ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING

AUTOMATIC QUESTION CLASSIFICATION FOR SPEECH BASED AMHARIC QUESTION ANSWERING

BEKELE MENGESHA H/MESIKEL

JANUARY 2017

ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING

A Thesis Submitted to the Department of Computing of Adama Science and Technology University in Partial Fulfillment of the Requirements for Degree of Master of Science in Software Engineering

BY

BEKELE MENGESHA H/MESIKEL

January 2017

ADAMA, ETHIOPIA

ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING

AUTOMATIC QUESTION CLASSIFICATION FOR SPEECH BASED AMHARIC QUESTION ANSWERING

By

BEKELE MENGESHA HAILEMESIKEL

Advisor: Million Meshesha (PhD)

Names and Signature of Members of the Examining Board

Name		**Signature**	**Date**
_____	Chairperson,	_____	_____
Dr. Million Meshesha___	Advisor,	_____	_____
Dr Tibebe Beshah_____	Examiner,	_____	_____
_____	Examiner	_____	_____

Dedicated

To my mother and my father, tries to make my life comfortable in a different situation always. It is because of you that I am today;

ACKNOWLEDGMENT

First and foremost, I would like to thank God for supporting and being with me in all journey of my life. God, I thank you so much with your mother St. Mary.

Next, my best gratitude goes to my advisor Million Meshesha (PhD) for his, unreserved comments, encouragement, guidance and motivation he gave me to accomplish this thesis. His support has not been limited to advising this research but also giving support to my study in this program.

I sincerely thank my whole family, without their support and encouragement I would have not success today. They have always been with me supporting, helping and appreciating in my journey to be a better man.

I want to thank Dr. Sied Mohammed for his comments and editing my paper.

I would like to thank Belisty Ayalew and Fitsum Seyoum, who gave me fast response for my questions regarding their work. I really thank you for your support and encouragement.

Last but not a list I would like to thank Fikirte Tirfe for commenting on the drafts of this paper & for giving me substantial ideas and for being with me through my research.

All the respondents who participated in filling out the questionnaire also deserve gratitude. Above all, I praise my God for giving me health and all the courage to finalize this work.

Abstract

A question answering system is one of the disciplines under an information retrieval that displays precise expected answers from the huge documents for a specific question. The existing QA system doesn't support visually impaired Amharic monolingual speakers. Hence, those users cannot able to access the required information. Therefore, this study focused on developing an interactive interface using both Amharic speech recognition and synthesizer for factoid question answering with an automatic question classification. Besides, it gives an emphasis particularly on designing audio application for visually impaired Amharic monolingual speakers.

This study is conducted for designing and constructing to automatic question classification for speech-based Amharic question answering. To this end, concatenate phoneme unit selection methodology is used for speech synthesis and SVM for question classification. After all, speech recognition, question answering, and speech synthesizer are combined to construct the study. Accordingly, various tools were using in order to construct a prototype of the system. For speech recognition, sphinx4 decoding tool; for question answering Lucene, LibSVM, and for developing the whole system, java NetBean 7.1.1 were used.

We have used 22600 news articles from different newspapers (Ethiopian News Agency, Ethiopian Reporter, and from the cloud) which are prepared in 4542 files for training and testing. In this study, used 2,016 speech question sentences corpus by 24 (9 Female and 15 Male) different peoples, each having read 84 question spoken words and the questions are numeric and person related. The experimental results of speech recognition system achieved 85.58% of accuracy. Furthermore, the speech synthesis correctly pronounced 80.86% with 3.17 and 3.45 accuracy in; intelligibility and naturalness based on MOS. In addition, the SVM question classification provides 73.91% precision and 94.44% recall and 82.92%F-measure.

In general, the speech-based Amharic question answering system achieves 72.75%. The challenge of these study is, it didn't use a synonyms words to parsing a query. Therefore, as recommendation designing and developing semantic similarity using ontology based structure is needed to enhance the performance of Speech based Amharic question answering system.

Keywords: ASR, Question Answering System, Question Classification, TTS.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

SBAQA	Speech Based Amharic Question Answering
ANN:	Artificial Neural Network
ASR	Automatic Speech Recognition
CV	Consonant - Vowel
HMM	Hidden Markov Model
IR	Information Retrieval
IE	Information Extraction
MFCC	Mel Frequency Cepstral Coefficients
MOS	Mean Opinion Score
QA	Question Answering
STT	Speech to Text
TREC	Text Retrieval Conference
TTS	Text to Speech
SVM	Support Vector Machines
WSASR	Web Service Automatic Speech Recognition

CHAPTER ONE
INTRODUCTION

This chapter presents the overall concepts of the thesis and it is organized into seven sections. The first section illustrates the background of the study, and the remaining sections present statement of the problem, objective of the study, significance of the study, methodology, scope and limitation of the study.

