extract the correct answer from the ranked extracted passages.

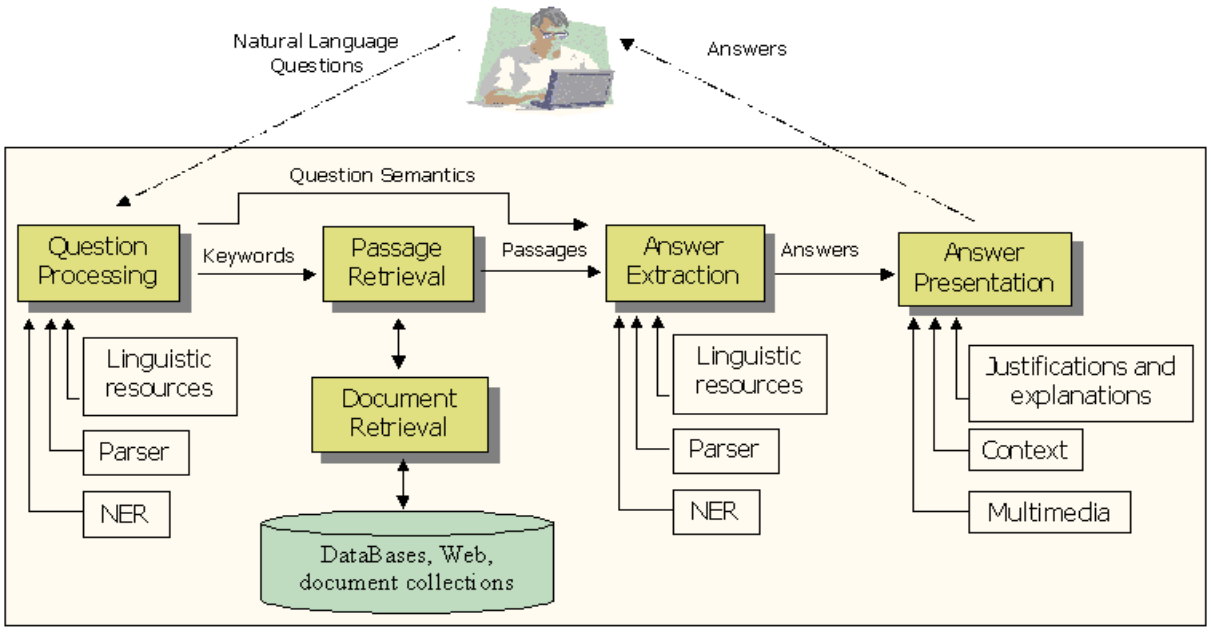


Figure 2-1: Architecture of a Question answering System

2.3.1 Question Analysis

Question analysis is the most important component of question answering. This component processes the question, analyzes the question type, and produces a set of keywords for retrieval. Question processing includes the task of tokenization, POS tagging, named entity recognition and natural language parsing and other pre-processing tasks. The question type further indicates what will be the expected answer types. Correctly identifying the expected answer type will help the later stage of answer extraction to identify answers correctly. In other way, if there is a mistake in question identification it will result wrong answer. Identifying Keywords and definition terms help the system to locate sentences where answers can probably be found [15].

Users could ask the same question in different ways. Depending on the retrieval and answer extraction approaches, some question analysis components also try to recognize semantics of the questions. The semantic context helps the system to decide between different possible answers by comparing expected answer and probable answer. For this purpose, a WordNet based tool can be used to process questions type. The semantic context also helps in approximating the type of the expected answer when the analysis is unable to obtain it. It also allows sub-classifying concepts which are broad for each particular question and it helps the system to decide between different possible answers by comparing expected answer and probable answer semantic contexts [15].

2.3.1.1 Question Classification

As indicated above the question analysis phase plays a great role in determining the answer type. In order to fulfill this purpose, we use question classification to determine the type of answer the user is expecting to receive from a given question. Question type is a semantic category of questions, which are characterized by common properties such as Questions of type Place, Person, Term Definition, Quantity, Listing, Explanation, True/False, and Time[8].

Question classification is concerned about assigning semantic classes to questions. This semantic classification reduces the search space of possible answers. If we can determine the specific

semantic class of a question, we can then easily look for the instances of that particular answer type. It can be done using hand written rules or using machine learning based approach.

2.3.2 Document Retrieval

Question answering systems use document retrieval in order to get documents that contain the expected answer for user's questions [5]. This indicates that question-answering systems depend mostly on retrieval systems. Question answering mostly contains a retrieval subsystem that will help in identifying documents or passages which may contain an answer for the question. The document retrieval component provides ranked documents that can be used by answer extraction component, which is the next component of the question answering system. Some question answering systems use the retrieval system to retrieve related documents, which are further used for another subsystem called passage retrieval.

2.3.2.1 Passage Retrieval

Although set of general documents are ranked by relevance, it does not mean the top ranked document is the answer for a user's question. This is because documents are not an appropriate unit to rank with respect to the goal of a question answering system [3]. There are a number of weighting schemes which can be used within vector space model for retrieval. The most common term-weighting strategy is known as the tf-idf strategy, which stands for term frequency and inverse document frequency. Term frequency refers to the number of times a term appears within a document. The inverse document frequency of a term is a measure of how rare the term is across the entire corpus. The insight is that if a term occurs frequently in a document, but not frequently in the corpus considered as a whole, then that term does a good job of describing that document. In tf-idf weighting, each term is weighted by the product of its term frequency and its inverse document frequency.

The first stage in the passage retrieval is to leave out the passages that do not contain expected answers from the retrieved documents and then rank the rest of the passages according to the probability they contain to answer the question. So, the first step will be to run a named entity or answer type classification on the retrieved passages. The answer type determined from the question will indicate the possible answer type we should expect.

The remaining passages are then ranked, relying on a small set of features such as: the number of named entities of the right type in the passage, the number of question key words, the rank of the document from which the passage will be extracted[16].

2.3.3 Answer Extraction

Answer Extraction is the last component in question answering, which is responsible for identifying, extracting and validating answers from the set of ordered paragraphs passed to it.

Answer extraction is implemented by inspecting the candidate passages, which is full of information including POS and named entity tags. If the named entity in the candidate passages corresponds to an expected answer type, the entity will be picked out as a candidate answer. When multiple candidate answers exist, the selection of the best candidate answer is performed with an answer ranking scheme that relies on heuristics method [1].

Answer extraction component for non-factoid questions also extracts answers from a paragraph of retrieved documents according to answer extraction patterns of each question type [17]. This component searches for linguistic behavior of each question type, which are mentioned above for each passage.

2.3.3.1 Named Entity Recognition for Answer Extraction

Named Entity Recognition is a specialized form of an information extraction task dedicated to identifying phrases in text that refer to entities like people, organizations, date and currency amounts, and extracting their semantics. It is also the most important task in question answering so that answer types can be detected easily. Names appear frequently in many types of texts, and identifying and classifying them simplifies further processing. Names are identified by a set of patterns which are stated in terms of parts-of-speech, syntactic features, and orthographic features (e.g. capitalization for English).

Combining the NER with semantic context will improve system performance by increasing the amount and quality of the information obtained from the question; it also improves possible answers detection and extraction.

2.3.4 Answer Presentation

In TREC systems, the most relevant set of answers is presented in different forms such as presentation of exact answers with a principle of one answer per one question, supported answer and an answer with the document where the exact answer string presented with its corresponding document. This component uses natural language generation for presentation of answers and artificial intelligence for answer justification.

2.4 Approaches to Question Answering

Approaches of QA systems are divided into groups based on the methods used. Some group of QA system belongs to simple natural language processing and information retrieval methods, while another group of QA systems depend on reasoning with natural language [11]. The previous one uses techniques of NLP such as syntax processing, NER and information retrieval techniques in order to achieve the goals of a question answering system; uses free text documents as a data source; the questions are mostly WH- type and use the existing IR evaluation techniques. However, the later uses higher reasoning techniques; knowledge base is used as a source of data, mostly domain oriented; might be beyond WH- type of questions and no evaluation techniques are introduced.

2.4.1 Rule Based Approach

In this approach, a wide coverage of NLP techniques is used in order to achieve accuracy of the answers retrieved. These systems first generate training data and test data through a semantic model. Some systems of this type generate rules for each type of questions such as the semantic classes who, when, where and why type questions. “Who” rules look for names that are the class of nouns. “When” rules consist of time expressions only. “Where” rules mostly look for locations but might be different in the case of Amharic. These systems require learning rules from training data and are mostly used in reading and answering comprehension questions. A system called Quarc used some heuristic rules that look for sematic and lexical hints to identify the question class [18].

2.4.2 Pattern Matching Approach

This approach makes use of text patterns in place of complex processing involved in other competing approaches. Many of the questions answering systems automatically learn the text patterns from passages rather than using linguistic knowledge or tools such as named entity,

WordNet, ontologies, etc. for retrieving answers. For example, the question “who is the founder of Addis Ababa?” will look for the pattern “who is <person name > of <location>?” and the answer be like “<person name> is the founder of Addis Ababa”.

Most of the patterns matching QA systems use the surface text patterns while some of them use templates for response generation.

2.4.2.1 Surface Text Pattern

Initially, the surface text pattern based method was aimed at finding answers to factual questions, as their answer is limited to one or two sentences. This approach extracts answers from the surface structure of the retrieved documents by relying on some list of patterns. Answer to a question is identified depending on the similarity between their patterns having certain semantics. These patterns are like regular expressions.

Some systems used this pattern matching as an alternative approach for difficult questions like acronym expansion questions, date of birth questions and location questions. A system was developed by some researchers by integrating surface patterns with a named entity tagger. Another system used a soft pattern matching based on bigram model and HMM instead of regular expression. This approach can be applicable for a small and medium sized websites [2].

2.4.2.2 Template Based Pattern

A template-based approach makes use of preformatted patterns for questions. The focus of this approach is more on illustration rather than interpretation of questions and answers. The set for templates is built in order to contain the optimum number of templates ensuring that it adequately cover the space of problem, and each of its members represents a wide range of questions of their own type. Templates have entity slots, which are missing elements bound to the concept of the question that has to be filled to generate the query template to retrieve the corresponding response from the database. The response returned by query would be raw data, which is returned to the user [2]. In template-matching precision of document retrieval is high because the key words for finding answers are selected using human intelligence[19].

2.4.3 Machine Learning Approach

Machine learning is one of the approaches of question classification in a question answering system. It can automatically construct question classification program with a high performance, which can be used for more features of questions[4]. Given more training data, the performance of a learned classification program usually improves. For some tasks such as identifying the question type, machine-learning classification can be taken as classical classification tasks that can be solved by different ML algorithms. The advantage of using machine learning approach is that because it is flexible, more adaptable and easily maintainable [20]. There is no need of handcrafting rules to maintain and adapt to new changes every time, which is why this research is using this approach. IBM Watson can be an example of machine learning based system; Maximum Entropy model was used for question and answer classification based on N-gram features [2].

In machine learning approach, a question classifier is trained in a supervised manner. Possible choices of classifiers include but are not limited to Nearest Neighbors (NN), Naive Bayes, Decision Tree (DT), Sparse Network of Winnows (SNoW), Hidden Markov Models (HMM) and Support Vector Machines (SVM). We will look through each in the paragraphs below:

The Nearest Neighbors algorithm is a simplified version of the well-known KNN algorithm, which has been successfully applied in document classification[21]. Given an unlabeled instance, the NN algorithm finds its nearest or more similar neighbors among the training examples, and uses the dominant class label of these nearest neighbors as its class label.

The Naive Bayes algorithm is a commonly studied algorithm in machine learning. It is considered as one of the top performing methods for document classification. Its basic idea is to estimate the parameters of a multinomial generative model for instances, then find the most probable class for a given instance using the Bayes' rule [22].

The Decision Tree algorithm is a method for approximating discrete valued target function, in which a tree of arbitrary degree that classifies instances represents the learned function. The C4.5 software is a widely used implementation of the DT algorithm [22].

The Sparse Network of Winnows (SNoW) algorithm is specifically tailored for learning in the presence of a very large number of features and can be used as a general purpose multiclass classifier. The learned classifier is a sparse network of linear functions [22].

Hidden Markov Model is very powerful from mathematical structure point of view and it has formed many theoretical bases of different applications, also if the model is implemented well it could be used for many applications, which is why we selected it as a candidate algorithm for this research[21]. In HMM the state sequence cannot be identified meaning the status sequence could not be observed and is hidden, so it is called HMM. It is one of the most important tools of analyzing the random sequences of finite status structure and a good method for QA system. It searches all documents according to the possibility of relation to the question and finally the lists of the documents are presented to the user. It presents the highest similarity to the asked question and one can say that it is guaranteed in HMM that the answer presented, is the accurate one[23].

HMM has 3 parameters: Start Probability, Transition Probability and Emission Probability[39]. Start Probability means the probability that a tag exists first in a sentence. Transition Probability means the ratio of probability of transition from given tag to the next tag and the Probability of occurrence of a given tag. Emission Probability is the ratio of probability of occurrence of particular word with tag t and the Probability of occurrence of a tag t.

Support Vector Machine is a binary linear classifier, which takes in a set of training data and classifies each member of the set into one of the two categories. The training algorithm builds a model that can be used to classify the items from a test data set into one category or another. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

Support Vector Machines are linear functions of the form $f(x) = \mathbf{w} \bullet \mathbf{x} + b$, where $\mathbf{w} \bullet \mathbf{x}$ is the inner product between the weight vector **w** and the input vector **x**. The SVM can be used as a classifier by setting the class to 1 if $f(x) > 0$ and to -1 otherwise. The main idea of SVM is to select a hyperplane that separates the positive and negative examples while maximizing the minimum margin, where the margin for example x_i is $y_i f(x_i)$ and $y_i \in \{-1,1\}$ is the target output. This corresponds to minimizing $\mathbf{w} \bullet \mathbf{w}$ subject to $y_i ((\mathbf{w} \bullet \mathbf{x}) + b \geq 1)$ for all i. Large margin classifiers are known to have good generalization properties. To deal with cases where there may be no separating hyperplane, the soft margin SVM has been proposed. The soft margin SVM minimizes $\mathbf{w} \bullet \mathbf{w}$.

2.5 Answering Factoid Questions

Answers for factoid questions are mostly short answers that require factual answers such as location, name and time. It was first proven by IBM Watson that supervised machine learning can be applied for factoid question answering. Factoid questions usually carry some information about the type of the answer. A factoid question has exactly one correct answer that can be extracted from short text segments[24]. Answering factoid questions is much simpler than the other categories. One of the important issues in the factoid question answering is the answer ranking. The correct answer for a question should be in the top of the produced answer list by a QA system.

2.6 Answering List Questions

A list question is a question, which requires a list as an answer to be collected from number of documents. List questions are organized into questions in which each has its own target. The instances of the answers to the list questions happen to occur with the target and question keywords. Some systems treat list questions as an extended version of factoid questions and answer a list question by simply returning the top N answers found by the factoid question answering system. List questions can also be answered by using relationships between question terms and answers [25].

A list question is answered in the following steps: First, the answer type of the question is determined. For this purpose, each question is associated to one of the known entities such as

person, country, organization, city and others. This is done by using lexical and semantic patterns. After the type of answer is predicted, a number of documents are retrieved using a query generated from the target and the question. Depending on the answer type, the candidate answers are extracted using a named entity tagger [25]. A similarity value is then, computed for each pair of candidate answers based on their co-occurrence within sentences. Candidates are grouped into the same entity group and determined their similarity; then the final candidate answers are selected.

2.7 Evaluation of a Question Answering System

An answer is judged correct if it is answered as expected. If collections of documents do not support a correct answer, it is unsupported. If bits are added or missing, it is judged as inexact answer. If a question does not have a correct answer in the document collection "no answer" is returned [26]. For automatic evaluation, pairs of correct answer patterns and supporting document identifier is provided for each question from sets of all correct answers.

There are four standard evaluation metrics used for evaluation of a question answering which are: Precision, Recall, F-measure and Mean Reciprocal Rank (MRR). Precision is the measure of the accuracy of an answer where proportion of the number of correct answers to the number of returned answers is calculated, and Recall is the proportion of the number of correct answers to the number of test questions. Both precision and recall are set based measures as they evaluate the quality of an unordered set of retrieved documents. F-measure is the harmonic mean of Precision and Recall. Mean Reciprocal Rank (MRR) considers the rank of the first correct answer in the list of possible answers [24]. It is mainly concerned about correctness which is denoted by 0 or 1.

2.8 A survey of Amharic Question Answering Research

A question answering system has been designed for different languages in the world such as English, Arabic, Chinese, and Hindi. Even though they are designed for fulfillment of an MSc research there are some works in local languages such as Amharic and Affan Oromo as well.

In this section, we present some local related works.

2.8.1 Amharic Question Answering (AQA) For Factoid Questions

The work of [6] attempted the first Amharic factoid question answering system. The research was conducted by taking the nature of Amharic questions in to consideration since construction of Amharic questions is different from English. Corpuses were collected from 15600 news articles from different newspapers. The documents collected passed through some pre-processing phase to put them in the appropriate format. The main pre-processing techniques used in the research are sentence/paragraph demarcation (fixing boundary), Ethiopic number normalization, character normalization, sentence based tokenization, stemming, stop-word removal and gazetteer preparations.

After preprocessing, the documents were indexed using Lucence indexer, the next step is to process the user question, which will help in generating a well structured query. User generated questions are received by the system and a rule based question type identification carried out to determine what type of question the user is asking and what type of answer is being expected from that question. For question classification, algorithms are developed and implemented using the Java programming language in the eclipse editor. After question classification, the document retrieval component uses the Lucene API with some modifications. In addition to the core Lucene API components, some packages such as RegexQuery for regular expression based searching were also used.

Rule-based answer selection techniques are considered for answering questions that cannot be answered directly by the named entity recognition technique. In the rule-based technique, rules have been developed to find some specific types of answer particles for some question types. The rules developed help in extracting foreign person names and place names where the person name and place name cannot be found in the gazetteer list.

Evaluation of the system was mainly for correctness of answers. The researcher evaluated the system using precision, recall and MRR. The rule based question classification module classifies about 89% of the question correctly. For answer selection, the researcher used different selection approaches and the gazetteer based answer selection using a paragraph answer selection technique answers 72% of the questions correctly.

The researcher recommended that the development of NER, POS tagger, stemmer, parser and Amharic spell checker is good for performance improvement. Integrating with search engines, incorporating with machine learning and statistical question classification and extending to other domains to apply to different organizations was also suggested.

2.8.2 Amharic Question Answering for List Questions

The work of [8] was aimed at solving the problem of users struggling to find answers in IR systems because what is returned after the search is a bunch of documents that are expected to contain the answers for their questions.

Question answering system for list questions is designed based on the distributional hypothesis, which states that "words occurring in the same contexts tend to have the same meanings" [8]. In list questions, the answer to the question includes one or more entities of a given type. The type is given as part of the question and used by the system to enhance the ability of addressing the question. Since the type of the concepts of interests that answer a given question are often obvious from the question itself, this form of information need representation using an answer type.

In document retrieval module of the system, documents that are containing the answers will be retrieved from the large corpus. For this task, a probabilistic IR system for Amharic language has been applied with some modifications on customizing steaming and query generation modules. Then the answer extraction module takes place to select the expected answer type.

After the answer extraction that is carried out next to document retrieval, unwanted answers are found in the list even if all the correct answers also exist in it. The reason for this problem is the co-occurrence module. Answer instances of a list question co-occur within the sentences of the

documents related to the target and the question [8].This module is developed by using sentences because answers co-occur within the sentences of the documents related to the answer type and the question.

Once all the data regarding co-occurrences is collected, the similarity between each pair of terms is computed to select candidate answers from the lists of the extracted answers. Candidate terms that co-occur more often with other candidate answers are more likely to be the answer. So the sum of the similarities of each term to other term is used as an indicator [8].

For evaluation purpose, the researcher used overall precision and overall recall because all instances have equal weight while in precision and recall, each question is given equal weight and the weight is further divided into the answer instances of the question. Therefore, instances in questions with fewer answers have more impact than those in questions with more answers. The overall precision of the overall system is 0.61 and the overall recall of the system is 0.636[8].

Verifying the list of answers using an already collected question answer match is recommended by the researcher to enhance the performance of the system. Development an IR system that can work for a small collection of documents as it provides the documents that contain the candidate answers and decrease number of documents to be processed for answer and the development of standard POS and NER is also suggested.

2.8.3 A Web Based Question Answering for Factoid Questions

Due to the richness of the web the researcher at [7] assumes it is feasible to construct a web based question answering system that returns short answers for users' questions. As the corpus used for the research is gathered from Amharic web sites, a J-spider crawler is used to download pages from the sites. Then, the language identification module will filter out non- Amharic content pages in the search space. The Amharic documents are then indexed in a way to facilitate information retrieval later in the system using the Lucene indexer. When the search space is ready, then the question answering part will start playing its role serving the users [7].

The system has two major parts, the search engine part (crawler and indexer) and the question answering part. The web crawler used in the search engine retrieve web pages and downloads them or their representations to a local repository used for searching. Language Identification Module (LIM) used for language identification takes the crawled web pages from the repository and puts into another repository that is to be used by the Lucene indexer. Then the other part of the search engine, the indexer, extracts a text from the downloaded documents. Lucene index contains a sequence of documents, where a document is a sequence of fields and a field is a named sequence of terms called strings.

In the question answering part, question classification is performed using a classifier algorithm SVM. For training the classifier, 1200 questions from the four classes of factoid questions (person, time, place and quantity) are used. For answer extraction, the researcher used pattern-matching approach with proximity between the query terms and candidate answer terms. These candidate answer terms are identified from the documents based on some pattern matching technique.
It should be noted that the passage retrieval component is not much of important in this research as it is an open domain QA and in such systems answers might be found from different documents not only from a single passage.

The attained accuracy of the system with application of a machine learning based classification algorithm, support vector machine (SVM), is 94.2% in question classification. In addition, the overall performance of the system in answering users' questions was of 77%.

The researcher recommended the improvement of Amharic stemmer, POS tagger and spell checker would result to an improved performance. Since the researcher only used the SVM classifier, trying out the other algorithms is also recommended.

2.8.4 Amharic Definitive Question Answering

The increase of electronic data made the IR system not to be helping in the retrieval of exact information needed in a short period. The QA systems that are constructed for foreign languages cannot be applied directly for Amharic language because the QA components are language

dependent so, it is vital to develop one for Amharic. These and other problems initiated the investigation of the researcher at [9] in Amharic question answering for definition questions.

The research focuses on a closed domain QA that answers questions about law. Amharic law documents and training materials related to law are collected because law documents include definitions and there is no standard Amharic corpus to experiment on open-domain definitions. Corpus is collected from Ministry of Justice, Federal Supreme Court and websites. The document collected contains proclamations, laws, directives and training manuals. For testing purpose, documents are pre-processed using sentence extractor, character normalization, sentence/ paragraph tokenization, lexical analysis with the aim of treating digits and punctuation marks, stemming and also synonym indexing is used in place of WordNet.

A definition question is not a fully formed question, like a factoid question; rather it is just the name of the thing the user wishes to define. Just like factoid questions the target of the definition has to be analyzed to enable relevant documents to be located [9].

The system is composed of two components: indexing and definition searching. Under indexing there are subcomponents for pre-processing of the corpus which deals with the language dependent task which are listed above and extraction of all possible definitions from the Amharic legal corpus. Definition searching component consists of question analysis, which extract the definiendum from natural language question and retrieve the possible definition. In this component, lexical patterns, that are constructed manually, are used to identify the candidate definitions from the selected snippets.

Since the first step of every QA is question analysis this system also follows this approach. So the result of this module is identification of the definiendum. To return the answer for the definition question, the indexed file, which contains concept-description pair, is opened and search for a match between the concept and definiendum is done. If the definiendum is found in the dictionary catalog it display the answer otherwise it display an error message [9].

Definition type questions require word meaning, term definition, and description of term [9]. So to provide full information, full sentences are extracted as answers for users' natural language

query. To identify defnitions with their descriptions hard pattern matching is used. In addition, definition signal words at the end of the sentence like ናቸው፣ነው and ይተረጎማል are used to identify the concept- description pairs.

The evaluation of the experiment was done directly by ten native Amharic speakers who presented two questions each to the system and was judged by legal experts, for the correctness of the answers. Result shows that normalization and stemming increase the effectiveness of the system significantly. On the average, a precision of 85.6%, a recall of 73.0% and 78.8% F-measure is achieved.

Finally, the researcher recommended that the development of a standard Amharic NER, stemmer, WordNet, spell checker; preparation of standard corpus can improve the performance of a question answering system. In addition, integration with query expansion techniques will help improve the performance. He also noted that it is possible to construct an open domain QA and integrate to a search engine.

2.8.5 Amharic QA for Definitional, Biographical and Description Questions

Amharic non-factoid questions are questions with complex answers most of them require definitions, descriptions, reasoning or procedural explanations. Amharic definitional, biographical and description questions cannot be answered by Amharic factoid question answering because the answers are not only entity names and require complex question processing.

The design and implementation of the system explored in this research comprises four major phases, which are document preprocessing, question analysis, document analysis, and answer extraction. Document preprocessing includes character normalization, short word expansion, stop word removal, stemming, morphological analysis, and indexing using Lucene API. Question analysis includes question classification, query generation, and query expansion. Document analysis contains document retrieval and document filtering. The fourth phase in the design process is answer extraction [27].

The question classifier subcomponent of the system identifies the type of a question as biography, definition or description. Rule-based and machine-learning approaches are used for identification of the type of question. The rule-based algorithm determines the question type by using the interrogative terms of the question and class indicative terms which can be used as a clue to indicate the class of the question type. The machine learning approach predicts the type of questions based on training models. The training and classifying tool used is Support Vector Machine algorithm (SVM).

Since there are three question classes used in this research, the SVM classifier cannot be directly used to identify the question type. For this reason, one class with other class technique is used to train the classifier and prepared three question groups i.e., definition versus biography, definition versus description, and description versus biography then; it constructs three models based on the three grouped data. Testing is done with the three generated models and as a result, the classifier returns a number that shows to which question type the classifier classifies the given question. Then the total of the three numbers are generated.

Definition and description questions require complex response which is a short paragraph which clearly defines the target or state concepts the user wishes to know more about [27]. Thus for the answer extraction component finding snippets or piece of information about the current target, ranking, selecting, and ordering them is very important. In order to perform these tasks, the definition-description answer extraction is done. Then according to the question type the snippet extractor sub-component extracts sentences from the tokenized sentences using manually crafted indicative patterns for definition and 5 rules for description questions are crafted by inspecting different Amharic definition and description bearing documents [27]. Gazetteer based NER is also used for answer extraction.

For biography questions whose focus is a person, important dates in their life (birth, marriage, and death), their major achievements and some other items are noted as ‘correct’ answer. Therefore, to generate an answer biography answer extraction component first merges the filtered documents, summarizes the merged document, and validates the summary. Finally, if the result seems valid, the summary is displayed as an answer.

In order to construct the final answer selection subcomponent ranks the sentences by their score, selects the top ranked definition/description sentences according to the definition/description length requirement which is 5 sentences for this research. In Biographical questions answers are selected if the validation of the summary approves.

The performance of the SVM based question type classifier is 83.3% and that of the rule based question type classifier is 98.3%. The answer extraction component is evaluated by 120 test questions and 300 documents. The average recall, precision, and F-score of the answer extraction component are 0.609, 0.683, and 0.592 respectively.

Finally, the researcher recommended that developing tools that are used for anaphora resolution, lexical chains and discourse structure would keep coherence since different documents are analyzed. For extraction of snippets well development of semantic analyzer will help increase performance. Generating lexical patterns using machine-learning algorithms, development of standard corpus, stemmer, morphological analyzer, POS, NER and Amharic spell checker are also recommended by the researcher. The table below summarizes the survey presented above:

Author	Title	Problem	Method/tool	Result
Seid Muhie[6]	Amharic question answering for factoid questions	The question and answer construction of Amharic is different from English and other languages and need a special consideration on the question answering system.	Java programming language to develop algorithm using the eclipse java editor, gazetteer used instead of named entity recognizer, Lucene indexer for preprocessing.	89% of questions are classified correctly and 72% of user questions answered correctly.
Brook Eshetu[8]	Amharic question answering for list questions	Users struggle while looking for answers in Amharic using IR systems because of information overload problem, previous work done on factoid questions used a handcrafted classification algorithm.	Machine learning approach using SVM classifier algorithm, answers are extracted using the tendency of co-occurrence of candidate answers.	The overall recall of the system is 0.636, 57.5% of F-score is gained.
Desalegn Abebaw[7]	A web based question answering for factoid questions using machine learning approach	The previous work on factoid question was a closed domain and handcrafted rules; due to the richness of the web it is more feasible to design a web based question answering tool.	J-spider crawler to retrieve web pages, SVM classifier algorithm.	94.2% accuracy in classification and 77% correctly answered
Wondwossen	Designing	Previously constructed QA	Synonym indexing	Precision

Teshome[9]	Amharic definitive question answering	systems for foreign languages cannot be applied for Amharic and the existing IR systems are not that much helping in getting precise answers for users' questions.	used in place of WordNet, sentence extractor and character normalization used for preprocessing, pattern matching is used for answer extraction and python 2.7 for implementation.	85.6%, recall 73%, F-measure of 78.8% attained.
Tilahun Abedissa[27]	Amharic question answering for definitional, biographical and description questions	Amharic definitional, biographical and description questions cannot be answered by Amharic factoid questions.	Lucene API for indexing, both rule based and SVM based approaches for question classification, text summarizer for biographical questions.	SVM based classifier 83.3%, rule based 98.3%, average recall 0.609, precision 0.683, F-score 0.592

Table 2-1 Summary of local related works

2.9 QA Research in Other Languages

2.9.1 QARAB: A Question Answering System to Support the Arabic Language

QARAB is a system that takes natural language questions expressed in the Arabic language and provides short answers to users. Corpus was collected from a newspaper called Al-Raya which is published in Qatar. The system has two parts: the IR system and the NLP system [28].

The IR system treats the question as a query that identifies the candidate documents that may contain the answer. The IR system first processes the text collection from the Al-Raya newspaper and constructs an inverted file system, from which the answers to the natural language questions will be extracted. The purpose of the IR system is to search the document collection to select documents containing information relevant to the user's query [28].

The NLP system is composed of a set of tools to tokenize and tag Arabic text and to identify proper names. The NLP techniques are used to parse the question and analyze the top ranked documents returned by the IR system.

The basic process in QARAB is composed of three major steps: processing the input question, retrieving the candidate documents (paragraphs) containing answers from the IR system, processing each one of the candidate documents (paragraphs) in the same way as the question is processed.

QARAB considers users' question as a "bag of words" against which the index file is searched to obtain a list of ranked documents that possibly contain the answer. The question processing begins by performing tokenization to extract individual terms. The stop-words are removed; remaining words are tagged for part-of-speech. The main effort taking task is identifying proper names because there is no capitalization in Arabic language. The interrogative words determine the type of answer we should be expecting [28]. Input to the Answer Generator is the "bag of words" and the paragraphs extracted from the ranked documents. Then it looks for keywords that identify a person name using personal names key words.

The overall success of the system is limited to the amount of available tools developed for the Arabic language.

2.9.2 IDRAAQ: Query Expansion and Passage Retrieval based QA for Arabic

The IDRAAQ system is a participant of a campaign called question answering for machine reading evaluation (QA4MRE) at CLEF 2012. The campaign represents an evolution of evaluation approaches in NLP. The test set is composed of 4 topics as a domain namely Aids, climate change, music and society and the fourth one is Alzheimer.

The IDRAAQ system is fully programmed in Java and designed based on the typical modules of a QA system. Question analysis and classification analyzes questions to extract key words, identify structure of expected answer and form a query to pass to the next module. Passage retrieval uses query passed from previous module and extract passages from information retrieval process and is the most important component of the system. Then answer validation module validates an answer from a list of candidate answers relying on passages that are provided by the previous module [29].

The performance of IDRAAQ system depends on the passage retrieval module as it provides candidate passages to the answer validation module for the answers to be selected. The passage retrieval (PR) module of this system is formed by two implemented levels: keyword-based level and structure-based level. The former integrates a semantic query expansion process and keyword is replaced by its synonyms in the WordNet while the latter uses a Distance Density N-gram based PR tool for re-ranking passages also new queries are generated[29].

For evaluation purpose, the type of questions used was factoid, causal, method and purpose. Moreover, the system was evaluated based on overall accuracy. From the experiment, most of the answered questions are factoid ones (When, Who, What, etc.) but failed in list and reason questions. This shows that using Arabic WordNet mapped with a system that contains high number of Named Entities has a positive impact on system performances especially when processing factoid questions.

2.9.3 Automatic Answering System for English Language Questions

The system at [19] tries to solve both open and closed domain problems for English language. The system is developed to enable users to ask question from a mobile phone and it is also designed to understand SMS language.

The overall architecture of the system is subdivided into three parts namely: pre-processing, answer discovery and answering. The pre-processing part contains two operations: one converting SMS abbreviations into general English words by referring a previously stored frequently used SMS abbreviations and the other is removal of stop words and stem words as this operation increases effectiveness of the system by saving time and disk space. For the stemming purpose Porter stemming algorithm was used.

Question –template matching is used for Answer discovery module. Templates are created for questions according to a specific syntax; words with synonyms are referred in a synonym file, which can be modified whenever it is necessary. In addition, understanding of the problem domain is not required for development of the system. A technique for enhancing the template matching is used throughout the system and disemvoweling is one of the techniques. Most of the spelling mistakes in English are said to occur because of omission, addition or out of order vowels [19]. So removing vowels will reduce amount of spelling mistakes and is used in this system for this purpose.

If users ask a question out of closed domain, search engine will search for an answer and return to the user. For extraction of a web data JSON is used which is a universal language independent format for data which translates html in to another format to make it easier for extraction.

After the answer discovery module, each template representing a question is stored in a database with its corresponding answer, when best match is found the answer will be returned to the user through SMS. NowSMS which is an SMS content delivery solution is used in the system for sending SMS to users [19]. The result of this system was represented as a smart and user-friendly automatic answering system, which is capable of answering any question forwarded in English or SMS languages.

Chapter 3

3 The Amharic language

3.1 Introduction

Amharic language is the working language of the Federal Government of Ethiopia and the second most spoken Semitic language next to Arabic. It has been a written language for about 500 years. It is written in a style of script known as the Ethiopic alphabet, which is composed of seven vowels, and 33 consonants that are always combined including ejectives that have no sounds in English language. The consonant sounds /p/, /t/, /k/, and /s/ can be produced as ejectives [30][31].

The Amharic language has been declared to have word categories as ስም(noun), ግስ (verb), ቅፅል(adjective), ተውሳከግስ (Adverb), መስተዋድድ(preposition), and ተውላጠስም (pronoun)[9].

Nouns in Amharic represent gender, number, definiteness, case, and direct object status by affixes prefixes and suffixes, mostly suffixes[30]. Amharic nouns may have a masculine or feminine gender. Suffixes are added to denote a masculine or feminine noun gender. Some nouns may have both masculine and feminine gender, while other nouns may only have one gender. The feminine gender is used to indicate female as well as smallness. For example, ቤቱ ትንሽ ነች፡፡ Plurals are formed by adding ዎች or ኦች whether the word ends with a vowel or consonant.

Verbs are words derived from roots and affixes to inflect person, number, gender, mood, voice, and polarity. Verbs agree with their subjects. Verb agreement with objects is optional. Verbs in Amharic are mostly placed at the end of the sentence.

Adverb: it can be used to qualify a verb by adding extra idea on the sentence. The Amharic adverbs are limited in number and include ትላንት፣ ዛሬ፣ ገና፣ ቶሎ፣ እንደገና …

Adjective: any word that modifies a noun or an adverb, which actually comes before a noun ነበዝ ተሜ፣ በጣም ነበዝ. Other specific property of adjectives is, when pluralized, it repeats the previous letter of the last letter for the word (e.g. ትንሽ፣ ትንንሽ).

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. It includes ስለ፣ ወደ፣ እንደ፣ ከ…

Pronoun: this category further can be divided as deictic specifier, which includes እሱ፣ እኔ፣ አንተ፣ አንቺ፣ እነሱ…; quantitative specifier, which includes ጥቂት፣ አንዳንድ፣ አንድ…; and possession specifier such as የአንተ፣ የኔ፣ የሱ…

3.2 Sentences in Amharic

A sentence in Amharic can be a statement which is used to declare, explain, or discuss an issue; an interrogative sentence which can be used for questioning; exclamatory and imperative [9]. Sentences are constructed from the noun phrase and verb phrase combinations. The noun phrase and the verb phrase further divided to different particles such as other sub noun phrase and verb phrase, noun, adjectives and so on. The interrogative sentences also have the same structure with little modifications and introduction of question particles (interrogative words).

The verb goes at the end of the sentence in subject/object/verb (SOV) order where in English the word order is SVO. The sentences are short in number of words they contain because they have lots of prefixes and suffixes. For interrogative sentences, if it is a yes or no question, the sentence order stays the same but the intonation is that of a question and no question word is used. If not the question word is placed right before the verb.

3.3 Amharic Punctuation Marks and Numerals

The Amharic documents collected must be pre-processed before proceeding to the other components of question answering system. Sentence, paragraph, and document indexing all utilize different punctuation marks for separating one from the other [6]. In Amharic, there are different punctuation marks used for different purposes. For example, sentence indexing will be done with the help of the Amharic full stop (።) for separating different sentences. So the system should understand this punctuation mark indicates the end of a sentence. In the old writing systems, a colon (two dots ፡) has been used to separate two words. These days the two dots are replaced with whitespace. In addition, ነጠላሰረዝ (፣ or ፥) is also used to separate lists of ideas just like the comma in English.

Similarly, numerals have greater impact on Amharic question answering systems. Since numbers are stored in different formats, some kind of standardization should be applied to help the retrieval technique and searching. Ethiopic and Arabic numbers should be normalized to the same standard to make a document suitable for the retrieval component. In Amharic, numbers can be represented using Arabic symbols. It has also its own number representations, Ethiopic number representations. Similarly numbers can be represented in a word alphanumerically which is representation of numbers in words [9].

3.4 Question and Answer Formation

Questions might be raised to know what we do not know or to make sure that we know something. To ask questions that we totally don't know about in Amharic, we use adverbs and to be sure about what we know we use other words or tones that indicate the sentences are interrogative [32].

ካሳ መቼ መጣ? When did Kasa came?

ካሳ መጣ ወይ? Did Kasa come?

ካሳ መጣ? Did Kasa come?

The first example shows that the interrogator does not know when Kasa is coming so will add the adverb when (መቼ). Such kinds of questions expect answers that indicate time such as

yesterday or today. In the second example, the interrogator knows that Kasa is coming but wants to be sure. In this sentence the last word ወይ indicates the sentence is interrogative and it can be used to be sure about something also it is always placed at the end of the interrogative sentence. The third sentence has no interrogative word the only thing that indicates it is a question is the question mark and the sound (tone) of the interrogator and the way of the sentence is read[30].

Questions might be raised about some action or condition, about the performer of an action, about the agent that performs the action or time and place, about the cause of the action or aim of the action, how the action is performed or techniques used to perform the action, and so on[30]. All of these types of questions might be within a single sentence.

In every language, questions are constructed with the help of interrogative words and a question mark (?), which is placed at the end of the question. The question mark is kept at the end of the statement, indicating that the sentence is a question. The interrogative words in Amharic are placed near the end of the sentence most of the time. Most of those words are multipurpose where a single interrogative word is used for different types of question formations. There should be extra information to determine the question type besides the interrogative words such as question focuses or complex grammatical structure analysis of the sentence [8].

In English, the interrogative words (WH words) who, what, where, when, why, how … are used to construct a question. There are a number of interrogative words in Amharic that will help in constructing a question [8]. The well-known Amharic interrogative words are shown below:

ማን, ለማን, ማ ው; እነ ማን, ማንማን, ማንን, ሞ ማን, እነ ማንን, የ ማን, ከ ማን ኛው;

ማን ኛው; ማን ኛይቱ, ማን ኛዋ, ማን ኛ ቸው; በ ማን,

ተና ገ ር, ጥቀ ስ, ግለ ፅ, ዘ ር ዝ ር, ጥራ,

የ ት, የ ቱ, በ የ ት, የ ቷ, የ ቲ ቱ, የ ቶ ቹ, ወደ ት, ከ የ ት ኛው; የ የ ት, የ ት ኛው; የ ት ኛዋ, የ ት ኞ ቹ,

ከ የ ት, እስ ከ የ ት, ወደ የ ት, የ የ ት ኛው; በ የ ት ኞ ቹ

ምን ,ምን ድን, የ ምን, ምን ህን, ምን ሽን, ምና ችሁን, ምዜ, ም ኑን, ስ ለ ምን, እን ደ ምን, ለ ምን,

በ ምን, እስ ከ ምን, ከ ምን, ወደ ምን, ምን ጊ ዜ, ምን ያ ህ ል,

መቸ, በመቸ, እስከመቸ, ለመቸ

ስንት, ስንቱ, ከስንት

From the words listed above, some of them are used for list questions and others are used for factoid questions.

There are different challenges in Amharic question answering. One of the main problems is that question particles (interrogative words) may not help in determining the question type. For example the question particle ማን might indicate a person or an organization. So in such kinds of cases extra analysis is required to determine the question type, to know the expected answer types. Secondly, some proper names belong to more than one word categories, such as verb and noun (e.g. "በቀለ" might be a person name and at the same time it has a context of growing up) so that determining whether that word is the expected proper name or not is very difficult. This problem sometimes increases, as there is no proper name capitalization in Amharic.

Chapter 4

4 Design of Amharic Question Answering

4.1 Introduction

As this research follows design science research methodology, it tries to create some artifact that define the ideas and products through which the analysis, design, implementation and use of the system being built can be effectively accomplished[33].Design science creates and evaluates artifacts that are intended to solve the identified problems. The result of a design science research is an artifact developed and implemented to address some problem. The design science process includes six steps: problem identification, definition of objectives for a solution, design and development, demonstration, evaluation, and communication[34]. This research takes these steps listed above in to consideration.

4.2 Problem Identification

Problem identification is the first step in design science research. It is a process of identifying key areas and needs consideration while doing a research on some topic. To identify and understand the problem we are working on, we did an observation and a literature review on the topic. Relaying on the problems identified through observation, asking people some questions about the study and literature review we attempted to design a solution out of them.

4.3 Definition of Objectives for a Solution

After identifying the problems one is working to overcome, the objectives of the research should be well defined. Then knowing our objectives, we can design and implement the study we are working on. Therefore, after having the problems identified, the objectives of this study are defined well and keeping these objectives in mind, we continue to the design and development of the study.

4.4 System Design and Development

4.4.1 System Architecture

For designing the architecture, the architecture in [14] is adopted and some of the components such as the POS tagger and the NER are included since these components are included in this study. This architecture can be used for both the factoid and list questions since the components

of both types of QA are the same. The differences are only in the extraction of answers where list QA uses pattern matching and co-occurrence.

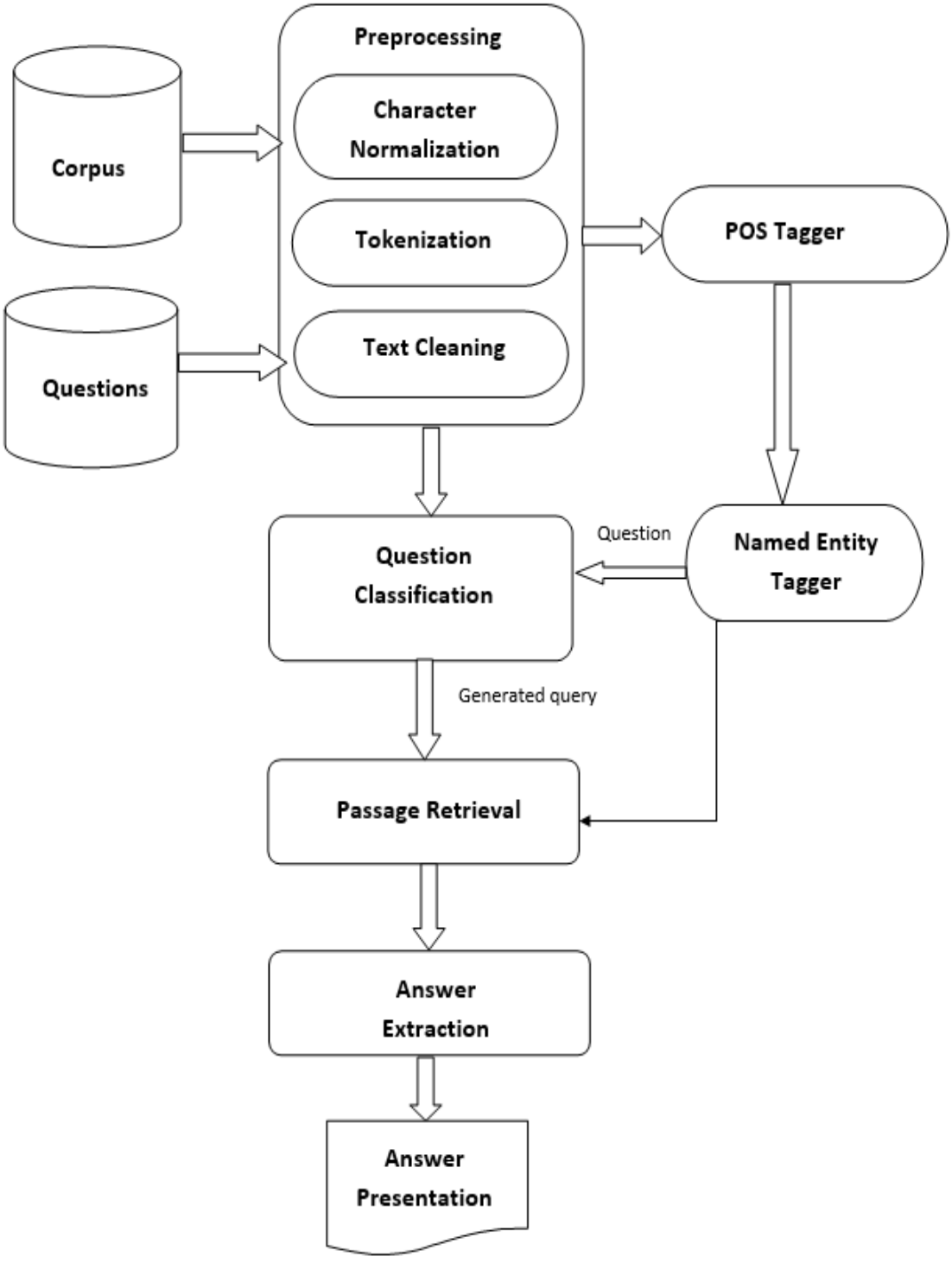


Figure 4-1: Architecture of Amharic question answering

4.4.2 Dataset Preparation

Before getting to the design and development steps, we need to perform corpus preparation, which is the first thing in the architecture of the system. The corpus used for this study is only concerned about Ethiopian history as the research focuses on a specific domain or closed domain question answering. We collected a 5000 sentence corpus from different Amharic history books, newspapers that have a history section, electronically available history books and from the web.

4.4.3 Preprocessing

After the data collection, preprocessing is the next thing that follows in order to make the corpus that we have on hand in the appropriate format and to make it easier for the machine to read. The preprocessing steps performed for this research include character normalization, short word expansion, tokenization, stop word removal at the passage retrieval phase and text cleaning. In addition, other tools such as POS tagger and NER are used before getting to the question answering components.

Character Normalization: some Amharic alphabets use the same pronunciation with the same meaning but have different symbols and are used interchangeably most of the time[30]. Some of the characters such as አ and ዐ, ሀ, ሐ and ኀ, ሰ and ሠ, ጸ and ፀ are the Amharic alphabets that are used interchangeably in a single word for example the words አለም and ዐለም, ሰው and ሠው have the same meaning of world and human/ person respectively but the alphabets are used according to an individuals' interest. Therefore, we must normalize these characters to a specific form of alphabet in order to make the machine understand each meaning.

For the normalization of those characters mentioned above, we used a python code that replaces the characters with the same meaning and come up with one representative. The figure below shows the python code we used for normalization.

```
# -*- coding: utf-8 -*-
# python

# find and replace many pairs of strings in a utf-8 file

filePath = "/home/medi/codes/lelatarik.txt"

outPath = filePath + ".new"

findReplacePairs = [(u"ፀ",u"ጸ"), (u"ፁ",u"ጹ"), (u"ፂ",u"ጺ"), (u"ፃ",u"ጻ"), (u"ፄ",u"ጼ"), (u"ፅ",u"ጽ"), (u"ፆ",u"ጾ"), (u"ሐ",u"ሀ"), (u"ሑ",u(u"ሒ",u"ሂ")
(u"ሕ",u"ህ"), (u"ሖ",u"ሆ"), (u"ዐ",u"አ"), (u"ዑ",u"ኡ"), (u"ዒ",u"ኢ"), (u"ዓ",u"ኣ"), (u"ዔ",u"ኤ"), (u"ዕ",u"እ"), (u"ዖ",u"ኦ"), (u"ሠ",u"ሰ"), ,u"ሡ"), (u"ሢ
(u"ሥ",u"ስ"), (u"ሦ",u"ሶ"), (u"ኀ",u"ሀ"), (u"ኁ",u"ሁ"), (u"ኂ",u"ሂ"), (u"ኃ",u"ሃ"), (u"ኄ",u"ሄ"), (u"ኅ",u"ህ"), (u"ኆ",u"ሆ"), (u"ኸ",u"ሀ"), ,u"ሁ"), (u"ኾ
(u"ኽ",u"ህ"), (u"ኾ",u"ሆ")]
inF = open(filePath, "rb")
fContent = unicode(inF.read(), "utf-8")
inF.close()

for pair in findReplacePairs:
    outText = fContent.replace(pair[0], pair[1])
    fContent = outText

outF = open(outPath, "wb")
outF.write(outText.encode("utf-8"))
outF.close()
```

Short word expansion is performed on those words that are written in short form for example school ት/ቤት is expanded to ትምህርት ቤት to make it easier for reading by the analysis component. In Amharic historical writings short words might appear frequently such as; ቤ/ክርስትያን should be expanded to ቤተክርስትያን, ብ/ጀነ ራል to ብርጋዴር ጀነ ራል, አ/ም to አመተ ምህረት. Therefore, to make the corpus available in a readable format short word expansion is performed manually.

Sentence tokenization is also another preprocessing technique performed since we need a single sentence on a line for analysis purpose. Tokenization is done by using different sentence demarcations such as question mark and full stop. For our study we used the Amharic full stop (: :) to indicate a sentence end. Whenever the tokenizer gets an Amharic full stop it splits the other sentences into a newline. This process will help the retrieval process to get key words

easily. Word tokenization also might appear important to some tasks such as NER because a word-tokenized data is provided to the named entity tagger.

As mentioned in the above paragraph, we used the Amharic full stop to split sentences. In the tokenization process a python code is used. The code takes a full text as an input and tokenizes it in a sentence level by looking for the sentence demarcation. In addition, word tokenization is also performed by using a white space. The figure below shows a python code we used for word tokenization.

```
import codecs
import sys
''' Run the programm from a command line using *** python word_tokenize.py file_name
'''

def tokenize(cont):
        tokenized = ''
        word_list = []
        for line in cont.splitlines():
                for word in line.split():
                        word_list.append(word)

        print '%d number of lines read ' % (len(cont.splitlines()))
        print '%d number of words found ' % (len(word_list))

        for word in word_list:
                tokenized += word +'\n'
        return tokenized


if __name__ == '__main__':
        file_name = sys.argv[1]
        fdata = codecs.open(file_name,'r','utf-8')
        cont = fdata.read()
        fdata.close()
        tokenized = tokenize(cont)
        new_file =  file_name + '_tokenized'
        fob =  codecs.open(new_file,'w','utf-8')
        fob.write(tokenized)
        print 'Tokenized written to %s ' % (new_file)
        fob.close()
```

Sentence cleaning is also important, as there are, some texts that need to be corrected in which there might be some spelling errors and other errors that should be corrected inside the data. Some indexes and symbols found in our corpus are removed. These and other preprocessing techniques and cleaning techniques were performed on the corpus to make it readable by the question answering components.

Stop word removal: this is done to make the document retrieval processes easier[6]. In this process, words that make no meaning change on a text are removed. There are lists of Amharic stop words stored in the system. Before getting to the passage retrieval component, the input question is checked for a stop word in the list. Then the stop word is removed from the question for generation of a query. Words like ናቸው; ናት, ነበሩ, ነው.. are considered as stop words and removed from the query that is generated.

POS tagging

POS tagging is an NLP tool that used to identify the part of speech of words. Once a user enters a question, answer type identification can be done using the part of speech of each word then the candidate answers can be reduced and ranked according to their relevance. As this tool helps in identifying answers quickly, researchers should consider it before entering to the other components of a question answering system. In addition, POS tagger is important for Named Entity Recognition as it helps the named entity recognizer to identify words that are nouns and assign the appropriate tag to each nouns that are named-entities.

The part of speech tagger developed for this study is a hidden Markov Model part of speech tagger using the tag sets noun, verb, adverb, adjective, preposition, pronoun, number, conjunction and punctuation[35][36]. In the process, the tagger adds these part of speech tags or other lexical class markers to each word in a text. Training data provided sentence-tokenized and manually tagged. It consists of a text in the word/TAG format with words separated by spaces and each sentence on a new line. The test data will be untagged development data provided tokenized with words separated by spaces and each sentence on a new line. Then the tagger adds the part of speech tags.

There are two programs in the POS tagger used for this research; One program learns a Hidden Markov Model from the training data and is invoked by using 'python hmmlearn.py /path to input/file containing training data'. The program will learn the model parameters to a file called 'hmmmodel.txt' as it is indicated in appendix F. The other program which is, the tagging program is invoked by using 'python hmmdecode.py /path to input/file containing test data'. Then the program will read parameters of HMM from 'hmmmodel.txt', tag each word in test data and write the results to an output file with the same format as the training data. The test data may contain words that the tagger never encountered in the training data, which will have an emission probability of zero for all tags. In addition, unseen transitions between two consecutive words have a probability of zero.

POS tagging is not useful by itself but is accepted as the first step to understand a natural language[36]. Other tasks such as question answering and NER heavily depend on it. NER needs POS tagged corpus in order to find the nouns and the question type identification needs a POS tagged data to know the type of the question.

Named Entity Recognition

NER is a very important tool that is incorporated with a question answering system as it is found to be important to recognize names of different entities in the corpus in order to achieve a good performance. Many of the fact-based answers are entity names that can be detected and extracted using NER. Making use of this tool makes the process of answer extraction way better and easier. The NER used in a QA system is developed as a stand-alone system designed independently of the QA task[37].Most QA systems gradually reduce contents that are not needed so, the NER is used to help in the reduction of unwanted strings that do not contain an answer. Development of an NER basically has two approaches Rule Based Approach and Statistical Approach. For this research, we used the statistical approach, which is based on training corpuses.

The data used in order to build this tool is a POS tagged corpus in which the word and tag are separated by a space (word TAG format) and tokenized at word level. Then the data used for

training will be prepared by replacing the nouns, which are named-entities by the appropriate tag. Since the NER is an HMM based there are some parameters used to model what the algorithm has learned. The parameters taken from the training file are the word and its POS frequency, POS and its frequency, the probability of POS transferring to the other POS and also the word and its probability of being the head of sentence as a POS. Then the named entity tagger will take the training data and the unseen test data, by checking for 'NN' (noun) in the test data and assigns the tags to the proper nouns using the parameters that are learned from the training data.

For tagging named entities that are composed of more than one words used begginig and inner entities in the training corpus. For example, the entity: "ዳግማዊ ምኒልክ ሆስፒታል" is tagged as "ዳግማዊ/B-ORG ምኒልክ/I-ORG ሆስፒታል/I-ORG" to represent a hospital and "ዳግማዊ አጼ ምኒልክ" is tagged as "ዳግማዊ/B-PER አጼ/I-PER ምኒልክ/I-PER" to represent Emperior Minilik.

Since question-answering systems gradually reduce the amount of data considered after receiving a question, it first selects relevant documents and filters out irrelevant pieces of text of these documents gradually until answer is found. So NER is used as an aid to filter out unwanted strings or strings that do not contain the answer.
As mentioned above we use NER to filter out candidate answers and it will remove only the wrong answers or part of the corpus that does not have much importance. Therefore, in this case recall will be preferred more than precision[37].

For this research we used an HMM based NER tool which recognizes the entities; person (PER), organization (ORG), date (DATE), time (TIME), location (LOC) and entity (ENTY) to represent other entities such as river, books and God which appeared in our corpus. The training data is annotated corpora with the entities listed above and is given to the training program and a test data of unseen POS tagged data that will be tagged using the program.

4.4.4 Question Processing

Different users ask questions in different ways. The ways might depend on the person asking. Before preceding to answer the questions, there must be some analysis in order to understand what the question is looking for. Therefore, this is why we need to perform question processing

in order to understand the question before going to the other steps of question answering. The question processing contains question analysis, classification, question type identification and query generation as its components.

4.4.4.1 Question Analysis

According to the architecture of a question answering system, the first component is question analysis and classification, which is a sub component of question processing. Question analysis identifies the type of question from the accepted questions and type of answer that is retrieved finally. Question type identification is performed using the Amharic factoid questions (ማን, ምን, ስንት, የት, መቼ, etc...) and list questions (ጥቀስ, ዘርዝር, የትኞቹ) as they are the question words used throughout the question answering process. For example, a question that contains the word where/የት is identified as a location question. This step also helps for the answer type identification by looking at the pattern of the sentence in the corpus.

4.4.4.2 Question Classification

Question classification is one of the sub components of question processing and is very important to identify the type of questions in order to know the type of answer expected. Question classification is performed using the SVM and HMM classifier. For training the classifier six classification models were generated (person/PER, location/LOC, organization/ORG, time/TIME, quantity/NUM, entity/ENTY, date/DATE, list/LIST) are developed and stored for use in testing the classifier.

One of the algorithms used in classification is, support vector machine (SVM). To apply this algorithm set of training examples containing feature vectors of candidate answers should be prepared. For the classification purpose n number of labeled questions containing the five classes of factoid questions (person, time, place, date, quantity and organization) and the list questions are given to the program that separates out the label or class and the question separately. Then the questions are put in the classes they belong (such as PER, LOC, ORG etc…) in separate files then the other program extracts word features, POS features and NER features and after that the classifier checks for POS tag and NER of the words in the training questions and use them to classify the unseen test questions using the SVM classifier.

The other classifier used in addition to the SVM classifier is an HMM classifier. The classifier is trained using questions labeled with the factoid questions we used and the list questions with the question words mentioned in the above section. The training program builds a model which will be used to classify the unseen data. Then 20% of the question we prepared is used for testing the classification. Some of the contextual meanings of the questions appear different from the question words in Amharic for example "የ ወጫሌ ወልፀ የ ተፈረመው በ ማን እና በ ማን መካከል ነ በር?" is asking the countries that signed the Wuchale treaty but the question type is classified as PERSON. Such cases could result the wrong answer to the question.

Classification of list questions is handled by checking a question if it is of a list type or not. The classifier is trained for list type questions with the question words ጥቀስ, ዘርዝር, የ ትኞቹ, ማን ማን and የ ት የ ት then it will identify the type of question as a list question when it encounters these question words.

4.4.4.3 Query Generation

Once question types, question focuses and expected answer types are determined, the next stage of question processing is to generate the proper query that will help in retrieving relevant documents. The query generated, which is based on the question types and the expected answer types, will be passed to the document retrieval component. This stage incorporates stop word removal and character normalization to make the retrieval process easier. If character normalization is not considered at query generation step it reduces recall as there would be unmatched documents with the query[9]. In addition, short word expansion and removal of interrogative words are important. For example, if the question is "የ አጼ ምኒሊክ ልጅ ማን ማን ናቸው?" Then the generated query will be "የ አጼ ምኒሊክ ልጅ". Then the retrieval component further uses the queries generated for extraction of answer bearing sentences from a text file.

4.4.5 Document Retrieval

Document retrieval component return list of ranked documents, which are relevant to the asked question, from the corpus and passage. The first step in this phase is document-based retrieval. This avoids the amount of texts that should be handled sequentially. Only the best n documents are used for retrieval. Next is the sentence selection step, which retrieves the relevant sentences.

Since document based retrieval do not always return the target candidates, the retrieval technique used for this study is more of sentence-based retrieval.

From the relevant documents found by the first phase, all sentences are ranked against the question by using tfidf. Only the most relevant sentences that is the most similar sentence to the query is returned according to their score. After having the relevant sentences, the system searches for the exact answers. Some of the sentence selection methods, while computing the distance, already find some possible answers. The remaining sentences after the sentence selection are checked for named entities and possible answers are found using these named entities then, answer selection will be performed[37]. For the answer selection, question is analyzed to determine the type of answer expected then the best answer with the highest score and matching answer type will be returned.

4.4.6 Answer Selection

After getting the most relevant sentences, we perform answer selection by using key words selected from the question. Since the type of question and type of the expected answer is already identified, answer selection component selects the appropriate answer by matching the key words with named entity tagged corpus. As mentioned in the above sections NER is important in question answering because it reduces the number of relevant sentences and words and limits the answer to the question focus. So answer is selected from a sentence if there is a good match with a query term and a sentence after the most relevant sentences are selected.

As mentioned in the architecture section, the system architecture of QA for factoid and list questions is the same. However, list questions differ in answer extraction techniques. To select answers for list questions, candidate extraction is performed by using pattern matching where, answers are identified depending on the similarity between the patterns of key words in the question and inside the retrieved sentence. After candidate answers are selected from the passages retrieved, the system filters the answer by co-occurrence extraction and candidate selection based on co-occurrence of answers and answer type. Then answer with the highest priority will be presented to the user.

4.5 Approaches Tools and Techniques

As we have reviewed in the chapter two of this study there are different types of QA, approaches and algorithms used in the system, tools and techniques. In this section, we discuss the different techniques we use to deploy our system.

4.5.1 Approaches

The approach selected to this particular research is a machine learning approach as it can automatically construct a question classification program with a high performance that can be used for more features of questions. For some tasks such as identifying the question type, machine-learning classification can be taken as classical classification tasks that can be solved by SVM and other algorithms. Machine learning approach is used in a question answering because it is more flexible in classifying questions and is easily maintainable[21]. From the possible classifier algorithms, we reviewed and used SVM and HMM for this particular research.

HMM has 3 parameters: Start Probability, Transition Probability and Emission Probability[39]. Start Probability means the probability that the type that is labled as a tag exists first in a sentence. Transition Probability means the ratio of probability of transition from given tag to the next tag and the Probability of occurrence of a given tag. Emission Probability is the ratio of probability of occurrence of particular word with tag t and the Probability of occurrence of a tag t[39].

SVM: the basic idea of SVM is to maximize the margin between positive (correct answers) and negative (incorrect answers) and mostly used in question answering for the purpose of answer selection. To apply this algorithm set of training examples containing feature vectors of candidate answers containing the key words, question type, question focus, question words, named-entities are prepared.

4.5.2 Tools

Some of the tools considered for this research are POS tagger and NER explained in the design section. These tools are very important for answer extraction in QA. Python programming language used for the preprocessing and other tasks such as for developing the POS tagger and NER, for the question classification, passage retrieval and answer selection tasks.

4.6 Evaluation

Automatic evaluation of a question answering can be done using the standard evaluation techniques such as precision, recall and F-measure. Precision measures the accuracy of an answer, recall measures the exhaustivity[26], and F-measure is the harmonic mean of Precision and Recall.

Evaluation of this study is mostly for accuracy or correctness of answers. Precision is calculated as the number of correctly answered questions over the total list of answers (correct, incorrect and no answer). The recall is also calculated as number of correctly answered questions among the list of expected answer sets where documents are first checked for the presence of correct answers[6]. Percentage computation is done for correct answers, wrong answers, and No answers over the total answers which is the main evaluation criteria for many QA systems[6].

$$Precision = \frac{Correct\ answers}{Correct\ answers + Wrong\ answers\ + No\ answers}$$

$$Recall = \frac{Correct\ answers}{Correct\ answers + Missed\ answers}$$

$$F - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Chapter 5

5 Experiment and Evaluation

5.1 Experimental Setup

To make the experiment running we collected around 5000 sentences from different sources that cover Ethiopian history. Then, as shown in appendix A we prepared around 118 questions that comprises both factoid and list questions for testing the question answering. Out of the whole questions, 21 are list questions and the rest are factoid questions. After the question preparation the first task was question classification using two algorithms; one is done using the SVM classifier and the other is using the HMM classifier. Then passage retrieval was performed using tfidf to return the relevant sentecnes. The third task was answer extraction from the top ranked sentence using NER. Then evaluation was done after both of the classifications.

We have used Ubuntu 16.04 as an operating system for development and testing. The hardware component comprises of Intel core i7 CPU of 2.5 GHz, 8 GB memory, and 1TB hard disk.

The Python programming language (python 2.7 and 3) is used for developing the tools used in this study such as the POS tagger and the named entity recognizer. In addition, the question classification, passage retrieval and answer extraction components are implemented using python.

5.2 Evaluation of POS and NER

As explained in the previous chapter an HMM POS tagger is developed to be used by the NER and the question classification component. We trained the POS tagger with a sentence tokenized and manually tagged training corpus and tested with a sentence-tokenized unseen test corpus, which is 20% (1000 sentences) of the prepared corpus. Then the performance of the tagger is evaluated automatically using accuracy and we come up with a 76% accuracy. The researcher believes that the performance is reduced because it needs a large sized data for training. In addition, the quality and type of the data also has an impact on the performance of the tagger. Since the corpus is on Ethiopian history, there are different nouns referring different entities and years representing events that occurred in the history of Ethiopia included inside it. After getting

such a result, we have corrected some of the sentences manually in order not to affect the next step (the NER).

The other tool used after the POS tagger was the NER as it is the most important tool in answer extraction. A POS tagged and word-tokenized data is given for training and a POS tagged unseen data was used for testing purpose. The NE tagger was trained using around 43,747 tokens and 11,500 tokens were used for testing. Then we got a precision rate of 72%. The quality of the corpus we used has an effect on the result; for example, an automobile was written as "አውቶ በል" and the NE tagger could not tag it with the appropriate class of named entities. The performance of the POS tagger and the type of data for example: a name of a church that is tagged as an organization, might also be a name of a place or a town that is tagged as a location affected the precision rate of the NER. Since the corpus is a historical document there are different names of people, places and organizations that are not encountered in the training data and the NE tagger might just tag them as a noun or give them a wrong tag.

5.3 Answer Judgment

Evaluation of a question answering systems focuses on accuracy of answers returned. And accuracy might be of different types as explained in [6]. Some of the question answering systems might expect a full sentence as an answer. Other systems require exact answers with one word or two. For example, for the question "የመጀመሪያውን ስልክ ወደ ኢትዮጵያ ያስገባው ማን ነው?" the answer is retrieved as "ራስ መኮንን". The full statement of the answer is "የመጀመሪያውን ስልክ ወደ ኢትዮጵያ ያስገቡት ራስ መኮንን ናቸው" but in our case, as mentioned above the returned answer is just an exact answer.

Evaluation could also contain how much of the questions are answered correctly, wrongly and how much of it has retrieved no answer. Some of the answers are retrieved as their similarity with the question type and named-entity with the asked question and end up being wrong. Some questions might not return any answer because of the corpus or the answer extraction.

The experiment in this study focuses on correctness and exactness by computing recall and precision. The actual performance of this system is evaluated by the number of correctly answered questions from the corpus. The first evaluation is done on the question classification phase as the more correctly classified the questions the more answer type can be identified easily

and the more correct answer is selected out of the prepared corpus. Then the document retrieval and answer selection phases are sequentially evaluated.

5.4 Evaluation of Question Classification

As mentioned in the previous section of this study, the evaluation of question classification is very important as it determines the type of the expected answer and the extracted final answer. If questions are classified wrongly, the retrieved answer will also be wrong.

The question classification with SVM classifier follows different steps. First, it separates out the types of the questions in to their classes, then it checks for POS tag and NER of the words in the training questions and use them to classify the unseen test questions using the SVM classifier. Number of total questions used in the classification is 118 where we used 95 of the questions for training and the rest 23 for testing. Given the training data, the SVM question classifier classified 88% of the questions correctly using automatic evaluation technique.

The HMM based question classifier is trained with 95 questions labeled with the types of questions (ማን, ምን, ስንት, የት, መቼ and list questions such as ምን ምን, ጥቀስ, ዘርዝር). Then the classifier takes the training data and checks the probability of the type of question words labeled according to the previous and next word then it will give the given label to the unseen test questions. Then the classifier classified 80% of the questions correctly. We evaluated the accuracy of the classifier using automatic evaluation technique.

Types of questions	ማን	ምን	ስንት	መቼ	የት	List questions
Distribution	38	6	10	19	24	21

Table 5-1Distribution of types of questions used in the experiment

Classifier	Correctly classified	Wrongly classified
SVM	88%	12%
HMM	80%	20%

Table 5-2 Performance of Question classification using SVM and HMM.

5.5 Evaluation of the document retrieval

As the documents we used for this experiment are sentence tokenized the retrieval process is based on sentence level. Moreover, we run the question answering by selecting the type of document, so there is no need of retrieval at a file level.

The document retrieval phase of question answering is also evaluated in terms of the relevant sentences that are ranked on the top. In the document retrieval process, the most relevant sentences are selected and the answer is further selected form the top sentence using NER.

Evaluation at a document retrieval level is important to check if the top ranked sentences or files contain candidate answers so in the case of this study 82 of the top ranked sentences contain the correct answers for the given questions and 36 of them do not contain correct answers. The remaining three correct answers were retrieved from sentences that contain the answers but according to the questions asked, the relevance of the top ranked sentences was not that important.

5.6 Evaluation of answer selection

To return an answer for a factoid question searching for a match in the NE tagged corpus will be performed using the key word in the input question. For list questions, the search will be for the co-occurrence of answer type with a candidate answer. If a match is found answer is extracted from the top relevant sentences using a named entity recognizer, then the correct answer will be displayed. Otherwise, nothing is returned. Sometimes a related sentence that could hold a key word is retrieved as an answer.

This evaluation considers the correctness or accuracy of the final answer. We perform answer selection based on the NER we have developed. The NER helps in getting the correct answer type by looking at the named entities and it really affects the recall of the system. After this

selection, answer was extracted and 85 out of 118 questions were answered correctly with in a response time of 2 seconds. Then we calculated the precision and recall rate manually and automatically.

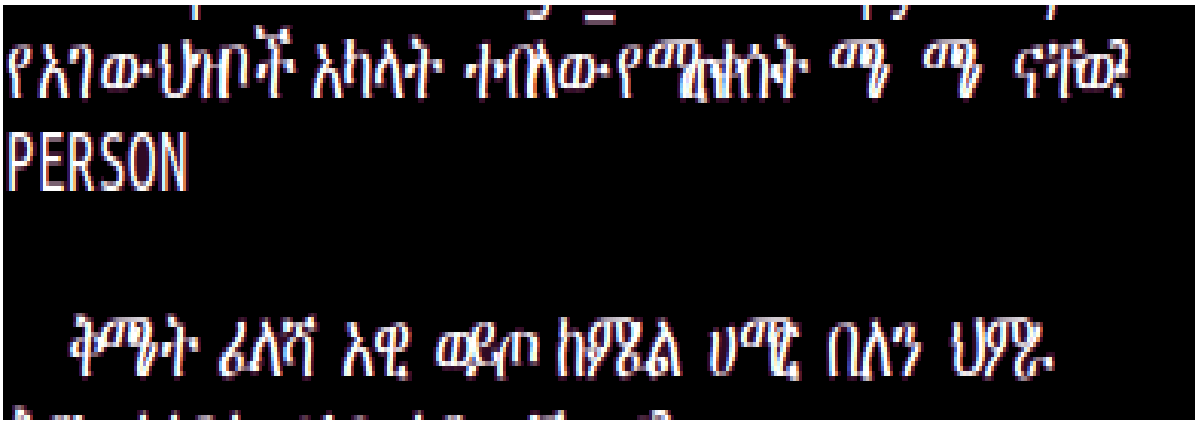
Correctly Answered	**Wrongly Answered**	**No Answer**	**Precision**	**Recall**	**F- Score**
85 questions	28 questions	5 questions	75%	72%	73%

Table 5-3 Evaluation of answer selection after question classification using SVM

The type of answer predicted using the SVM classifier helped the answer extraction well in order to identify correct answers. As the two tables indicate the answer extraction component performed well after the SVM classification than after the HMM classification. The HMM classifier adds the type of the questions to the question words by looking at the previous and next words. When it encounters different words around the question word, it behaves differently. The researcher believes that the performance is lower because of this reason.

Correctly Answered	**Wrongly Answered**	**No Answer**	**Precision**	**Recall**	**F-Score**
75 questions	30 questions	13 questions	71%	63%	65%

Table 5-4 Evaluation of answer selection after question classification using HMM



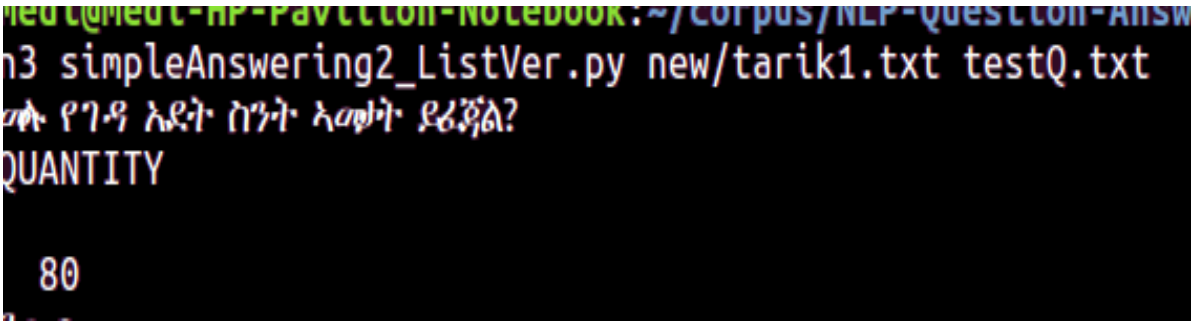


Figure 5-1 screen shot of correct answers

The above figures show how the system returns answers to the asked questions. Both figures are samples of the correctly answered questions. The system accepts a file containing the question then prints the question type and then prints the answer.

Answer extraction component could be affected by the type of the question for example the question "የ ውጫሌ ውል የ ተፈረመው በ ማን እና በ ማን መካከል ነ በር?" is asking for the countries that signed the Wuchale treaty but the machine understands that the question is of a PERSON type. It could also be affected by the retrieval component. If the retrieval component returns the wrong sentence, the answer that will be returned is also the wrong one.

5.7 Findings

The overall performance of the system is affected at every stage or components. The accuracy of the POS tagger is 76%. The rest (24%) of the corpus was not correctly tagged because the training corpus was small and because of the quality of the corpus. This reasons resulted the performance of the tagger to be low. Then this performance influences the NER in finding names and tagging them appropriately, as the wrongly tagged parts of the corpus has some effect on the NER. In addition, question classification is influenced as it uses POS and NER to determine the type of the questions.

The performance of the NER is the result of the POS tagged training data, which affects the answer selection and extraction component in such a way that if a place is tagged as an organization the answer extraction might return a wrong, or no answer. In addition, some names are not written correctly and the NER could not find and tag them. The NER also has an effect on question classification as SVM uses both POS and NER for classification. The performance of question classification affects the question answering by not returning the right type of

question. If a question word is of person type and if it is classified as location, the question answering cannot return any answer or will return a wrong answer. The performance of passage retrieval affects the question answering by not ranking the relevant sentences at the top. This could happen when the retrieval component finds the wrong sentences to be the top ranked. If the top ranked sentence does not contain the candidate answer, the performance of answer selection would also be influenced. As a whole the final performance or accuracy of the system is a result of all of the components mentioned above.

Chapter 6

6 Conclusion and Recommendation

6.1 Conclusion

In this study, we attempted to design a multi-type QA for Amharic by integrating factoid and list questions. Both types follow the same architecture but have different answer extraction techniques. After identifying the question types, we have extracted answers using co-occurrence of answer types with answers in the corpus to return answer for list questions and a key word matching technique for answering factoid questions. Using two algorithms for classification, we have identified types of input questions. Doing this, SVM classifier returned a better result in classification and answer extraction than an HMM classifier. The development of POS tagger and NER also helps the QA in many ways as NER assists in getting the exact answer for the proposed question. Therefore, we have realized that using NER helps gain a better performance from previous works which attained an F-score of 72% [6] and 65% [8] and get exact answers easily.

The performance of the QA system is affected by the different components as mentioned in the experiment section. In addition, the type, quality and size of corpus used in the process influences the performance of the tools mentioned above. From the experiment we did, we realized that if the size of the corpus is increased the performance of POS tagger and NER will also increase. Then the performance of answer extraction also increases. Finally, we conclude that having a machine learning based question classification and using the appropriate NLP tools, we can get a better performing QA for Amharic.

6.2 Recommendation

After the results shown from the experiments the researcher believes that there are some improvements that can be done for future works.

- Having a quality corpus is a very important thing in designing NLP systems. Whenever the size of the corpus increases the performance of the POS tagger and NER will also increase. Therefore, researchers should prepare a corpus larger in size and in the domain of their research with a good quality.
- Amharic WordNet should be applied as it contains word synonym hyponym and antonym and this can increase the recall of a question answering. In addition, it assists in retrieval of sentences containing candidate answers. If we have a WordNet a query expansion technique can also be applied.
- Question classification and answer type identification can be performed using other machine learning algorithms and deep learning.
- Researches done on closed domain question answering should also be applied on open domain QA system and researchers should come up with an extended research work. More over researchers should also integrate factoid and the other non-factoid question and come up with a full-fledged QA.

7 References

[1] P. Aggarwal and G. Sharma, "Questioning Answering Mechanism Using Natural Language Processing," vol. 6, no. 7, pp. 22020–22026, 2017.

[2] S. K. Dwivedi and V. Singh, "Research and Reviews in Question Answering System," *Procedia Technol.*, vol. 10, pp. 417–424, 2013.

[3] D. Jurafsky and J. H. Martin, "Question Answering," *Speech Lang. Process.*, 2017.

[4] A. Purwarianti, M. Tsuchiya, and S. Nakagawa, "A machine learning approach for Factoid Question Answering," *Proc. IASTED AIA*, pp. 131–136, 2007.

[5] L.HIRSCHMAN and R. GAIZAUSKAS, "Natural language question answering : the view from here," vol. 7, no. 4, pp. 275–300, 2001.

[6] Seid Muhie, "AMHARIC QUESTION ANSWERING SYSTEM FOR FACTOID QUESTIONS," Addis Ababa University, 2009.

[7] Desalegn Abebaw, "A Web Based Amharic Question Answering System for Factoid Questions Using Machine Learning Approach," ADDIS ABABA UNIVERSITY, 2013.

[8] Brook Eshetu, "Amharic Question Answering for list questions," Addis Ababa University, 2013.

[9] Wondwossen Teshome, "Designing Amharic Definitive Question Answering," Addis Ababa University, 2013.

[10] B. Kuechler and V. Vaishnavi, "Theory development in design science research: anatomy of a research project," *European Journal of Information Systems*, vol. 17, no. 5. pp. 489–504, 2008.

[11] P. Gupta, "A Survey of Text Question Answering Techniques," vol. 53, no. 4, pp. 1–8, 2012.

[12] R. P. Kamdi and A. J. Agrawal, "Keywords based Closed Domain Question Answering System for Indian Penal Code Sections and Indian Amendment Laws," *I.J. Intell. Syst. Appl. Intell. Syst. Appl.*, vol. 12, no. 12, pp. 57–67, 2015.

[13] I. Maleki, S. R. Khaze, S. Amir, R. Abedini, and D. Branch, "APPLICATION OF HIDDEN MARKOV MODEL IN QUESTION ANSWERING SYSTEMS," vol. 3, no. 5, pp. 17–27, 2013.

[14] F. Benamara, "Question Answering Systems: State of the Art and Future Directions," *LAICS-NLP Summer Sch. Lang. Artif. Intell. Comput. Sci. Nat. Lang. Process.*, no.

October, 2006.

[15] J. L. Vicedo and A. Ferrández, “A semantic approach to Question Answering systems,” *Nist Spec. Publ. Sp*, pp. 511–516, 2001.

[16] O. Kolomiyets and M. F. Moens, “A survey on question answering technology from an information retrieval perspective,” *Inf. Sci. (Ny).*, vol. 181, no. 24, pp. 5412–5434, 2011.

[17] J. Fukumoto, “Question Answering System for Non-factoid Type Questions and Automatic Evaluation based on BE Method,” *Proc. NTCIR-6 Work. Meet.*, pp. 441–447, 2007.

[18] D. Melo, I. P. Rodrigues, and V. B. Nogueira, “A Review on Cooperative Question-Answering Systems,” *Jornadas Informática da Univ. Évora*, vol. 3, pp. 23–30, 2013.

[19] A. Mone, I. Mete, P. Gangarde, and M. Kharad, “Automatic Answering System for English Language Questions,” vol. 5, no. 2, pp. 2077–2079, 2014.

[20] H. T. Ng, J. Lai, and P. Kwan, “A Machine Learning Approach to Answering Questions for : Reading Comprehension Tests,” *Proc. EMNLP/VLC*, pp. 124–132, 2000.

[21] D. Sonntag, “Distributed NLP and Machine Learning for Question Answering Grid.”

[22] D. Zhang and W. S. Lee, “Question classification using support vector machines,” *SIGIR ’03 Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 26–32, 2003.

[23] C. Giannone, V. Bellomaria, and R. Basili, “A HMM-based Approach to Question Answering against Linked Data.”

[24] N. P. Er and I. Cicekli, “A Factoid Question Answering System,” no. October, pp. 854–858, 2013.

[25] M. Razmara and L. Kosseim, “Answering List Questions using Co-occurrence and Clustering,” pp. 3218–3222, 2007.

[26] L. Gillard, P. Bellot, and M. El-Bèze, “Question Answering Evaluation Survey,” vol. 9, pp. 1133–1138, 2004.

[27] Tilahun Abdisa, “Amharic Question Answering For Definitional, Biographical and Description Questions,” ADDIS ABABA UNIVERISTY, 2013.

[28] B. Hammo and S. Lytinen, “QARAB : A Question Answering System to Support the Arabic Language.”

[29] L. Abouenour, K. Bouzoubaa, and P. Rosso, “IDRAAQ : New Arabic Question Answering system based on Query Expansion and Passage Retrieval.”

[30] S. Russell, “Amharic,” 2009.

[31] A. Wimsatt and R. Wynn, “Amharic Language and Culture Manual,” 2011.

[32] C. W. Isenburg, *Grammar of The Amharic Language*. London, 1842.

[33] S. Y. R. Esearch, B. A. R. Hevner, S. T. March, J. Park, and S. Ram, “D Esign S Cience in I Nformation,” *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004.

[34] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A Design Science Research Methodology for Information Systems Research,” *J. Manag. Inf. Syst.*, vol. 24, no. 3, pp. 45–77, 2007.

[35] Seid Muhe Yimam, “A Brief Analysis of Amharic NLP : From POS Tagging to Question Answering My profile,” no. September, 2016.

[36] Binyam Gebre and L. Processing, “Part of speech tagging for amharic,” pp. 1–20.

[37] D. Mollá, M. Van Zaanen, and D. Smith, “Named entity recognition for question answering,” *Proc. ALTW*, pp. 51–58, 2006.

[38] B. C. Das, “A Survey on Question Answering System,” Indian Institute of Technology, Bombay.

[39] D. Chopra and S. Morwal, “N AMED E NTITY R ECOGNITION I N E NGLISH U SING HIDDEN MARKOV MODEL,” no. 1, pp. 1–6, 2013.

[40] J. Suzuki, Y. Sasaki, and E. Maeda, “SVM answer selection for open-domain question answering,” *Proc. 19th Int. Conf. Comput. Linguist.*, pp. 1–7, 2002.

8 Appendices

A. Questions Prepared for Testing

1. ብጹእ አቡነ ጴጥሮስ የቀረበባቸውንጀል ምን ምን ነበር?
2. በላስታና ዋግ ከድንጋይ ፍልፍል አቢያተ ክርስትያናት የት ናቸዉ?
3. ሙሉ የገዳ ኡደት ስንት ኣመታት ይፈጃል?
4. የ ሜጨሻውየ ካፋ ንጉስ ማን ናቸዉ?
5. የ አባቶች ገዳ መደቦች ጥቀስ?
6. የ አገ ውህዝቦች አካላት ተብለውየ ሚጠቀሱት ዘርዝር?
7. ቅዱስ ላሊበላ ዙፋን ላይ መቼ ወጣ?
8. የ ላሊበላ የ ዘወድ ስምማን ይባላል?
9. የ ዛጉዌ ስርወ መንግስት የ መጀመሪያውንጉስ ማን ነ ዉ?
10. በጎንደር የ ሚኖሩ የ አገ ውህዝቦች ማን ይባላሉ?
11. በገዳ ስርኣት ወስጥ በአባትና በልጅ መካከል ስንት ኣመት ልዩነ ት ይኖራል?
12. ቅዱስ ላሊበላ መቼ ተወለደ?
13. አጼ ዮሀንስ በማን እጅ ተቀብተውነገሱ?
14. በአዲስ አበባ ለመጀመሪያ ጊዜ አውቶሞቢል የገባችው መቼ ነ ዉ?
15. በኢትዮጵያ ለመጀመሪያ ጊዜ ባንክ የ ተቋቋመው መቼ ነ ዉ?
16. ሆስፒታል ስራ የ ተጀመረውመቼ ነ ዉ?
17. ኢትዮጵያ የ ኣለም ፖስታ ድርጅት አባል የ ሆነ ችው መቼ ነ ዉ?
18. የ ስልክ መሳ ጋገሪያ መሳሪያ ወደ ኢትዮጲያ ያስገባውማን ነ ዉ?
19. የ መጀመሪያው ባቡር አዲስ አበባ መቼ ገባ?
20. በኢትዮጵያ ወስጥ ዘመናዊ የ ትምህርት ዘር የ ዘሩ ንጉስ ማን ናቸዉ?
21. ጣልያንን በዶጋሊ ጦርነ ት ያሸነ ፈውአርበኛ ማን ነ ዉ?
22. የ አድዋ ጦርነ ት መቼ ተካሄደ?
23. የ ወጫሌ ወል የ ተፈረመውበማን እና በማን ነ በር?
24. ከአድዋ እስከ ሚይጨውየ ተደረጉት አበይት ጦርነ ቶች ምን ምን ናቸዉ?
25. ከአድዋ እስከ ሚይጨውየ ተደረጉት አበይት ጦርነ ቶች ስንት ናቸዉ?
26. በኢጣልያና በኢትዮጵያ ጦር መካከል የ መጀመሪያውዉጊያ የ ት ነ በር?
27. በአድዋ ዘመቻ የ ኢትዮጲያ የ ጦር አበጋዞች ማን ማን ነ በሩ?
28. ንጉስ ሚክኤል የ የ ት ባላባት ነ በሩ?
29. አባ ጀፋር የ የ ት ባላባት ናቸዉ?
30. የ ዳግማዊ አጼ ምኒልክ ልጆች ዘርዝር?
31. ዳግማዊ አጼ ምኒልክ የ ተቀበሩት የ ት ነ ዉ?
32. አጼ በካፋ የ ነገሱት ከመቼ እስከ መቼ ነ በር?
33. በአጼ ፋሲል ከተመሳረቱት አብያተ ክርስቲያናት ታዋቂወቹ ዘርዝር?
34. ጎንደር የ ኢትዮጲያ ዋና ከተማየ ሆነ ችውመቼ ነ በር?
35. አጼ ፋሲል መቼ ነገሱ?
36. አጼ ፋሲል የ ተቀበሩት የ ት ነ ዉ?
37. አጼ በካፋ የ ተቀበረውየ ት ነ ዉ?

38. የ አጼ በካፋ የ ማን ልጅ ነ ው?
39. የ አጼ በካፋ ሚስት ማን ትባላለች?
40. የ ጎንደር ከተማን የ ሚቋረሻ ግንቦች ያሰሩት ማን ማን ናቸው?
41. አጼ በካፋ የ ማን ወንድም ነ በር?
42. በካፋ ህጻንነቱን ያሳለፈውየ ት ነ በር?
43. የ ንግስት ምንትዋብ ስም ማን ነ በር?
44. የ አጼ እያሱ ተከታይ ማን ነ ው?
45. የ ዳግማዊ አጼ ኢያሱ እናት ማን ትባላለች?
46. ዳግማዊ አጼ ኢያሱ ስልጣን ላይ የ ወጣውበስንት አመቱ ነ ው?
47. የ ዳግማዊ አጼ ኢያሱ የ ዙፋን ስም ማን ይባላል?
48. አጼ እያሱ የ ነገሰውበምን ስርወ-መንግስት ነ ው?
49. የ ዘመነ መሳፍንት ማበቂያ የ ሆነ ጦርነት ምን ነ በረ?
50. አጼ ቴዎድሮስ የ ሞቱት የ ት ነ ው?
51. አጼ ቴዎድሮስ ራስ አሊን የ ት አሸነፈው?
52. ቴዎድሮስ ንጉስ ተብሎበማን ተቀባ?
53. የ አጼ ቴዎድሮስ ልጆች ማን ማን ናቸው?
54. የ አጼ ቴዎድሮስ ስርአተ ንግስ የ ተከናወነ ውየ ት ቤተክርስቲያን ነ ው?
55. በ 1848 አ.ምየ ነ በረውየ ቴዎድሮስ ግዛት የ ት የ ት ያጠቃልል ነ በር?
56. የ ደጅ አዝማች ወቤ ዋና ከተማየ ት ነ በረ?
57. የ ዘመነ መሳፍንት የ ሚቋረሻ ባላባት ማን ነ ው?
58. የ አጼ ቴዎድሮስ ወታደራዊ ስማቸውማን ይባላል?
59. የ አጼ ቴዎድሮስ የ ፈረስ ስማቸውማን ይባላል?
60. ራስ ተፈሪ መኮንን ዘወድየ ጫት ምን ቤተክርስትያን ነ ው?
61. መኳንንትና ህዝብ መከሮና ዘከሮ በምርጫበኢትዮጵያ ዙፋን ላይ የ ተቀማጡበቸኛዋ ንግስት ማን ናቸው?
62. ኢትዮጵያ የ ኣለምመንግስታት ማህበር አባል የ ሆነ ችውመቼ ነ ው?
63. ከእንግሊዝ አገር የ መጣውየ አጼ ቴዎድሮስ ዘወድ መቼ ተሞሊሰ?
64. በሰለሞን ስርወ-መንግስት ወስጥ የ ነበሩ ነገስታት ጥቀስ?
65. ለመጀመሪያ ጊዜ በምኒልክ መልክ ታትመውየ ወጡትን ገንዘቦች ዘርዝር?
66. ጀነ ራል ግራዚያኒ ላይ ቦምብ የ ጣሉት እነ ማን ናቸው?
67. የ መጀመሪያዋ የ ኢትዮጵያ የ ጦር ሚከብ ምን ትባላለች?
68. ቀሪን ገረመውየ ሚዉመጽሃፍ የ ማን ነ ው?
69. ራስ አበበ የ ተወለዱት የ ት ነ ው?
70. ቅዱስ ያሬድ የ ት ተወለደ?
71. ብራና ከምን ከምን ሊዘጋጅ ይችላል?
72. ኢጣሊያ ኢትዮጵያን በድጋሚየ ወረረችውመቼ ነ ው?
73. በአምስቱ አመት የ ኢትዮጵያ የ ነጻነት ትግል ወስጥ የ ሚከራ ታላቅ አርበኛ ማን ነ ው?
74. አጼ ካሌብ የ ማን ልጅ ነ ው?
75. ከኢትዮጵያ ወጪወጪያ ያካሄደ የ መጀመሪያውንጉስ ማን ነ ው?
76. ብጹእ አቡነ ጴጥሮስ የ ቀረበባቸውወንጀል ምን ምን ነ በር?

77. አቡነ ጴጥሮስ ንጉሰነገስቱን የዘመቱት የትነው?
78. እቴጌ ማን ከቀዳማዊ ሃይለስላሴ ያፈሯቸውን ልጆች ጥቀስ?
79. አብዲሳ አጋ የኢትዮጵያን ጦር የተቀላቀለውበስንት አመቱ ነው?
80. ዋና ዋናዎቹ የብራና ጽሄፎች በኢትዮጵያ ወስጥ የሚኙት የት ቤተክርስትያን ነው?
81. እቴጌ ማን የት ቤተክርስትያንን አሰሩ ?
82. እቴጌ ማን ዘውድ የጫት መቼ ነው?
83. አጼ ሃይለስላሴ ከስልጣን የወረዱት መቼ ነው?
84. አዲስ አበባ መቼ ተቆረቆረች?
85. ንግስተ ሳባ መቼ ነገሰች?
86. ወታደራዊ አስተዳደር ደርግ የተማከረተውመቼ ነው?
87. የኢትዮጵያ አየር መንገድ ድርጅት መቼ ተመሰረተ?
88. ዶክተር ነጋሶ ጊዳዳ ኢትዮጵያን በፕሬዘዳንትነት የሞት መቼ ነው?
89. የመጀመርያውየ ህዝብ ሲኒማቤት በአዲስ አበባ የተከፈተውመቼ ነው?
90. ሼህ ሆጀሌ የየት አካባቢ ገዥነበሩ?
91. ልእልት ዘነበወርቅ ለቀዳማዊ ሃይለስላሴ ስንተኛ ልጅ ናቸው?
92. የአጺ ፋሲል ግቢ ንጉሰ ነገስቱ የሚቡበት በር ምን ይባላል?
93. ብጹእ ወቅዱስ አቡነ ቴዎፍሎስ ስንተኛውፓትርያርክ ናቸው?
94. ከንጉስ ሰለሞን ለንግስት ሜክዳ የተበረከተላቸውቦታዎች የት ናቸው?
95. ንግስተ ሳባ ኢትዮጵያን ለስንት አመት ስታስተዳድር ቆየች?
96. ሰባራ ድልድይ በማን ጊዜ ነውየ ተሰራው?
97. በዳግማዊ ምኒልክ የተሾመት ሚስትሮች ማን ማን ናቸው?
98. በአዲስ አበባ ከተማየ መጀመርያውየ መድሃኒት መቸጫመቼ ተከፈተ?
99. አባ መላ በሜል ስምየ ሚጠሩት የጦር መሪ ማን ናቸው?
100. አጼ ገላውዴዎስ የማን ልጅ ናቸው?
101. ቀዳማዊ ሃይለ ስላሴ ስንተኛ ንጉስ ናቸው?
102. ግራኝ አህመድ የተገደለውየት ነው?
103. አጼ ገላውዲወስን የገደለውማን ነው?
104. ግራኝ አህመድ ተብሎየ ሚታወቀውየ ጦር መሪ ማሉ ስሙማን ነው?
105. ባቲ ድል ወምባሬ የማን ሚስት ነበረች?
106. ግራኝ አህመድ የልብነ ድንግልን ሰራዊት ያሸነፈውየት ነው?
107. ግራኝ አህመድ የተገደለውበምን ጦርነት ነው?
108. አጼ ዘርአ ያእቆብ ስንተኛ ንጉስ ነበሩ?
109. የአጼ ዘርአ ያእቆብ የዘውድ ስምማን ይባላል?
110. አጼ ዘርአ-ያእቆብ የተወለዱት የት ነው?
111. የአጼ ዘርአ ያእቆብ እናታቸውማን ናቸው?
112. የአጼ ዘርአ ያእቆብ ተከታይ ማን ነው?
113. አጼ ዘርአ-ያእቆብ ያረፉት የት ነው?
114. አጼ ዘርአ ያእቆብ በግዞት የኖሩት የት ቦታ ነው?
115. አጼ ዘርአ ያእቆብ የደረሳቸውመጻህፍት ጥቀስ?

116. ደብረ ብርሃን የኢትዮጵያ ዋና ከተማየነበረችውበማን ጊዜ ነው?
117. አጼ ዘርአ ያእቆብ ማን አገቡ?
118. መላኣ ጽዮን የተሰኘውየ ጥንቱ የኢትዮጵያ ኦርቶዶክስ ተዋህዶ ጳጳቅ የኖረውበማን ጊዜ ነው?

B. Sample Sentences Retrieved by the Sentence Retrieval

1. መላእ ጽዮን የተሰኘውየ ጥንቱ የኢትዮጵያ አርቶዶክስ ተዋህዶ ጳጲቅ የኖረውበኒሁ ንጉስ ዘመነ በር ::
2. አጼ ዘርአ ያእቆብ ንጉስ ከሆኑ በኋላ ንግስት እሌኒን በ1434 አገቡ፣ ከዚያም በ1436 ዘወዳቸወን ጫኑ ::
3. እስከ እለት ህልፈታቸወም ደብረ ብርሃን የኢትዮጵያ ዋና ከተማነበረች
4. አጼ ዘርአ ያእቆብ በዘወድስ ሞቹውቆስጠንጥንዮስ ከአባታቸውቀዳሚዊ ዳዊት እና ከእናታቸውንግስት ክብረ እግዚ በ1399 እ.ኤ.አ ከአዋሽ ወንዝ አጠገብ ትገኝ በነበረውትልቅ ተብላ በምትጠራውመንደር የድሮውፈተገር ክፍለ ሀገር ተወለዱ::
5. አጼ ዘርአ ያእቆብ በዘወድስ ሞቹውቆስጠንጥንዮስ ከአባታቸውቀዳሚዊ ዳዊት እና ከእናታቸውንግስት ክብረ እግዚ በ1399 እ.ኤ.አ ከአዋሽ ወንዝ አጠገብ ትገኝ በነበረውትልቅ ተብላ በምትጠራውመንደር የድሮውፈተገር ክፍለ ሀገር ተወለዱያረፉትምበደጋ ደሴት፣ ጣና ሃይቅ ነው::
6. አጼ ዘርአ ያእቆብ በዘወድስ ሞቹውቆስጠንጥንዮስ ከአባታቸውቀዳሚዊ ዳዊት እና ከእናታቸውንግስት ክብረ እግዚ በ1399 እ.ኤ.አ ከአዋሽ ወንዝ አጠገብ ትገኝ በነበረውትልቅ ተብላ በምትጠራውመንደር የድሮውፈተገር ክፍለ ሀገር ተወለዱያረፉትምበደጋ ደሴት፣ ጣና ሃይቅ ነው::
7. አጼ ዘርአ ያእቆብ በዘወድስ ሞቹውቆስጠንጥንዮስ ከአባታቸውቀዳሚዊ ዳዊት እና ከእናታቸውንግስት ክብረ እግዚ በ1399 እ.ኤ.አ ከአዋሽ ወንዝ አጠገብ ትገኝ በነበረውትልቅ ተብላ በምትጠራውመንደር የድሮውፈተገር ክፍለ ሀገር ተወለዱያረፉትምበደጋ ደሴት፣ ጣና ሃይቅ ነው::
8. አጼ ዘርአ ያእቆብ ከኢትዮጵያ ነገስታት 265ተኛ ነበሩ ::
9. በዚህ ጊዜ የአጼ ልብነ ድንግልን ሰራዊት መምጣት የሰሙት የግራኝ ወታደሮች በፍርሃት ተወጠውለሻሽ ሲሞክሩ፣ ግራኝ ግን በጽናትና ቆራጥነት አብዛኞቹ ባሉበት እንዲቆመብሚድረግ የልብነ ድንግልን ሰራዊት ከአዲስ አበባ 80 ኪሎሜትር ደቡብ ምስራቅ በምትገኘውበሽምበራ ቁሬ ጦርነት መጋቢት 1529 አ.ምአሸነፈ ::
10. በዚህ ጊዜ የአጼ ልብነ ድንግልን ሰራዊት መምጣት የሰሙት የግራኝ ወታደሮች በፍርሃት ተወጠውለሻሽ ሲሞክሩ፣ ግራኝ ግን በጽናትና ቆራጥነት አብዛኞቹ ባሉበት እንዲቆመብሚድረግ የልብነ ድንግልን ሰራዊት ከአዲስ አበባ 80 ኪሎሜትር ደቡብ ምስራቅ በምትገኘውበሽምበራ ቁሬ ጦርነት መጋቢት 1529 አ.ምአሸነፈ ::
11. በዚህ ጊዜ የአጼ ልብነ ድንግልን ሰራዊት መምጣት የሰሙት የግራኝ ወታደሮች በፍርሃት ተወጠውለሻሽ ሲሞክሩ፣ ግራኝ ግን በጽናትና ቆራጥነት አብዛኞቹ ባሉበት እንዲቆመብሚድረግ የልብነ ድንግልን ሰራዊት ከአዲስ አበባ 80 ኪሎሜትር ደቡብ ምስራቅ በምትገኘውበሽምበራ ቁሬ ጦርነት መጋቢት 1529 አ.ምአሸነፈ ::
12. በተለምዶውግራኝ አህመድ ተብሎየሚታወቀውየ ጦር መሪ መሉ ስመአህመድ ኢብን ኢብራሂምአልጋዚ ሲሆን የኖረወምከ1507 አ.ምእስከ ታህሳስ 21 1543 አ.ም(እ.ኤ.አ) ነበር ::
13. ከዚህ ጥቃት የተረፉትና የአጼ ገላውዲወስ ሰራዊት በሜጫሻ ተገናኝተውሃይላቸወን በሚጠከር የካቲት 21 ፣ 1543 የወይና ደጋ ጦርነት ተብሎበሚታወቀውጥቃት አደረሱና ግራኝ ተገደለ ::

C. A Sample NE tagged Corpus

1. በካፋ/B-PER የቀዳማዊ/B-PER እያሱ/I-PER ልጅ/NN ሲሆን/V ከሱ/PRO ቀድሞ/NN የነገሱት/V ቀዳማዊ/B-PER ተክለሃይማኖትና/I-PER ሳልሳዊ/B-PER ዳዊት/I-PER ወንድም/NN ነበር/V ::/PUNC
2. በካፋ/B-PER ህጻንነቱን/NN ያሳለፈው/V ወህኒ/B-LOC አምባ/I-LOC ላይ/PREP ነበር/V ::/PUNC
3. ሆኖምን/CONJ በንጉስ/B-PER ዮስጦስ/I-PER ሜጨሻ/NN ዘመን/NN አካባቢ/NN በተነሳ/V ግርግር/NN ሳቢያ/NN ከወህኒ/B-LOC አምልጦ/V ከኦሮሞ/NN ቡድኖች/NN ስር/PREP ተደበቀ/V ::/PUNC
4. ሳይቆይም/V በመሚኩ/V ለወደፊት/ADV እንዳይነገስ/V አፍንጫው/NN ላይ/PREP ጠባሳ/NN ተደረገበት/V ::/PUNC
5. ነገርግን/CONJ በ1721/B-DATE ወንድማNN ዳዊት/B-PER ሳልሳዊ/I-PER በሞዝ/NN ስለተገደለ/V ከነበሩት/V ተወዳዳሪወች/NN እሱ/PRO ተሞጦ/V ነገሰ/V ::/PUNC
6. የአገሪቱን/NN ሃይል/NN መዳከም/V ተከተሎ/V በመጣ/V ድንጋጤNN ምክንያት/NN በበካፋ/B-PER ዘመን/NN የነበረችው/V ኢትዮጵያ/B-LOC በሴራና/NN ተንኮል/NN የተተበተበች/V እንደነበር/V ጀምስ/B-PER ብሩስ/I-PER ያትታል/V ::/PUNC
7. ለዚህ/PRO ይመስላል/V በካፋ/B-PER ዝምተኛ/V ፣/PUNC
8. ምስጢረኛ/ADJ ፣/PUNC ልቡ/NN የሚይዝና/V በራሱ/PRO ባሪያ/NN ወታደሮችና/NN በራሱ/PRO ምስል/NN በሰራቸው/V ሰወች/NN የተከበበ/V እንደነበር/V ሃኪም/B-PER ይጋቤ/I-PER መዘግቧል/V ::/PUNC
9. በንጉሱ/B-PER ዜና-መዋል/B-ENTY ላይ/PREP የሰፈረው/V ጽሁፍ/NN የንጉሱን/B-PER ቆራጥነትና/NN መልካም/NN አስተዳደር/NN ቢያሳይም/V አላማው/NN ግን/CONJ የተሰወረ/V እንደነበር/V ብሩስ/B-PER ታዝቧል/V ::/PUNC
10. በንጉስ/B-PER በካፋ/I-PER ዘመን/NN ብዙ/ADJ ጦርነት/NN እንዳልነበረና/V አብዛኛው/ADJ የንጉሱ/B-PER ክንድ/NN ያረፈው/V የተንሰራፋውን/V የባላባቱን/NN ሃይል/NN ሰብሮ/V የመከለኛው/NN መንግስትን/NN ሃይል/NN በማጠናከር/V እንደነበር/V ታሪክ/NN አጥኝው/NN ዶናልድ/B-PER ሌቪን/I-PER ይናገራል/V ::/PUNC
11. ታሪክ/NN አጥኝው/NN ፓል/B-PER ሄንዝ/I-PER በበኩሉ/PRO ለኢትዮጵያ/B-LOC ታሪክ/NN በካፋ/B-PER ያበርከተው/V ታላቁ/ADJ አስተዋጾ/NN ሁለተኛው/NUM ሚስቱ/NN ንግስት/B-PER ምንትዋብ/I-PER እንደነበረች/V ይዘግባል/V ::/PUNC
12. የጎንደር/B-LOC ከተማን/I-LOC የሜጨሻ/NN ግንቦች/NN ያሰሩት/V በካፋና/B-PER ሚስቱ/NN ምንትዋብ/B-PER ነበሩ/V ::/PUNC
13. የበካፋ/B-PER የመጀመሪያው/NN ሚስቱ/NN የተጋበዘችው/V እዚህ/PRO ሲሆን/V ህንጻውን/NN ያስገነባው/V እራሱ/PRO በካፋ/B-PER ነው/V ንጉሱ/B-PER በካፋ/I-PER የመጀመሪያ/NN ሚስቱን/NN ካገባ/V በኋላ/ADV የዙፋን/NN ተከሊል/NN ደፋላት/V ::/PUNC
14. ቀጥሎም/ADV ለዚህ/PRO ስርአት/NN፣ ክብር/NN ሲባል/NN በአዳራሽ/NN ወስጥ/PREP ግብዣ/NN አድርጎ/V ሲያበቃ/V ሚስቲቱ/NN ከቀረበው/V ምግብ/NN እንደበላች/V ታመመችና/V ሚታውኑ/ADV አረፈች/V ::/PUNC

15. ከበካፋ/B-PER በፊት/ADV የነበረው/V ንጉስ/B-PER በሚገዝ/NN ስለሞተና/V ይቺም/PRO ሚስቱ/NN በዚያው/PRO እንደሞተች/V ከፍተኛ/ADV ወሬ/NN ስለተናፈሰ/V ከፍተኛ/ADV ሽብር/NN ከተሜቱን/NN አናወጣት/V ::/PUNC
16. በካፋ/B-PER በህዝብ/NN ዘንድ/PREP ክብርን/NN ካስገኘለት/V ስራው/NN አንዱ/NUM ብዙ/ADJ ጊዜውን/NN በማዋትና/V እራሱን/PRO ደብቆ/V በግዛቱ/NN ሁሉ/PRO እየተዘዋወረ/V ስህተት/NN የተሰራውን/V ማቃናቱ/V ነበር/V ::/PUNC
17. በዚህ/PRO ሁኔታ/NN አንድ/NUM ቀን/NN ተደብቆ/V ከጣና/B-LOC ሃይቅ/I-LOC በስተምእራብ/ADV ሲጓዝ/V ቋራ/B-LOC ላይ/PREP ታመመና/V ከአንድ/NN ገበሬ/NN ቤት/NN አረፈ/V ::/PUNC
18. የዚያ/PRO ቤት/NN ባለቤት/NN የብርሃነ/B-PER ሞገሴ/I-PER አባት/NN ሲሆኑ/V እሷም/PRO ታማሚውን/NN በካፋ/B-PER ተንከባከባ/V ለጤንነት/NN ስላበቃችው/V ወዲያው/ADV አገባት/V ::/PUNC
19. ከላይ/PRO እንደተጠቀሰው/V ከመጀመሪያይቱ/NN ሚስት/NN ሞት/NN በኋላ/ADV ከፍተኛ/ADV ፍርሃትና/NN ሽብር/NN ይታይ/V ነበር/V ::/PUNC
20. ይህ/PRO በዚህ/PRO እንዳለ/V ነበር/V ብርሃነ/B-PER ሞገሴ/I-PER ስሟን/NN ቀይራ/V በንግስና/NN ስሟNN ምንትዋብ/B-PER ተብላ/V ፋሲል/B-ENTY ግምባ/I-ENTY የገባችው/V ::/PUNC
21. በፍርሃትና/ADV ጥርጣሬ/NN በተሞላ/V ከተማ/NN ወስጥ/PREP የነበረውን/V አደገኛ/ADV ሴራ/NN ሁሉ/PRO ተቋቁማ/V ነበር/V ሃይሏን/NN አጠናክራ/V በኋላም/ADV ለከተሜቱ/V ከፍተኛ/ADV አስተዋጾን/NN ያደረገች/V ::/PUNC
22. እየተዘዋወረ/V መቆጣጠሩ/V አልበቃው/V ሲል/V እ.ኤ.አ/B-DATE 1727/I-DATE ላይ/PREP የህዝቡን/NN ስነ-ልቡና/NN ለሜቱን/V ፈለገ/V ::/PUNC
23. ለዚህ/PRO እንዲረዳው/V ካሰራው/V ግምባ/NN ወስጥ/PREP በሞሸሸግ/V ለብዙ/ADJ ቀናት/NN ሳይታይ/V ጠፉ/V ::/PUNC
24. በጊዜው/NN የነበሩ/V መኳንንት/NN በዚህ/PRO ጉዳይ/NN በመደናገጣቸው/V ግርግር/NN ተነሳ/V ::/PUNC
25. ስለዚህም/CONJ የከተማው/NN ከንቲባ/B-PER በቤተመንግስቱ/B-ORG ዙሪያ/NN ዘብ/NN አቆመ/V ::/PUNC
26. ይህ/PRO በዚህ/PRO እንዳለ/V ንጉሱ/B-PER ከተደበቀበት/V ወጥቶ/V ወደ/PREP ደብረ-ብርሃን/B-ORG ስላሴ/I-ORG ቤተክርስቲያን/I-ORG ፈረሱን/NN ጋለበ/V ::/PUNC
27. በሜቃጥለው/ADV ቀን/NN ከንቲባው/B-PER እና/CONJ ከሱ/PRO ጋር/PREP አብረወት/V የሰሩት/V ሰወች/NN የሞት/NN ፍርድ/NN ተፈረደባቸው/V ::/PUNC
28. የበካፋን/B-PER መጥፉት/V ተከተሎ/V ከንቲባወና/B-PER ግብረ/NN አበሮቹ/NN ብቻ/ADV ሳይሆኑ/V የከተማው/V ህዝቡ/PUNC ሁሉ/PRO በጣም/ADJ ተደስቶ/V እንደነበርና/V በኋላም/ADV በህይወት/NN መኖሩን/V ሲያውቁ/V በከተሜቱ/NN ሽብር/NN ተነስቶ/V ብዙው/NN ህዝብ/NN ከከተያሞቱ/NN እንደሸሸ/V ሃኪም/B-PER ይጋቤ/I-PER ይናገራል/V ::/PUNC
29. ይሁንና/CONJ ንጉሱ/B-PER ይቅርታና/NN ምህርተን/NN ስለፈቀደ/V ሽብሩ/NN በቶሎ/ADV ሊያቆም/V ችሏል/V ::/PUNC

30. ይህን/PRO አስሟከቶ/V፣ ንጉሱ/B-PER ሲናገር/V የጎንደርን/NN ህዝብ/NN እንደሚመድ/V ግን/CONJ ህዝቡ/NN በአጸፋው/NN እንደሚከሱት/V በምሬት/ADV ተናግሯል/V ::/PUNC
31. በበካፋ/B-PER ዜና-መዋእል/B-ENTY ላይ/PREP እንደተመዘገበ/V በ1726/B-DATE ላይ/PREP በማኩV ሁለት/NUM ግብጻውያን/V ፣/PUNC ድሚጥሮስና/B-PER ጊዮርጊስ/B-PER ፣/PUNC የተሰሩ/V አዲስ/ADJ አይነት/NN ታንኳወች/NN በጣና/B-LOC ሃይቅ/I-LOC ላይ/PREP እየተንሳፈፉ/V የብዙን/NN ህዝብ/NN ልብ/NN ሰርቀዋል/V ::/PUNC
32. በነበረው/V የሴራና/NN እርስ/NN በርስ/NN ጥርጣሬ/NN ምክንያት/NN በበካፋ/B-PER ዘመን/NN ብዙ/ADJ ጦርነት/NN አልታየም/V ::/PUNC
33. ይሁንና/CONJ በዳሞት/B-LOC ፣/PUNC በጌምድር/B-LOC እና/CONJ ላስታ/B-LOC ዘመቻወችን/NN አካሂዷል/V ::/PUNC
34. የኦሮሞ/B-ENTY ቡድኖችንም/I-ENTY ሃይል/NN መጠት/NN ስላልቻለ/V ምስራቅ/B-LOC ሸዋ/I-LOC በነኝሁ/PRO ቡድኖች/NN ቁጥጥር/V ስር/PREP ዋለ/V ፣/PUNC
35. ሲዳሞም/B-LOC በዚሁ/PRO ከተቀረው/V ክፍል/NN የተነጠለው/V በዚህ/PRO ጊዜ/NN ነው/V ፣/PUNC
36. እንራያ/B-LOC የተባለውም/V የክርስቲያን/B-ORG አገርም/NN የኦሮሞ/NN ቡድኖች/NN ስለተቆጣጠሩት/V ከዚህ/PRO በኋላ/ADV አበቃለት/V ::/PUNC
37. በዚህ/PRO ሁኔታ/NN ሸዋ/B-LOC እራሱን/PRO ችሎ/V በንጉስ/B-PER አብይ/I-PER መተዳደር/V ጀመረ/V ::/PUNC

D. A sample POS Tagged Corpus

በካፋ/NN የቀዳሚ/NN እያሱ/NN ልጅ/NN ሲሆን/CONJ ከሱ/NN ቀድሞ/V የነገሱት/V ቀዳሚ/NN ተክለሃይማኖትና/NN ሳልሳዊ/NN ዳዊት/NN ወንድም/NN ነበር/V ::/PUNC

በካፋ/NN ህጻንነቱን/NN ያሳለፈው/V ወህኒ/NN አምባ/NN ላይ/PREP ነበር/V ::/PUNC

ሆኖም/CONJ ግን/CONJ በንጉስ/NN ዮስጦስ/NN ሞጪሻ/NN ዘመን/NN አካባቢ/NN በተነሳ/V ግርግር/NN ሳቢያ/NN ከወህኒ/NN አምልጦ/V ከኦሮሞ/NN ቡድኖች/NN ስር/PREP ተደበቀ/V ::/PUNC

ሳይቆይም/V በመሚኩ/V ለወደፊት/NN እንዳይነገስ/V አፍንጫው/NN ላይ/PREP ጠባሳ/NN ተደረገበት/V ::/PUNC

ነገርግን/NN በ1721/NUM ወንድሙNN ዳዊት/NN ሳልሳዊ/NN በሞዝ/NN ስለተገደለ/V ከነ በሩት/NN ተወዳዳሪወች/NN እሱ/NN ተሞጦ/V ነገሰ/V ::/PUNC

የአገሪቱን/NN ሃይል/NN መዳከም/NN ተከተሎ/V በመጣ/V ድንጋጤNN ምክንያት/NN በበካፋ/NN ዘመን/NN የነበረችውNN ኢትዮጵያ/NN በሴራና/NN ተንኮል/NN የተተበተበች/NN እንደነበር/V ጀምስ/NN ብሩስ/NN ያትታል/V ::/PUNC

ለዚህ/PRO ይመስላል/V በካፋ/NN ዝምተኛ/ADJ ፣/PUNC ምስጢረኛ/ADJ ፣/PUNC ልቡ/NN የሚገኝና/V በራሱ/NN ባሪያ/NN ወታደሮችና/NN በራሱ/NN ምስል/NN በሰራቸው/V ሰወች/NN የተከበበ/V እንደነበር/V ሃኪም/NN ይጋቤ/NN መዘግቧል/V ::/PUNC

በንጉሱ/NN ዜና-መዋል/NN ላይ/PREP የሰፈረው/V ጽሁፍ/NN የንጉሱን/NN ቆራጥነ ትና/NN ሞልካም/NN አስተዳደር/NN ቢያሳይም/V አላማው/NN ግን/CONJ የተሰወረ/V እንደነበር/V ብሩስ/NN ታዝቧል/V ::/PUNC

በንጉስ/NN በካፋ/NN ዘመን/NN ብዙ/NUM ጦርነት/NN እንዳልነበረና/V አብዛኛው/ADJ የንጉሱ/NN ከንድ/NN ያረፈው/V የተንሰራፋውን/V የባላባቱን/NN ሃይል/NN ሰብሮ/V የመካከለኛው/NN መንግስትን/NN ሃይል/NN በማጠናከር/V እንደነበር/V ታሪክ/NN አጥኝው/NN ዶናልድ/NN ሌቪን/NN ይናገራል/V ::/PUNC

ታሪክ/NN አጥኝው/NN ፓል/NN ሄንዝ/NN በበኩሉ/PRO ለኢትዮጵያ/NN ታሪክ/NN በካፋ/NN ያበርከተው/V ታላቁ/ADJ አስተዋጾ/NN ሁለተኛው/NUM ሚኒቱ/NN ንግስት/NN ምንትዋብ/NN እንደነበረች/V ይዘግባል/V ::/PUNC

የጎንደር/NN ከተማን/NN የሞጪሻ/NUM ግንቦች/NN ያሰሩት/V በካፋና/NN ሚኒቱ/NN ምንትዋብ/NN ነበሩ/V ::/PUNC

የበካፋ/NN ምላብ/NN አዳራሽ/NN ፡/PUNC የበካፋ/NN የመጀመሪያው/NN ሚኒቱ/NN የተጋበዘችው/V እዚህ/PRO ሲሆን/PRO ህንጻውን/NN ያስገነባው/V እራሱ/NN በካፋ/NN ነው/NN

ንጉሱ/NN በካፋ/NN የመጀመሪያ/NN ሚኒቱን/NN ካገባ/V በኋላ/ADV የዙፋን/NN ተከሊል/NN ደፋላት/V ::/PUNC

ቀጥሎም/PRO ለዚህ/PRO ስርኣት/NN ከብር/NN ሲባል/V በአዳራሽ/NN ወስጥ/PREP ግብዣ/NN አድርጎ/V ሲያበቃ/V ሚኒቲቱ/NN ከቀረበው/V ምግብ/NN እንደበላች/V ታመመችና/V ሚታውኩ/ADV አረፈች/V ::/PUNC

ከበካፋ/NN በፊት/ADV የነበረው/V ንጉስ/NN በሚገዝ/NN ስለሞተና/NN ይቺም/PRO ሚኒቱ/NN በዚያው/PRO እንደሞተች/NN ከፍተኛ/ADV ወሬ/NN ስለተናፈሰ/V ከፍተኛ/ADV ሽብር/NN ከተማይቱን/NN አናወጣት/V ::/PUNC

በካፋ/NN በህዝብ/NN ዘንድ/NN ክብርን/NN ካስገኘለት/V ስራው/V አንዱ/NUM ብዙ/ADJ ጊዜውን/NN በማሳየትና/V እራሱን/NN ደብቆ/V በግዛቱ/NN ሁሉ/NN እየተዘዋወረ/V ስህተት/NN የተሰራውን/V ሚቃናቱ/V ነበር/V ::/PUNC

በዚህ/PRO ሁኔታ/NN አንድ/NUM ቀን/NN ተደብቆ/V ከጣና/NN ሃይቅ/NN በስተምእራብ/NN ሲጓዝ/V ቋራ/NN ላይ/PREP ታመመና/V ከአንድ/NUM ገበሬ/NN ቤት/NN አረፈ/V ::/PUNC

የዚያ/PRO ቤት/NN ባለቤት/NN የብርሃነ/NN ሞገሴ/NN አባት/NN ሲሆኑ/V እሷም/NN ታመምውን/NN በካፋ/NN ተንከባከባ/V ለጤንነት/NN ስላበቃችው/V ወዲያው/ADV አገባት/V ::/PUNC

ከላይ/PREP እንደተጠቀሰው/V ከመጀመሪያይቱ/NUM ሚስት/NN ሞት/NN በኋላ/ADV ከፍተኛ/ADV ፍርሃትና/NN ሽብር/NN ይታይ/V ነበር/V ::/PUNC

ይህ/PRO በዚህ/PRO እንዳለ/V ነበር/V ብርሃነ/NN ሞገሴ/NN ስሟን/NN ቀይራ/V በንግስና/NN ስሟ/NN ምንትዋብ/NN ተብላ/V ፋሲል/NN ግምባ/NN የገባችው/V ::/PUNC

በፍርሃትና/NN ጥርጣሬ/NN በተሞላ/V ከተማ/NN ወስጥ/PREP የነበረውን/NN አደገኛ/ADJ ሴራ/NN ሁሉ/NN ተቋቁማ/V ነበር/V ሃይሏን/NN አጠናክራ/V በኋላም/ADV ለከተሚቱ/NN ከፍተኛ/ADV አስተዋጾን/NN ያደረገች/V ::/PUNC

እየተዘዋወረ/V መቄጣጠሩ/V አልበቃው/V ሲል/V እ.ኤ.አ/ABBR 1727/NUM ላይ/PREP የህዝቡን/NN ስነ-ልቡና/NN ለመፈተን/V ፈለገ/V ::/PUNC

ለዚህ/PRO እንዲረዳው/V ካሰራው/V ግምባ/NN ወስጥ/PREP በመሸሸግ/V ለብዙ/NUM ቀናት/NN ሳይታይ/V ጠፉ/V ::/PUNC

በጊዜው/NN የነበሩ/V መኳንንት/NN በዚህ/PRO ጉዳይ/NN በመደናገጣቸው/V ግርግር/NN ተነሳ/V ::/PUNC

ስለዚህም/CONJ የከተማው/NN ከንቲባ/NN በቤተመንግስቱ/NN ዙሪያ/NN ዘብ/NN አቆመ/V ::/PUNC

ይህ/PRO በዚህ/PRO እንዳለ/NN ንጉሱ/NN ከተደበቀበት/V ወጥቶ/V ወደ/PREP ደብረ-ብርሃን/NN ስላሴ/NN ቤተክርስቲያን/NN ፈረሱን/NN ጋለበ/V ::/PUNC

E. A Sample of the Question Answering

No_	Test Questions	Returned Answers	Expected Answers	Remark
1	በአድዋ ዘመቻ የኢትዮጲያ የጦር አበጋዞች ማን ማን ነበሩ?	ራስ ሚኪኤል ራስ መኮንን ራስ መንገሻ ንጉስ ተክለ ሃይማኖት	ራስ ሚኪኤል ራስ መኮንን ራስ መንገሻ ንጉስ ተክለ ሃይማኖት	Correct
2	ንግስተ ሳባ ኢትዮጵያን ለስንት አመት ስታስተዳድር ቆየች?	አንድ	31	Wrong
3	የአባቶች ገዳ መጀቦች ጥቀስ?	-	ሜልባ መጀና ኪለሌ ቢፎሌ ሚቹሌ	No Answer
4	የመጀመሪያዋ የኢትዮጵያ የጦር ሚኒስትር ምን ትባላለች?	ዘራይ ደረስ	ዘራይ ደረስ	Correct
5	የአድዋ ጦርነት መቼ ተካሄደ?	1888 አ.ም	1888 አ.ም	Correct
6	የወጫሌ ወል የተፈረመውበማን እና በማን ነበር?	ምኒልክ ምኒልክ	ኢትዮጲያና ኢጣሊያ	Wrong

F. The Training Code for POS Tagger

```
from read_data_from_file import *
from math import log
import sys
class CalProbabilities:
    def __init__(self):
        self.word_tag_count = {} # dictionary to store the number of occurrences for a given word-tag comb
        self.word_count = {}         # dictionary to store the number of occurrences for a word
        self.trigram_tags_count = {}  # dictionary to store the number of occurrences of triplet or trigram of
tags
        self.bigram_tags_count = {}         # dictionary to store the number of occurrences of bigram tags
        self.emission_probabilities = {} # dictionary to store the emission probability for a given word-tag
        self.transition_probabilities = {} # dictionary to store the transition probability for a given trigram
        # a dictionary containing all the tags for every word. So key is a word and value is a set/list of tags
        self.word_tags_set = {}
        self.unique_tags = set()         # a set of all unique tags in the training corpus
    # calculate the word-tag counts and word counts and populate the respective dictionaries
    def populate_count_dicts(self, filename):
        read_files = ReadFiles(filename)
        all_tuples = read_files.word_tag_tuples()
        self.word_tags_set = read_files.word_tags
        self.unique_tags = read_files.unique_tags
        for sentence in all_tuples:
            for i in range(2, len(sentence)):
                # populate word-tag dictionary
                if sentence[i] in self.word_tag_count:
                    self.word_tag_count[sentence[i]] += 1
                else:
                    self.word_tag_count[sentence[i]] = 1
```

```
            # populate word count dictionary
            if sentence[i][0] in self.word_count:
                self.word_count[sentence[i][0]] += 1
            else:
                self.word_count[sentence[i][0]] = 1
            # populate trigram dictionary
            words_trigram = (sentence[i-2][1],sentence[i-1][1],sentence[i][1])
            if words_trigram in self.trigram_tags_count:
                self.trigram_tags_count[words_trigram] += 1
            else:
                self.trigram_tags_count[words_trigram] = 1
            # populate bigram dictionary
            words_bigram = (sentence[i - 2][1], sentence[i - 1][1])
            if words_bigram in self.bigram_tags_count:
                self.bigram_tags_count[words_bigram] += 1
            else:
                self.bigram_tags_count[words_bigram] = 1
    def _save(self):
        dictionary = {"transition": self.transition_probabilities,
                      "emission": self.emission_probabilities,
                      "word2tag": self.word_tags_set,
                      "unique_tags": self.unique_tags,
                      "bigram": self.bigram_tags_count}
        output = open("hmmmodel.txt", "wb")
        pickle.dump(dictionary, output)
        output.close()
    def run(self, filename):
        self.populate_count_dicts(filename)
        self.calculate_emission_probabilities()
```

```
        self.calculate_transition_probabilities()
        self._save()
    # calculate emission probabilities
    # p(word/tag)
    def calculate_emission_probabilities(self):
        # getting key value pair from dictionary word_tag_count
        for word_tag, word_tag_count in self.word_tag_count.items():
            # emission probability for a word-tag pair.
            # no. of occurrences of a given word-tag divided by no. of occurrences of the word.
            self.emission_probabilities[word_tag] =
log(float(word_tag_count)/float(self.word_count[word_tag[0]]))
    # calculate transition probabilities - trigram probabilities - tag given previous two tags.
    def calculate_transition_probabilities(self):
        # getting trigram tuple(key) and  count (value) pair from dictionary
        for trigram_tuple, trigram_tuple_count in self.trigram_tags_count.items():
            # getting bigram count
            bigram_count = self.bigram_tags_count[(trigram_tuple[0],trigram_tuple[1])]
            unique_tags_count = len(self.unique_tags)
            # transition probability
            # adding one to numerator and unique tag counts to denominator for smoothing
            self.transition_probabilities[trigram_tuple] = log(float(trigram_tuple_count+1)/float(bigram_count
+ unique_tags_count))
if __name__ == "__main__":
    filename = sys.argv[1]
    prob = CalProbabilities()
    prob.run(filename)
```