1.1 Background of the Study

The internet can run one or more applications that are published and retrieved using WWW from all over the world. The web has a large number of databases that can store multimedia information in the form of text, image, audio, and video in different repositories. The repositories are containing a number of Amharic documents on the web. In recent years, the technology is growing very fast and the people are able to easily access information using the internet from their computers, such as a laptop, desktop, mobile, etc. [1]. The reason for easy accessibility of information on the internet is due to two mechanisms [1, 2, 3]; Information Retrieval (IR) and Information Extraction (IE). As a result, Information Retrieval and Information Extraction are becoming very important for effectively looking up and making use of this information.

The IE technique involves NLP tools for precisely indicating a correct text. IE is a deep analysis of queries (i.e., user questions) to understand the users' intention as well as deep analysis of the document to extract correct answers (sentences or passages). The users' were not satisfied with the performance of the search engines since they are only able to return ranked lists of the related document [4]. In order to return exact answers which are asked using natural language, relevant documents should be retrieved using question answering system.

In the above-mentioned issues in search engines as a solution varies scholars propose question answering (QA) system [5, 6]. QA is one of the IE techniques that is used to extract precise answers for a specific question. It is a task in NLP that will automatically deliver answers to questions posed in natural language [7]. QA system was also proposed in different local language [8, 9].

For a question given in a natural language, the question type and the expected answer type should be analyzed. There are different question types, such as acronym, counterpart, definition [9],

famous, stand for, synonym, why, where, when, who, what/which, how, yes/no and true/false. Where, who, when, which, yes/no, true/false, etc. are kinds of factoid questions. As an example, "where is Mt. Ras Dashen located?" is a factoid question type seeking location. The study focused on the factoid question answering system. Two types of QA system that are an open-domain or closed-domain [6, 8].

Open-domain QA system supports any domain questions and answers which are collected from different sources, such as; internet, reporters, newspaper, and articles. Open-domains which are questions almost about everything. In closed domain QA system, it is designed only for a specific domain such as medical, Agriculture, Law, and Aerospatiale. There is a different type of questions in QA, such as factoid (like who, when, where, how much, etc.), definition (like How, why), opinion, comparative and evaluative questions [6]. The answers are extracted from the document in the form of passages based on the question type.

A QA system includes; question processing, document processing, and answer extraction components. The **question processing module** is responsible for determining the question types, the expected answer types, question focus, and determines the proper question to be submitted to the document retrieval component. Determining the question type, i.e., about what the question is, and what be able to done with the help of SVM question classification. The SVM question classification is known with the help of the question focus. The expected answer type is directly related to the question type and the question focuses. The crucial goal of determining the expected answer type is to easily extract the correct answer by the answer extraction module. The question processing module also generates a proper question that will help in matching relevant documents that might allow the correct answer. The **document retrieval component** is responsible for retrieving relevant documents from a collection. It is comparable to IR systems where IR systems such as search engines deliver relevant documents to the user based on the question submitted. It is clear that the document retrieval component is essential as an irrelevant document result in a wrong or NO answer. The document retrieval component might incorporate paragraph/sentence retrieval depending on the needs and techniques used in the QA system. The **answer extraction module**, which is the very core component of QA systems, involves different techniques in extracting the correct answer. This module will apply different algorithms and techniques to correctly determine the exact answer. It applies maximum effort in selecting the best answer and

incorporates a number of techniques. The performance of question answering system is evaluated from different angles. The main evaluation criterion is correctness of the returned answer. The correctness of the returned answer has different variations based on different systems. Some QA systems need the answer to be exactly such as person name and place name whereas others consider sentence or paragraphs bearing the correct answer is acceptable as an answer. Some QA systems also accept ranked answers so that the system will be evaluated based on the availability of the correct answer among the top n answers while others strictly require only the top answers as acceptable. The second evaluation criterion is the efficiency of the system towards processing time and memory space. Most of the time, QA systems will take a considerable amount of processing time and negatively affect the response time.

QA is regarded as requiring more complex natural language processing (NLP) techniques than other types of information retrieval, such as document retrieval [10]. Users expect very concise answers for factoid questions. For example, a question "Who was the first person to climb Everest? Or Where is Taj Mahal located?" are factoid questions which need a brief answer; "Edmund Hillary and his Sherpa guide Tenzing Norgay were the first humans to reach Earth's highest point: the summit of Mount Everest in the Himalayas. "Taj Mahal is located in Agra, India" respectively [11].

1.2 Statement of the Problem

Nowadays, there are 1.28 million visually impaired persons living in Ethiopia [12]. And also, as to the World Bank and World Health Organization, there are estimated 15 million children, adults and elderly persons with disabilities, representing 17.6 percent of the population [13]. Those visually impaired users cannot access the internet to get information from the information repositories. In the previous year's visually impaired users extract information using a screen reader that was working only on DOS operating system, not in a graphical user interface [14]. Screen readers are expensive, and not installed on most computers. These problems collectively limit the accessibility and usability of information for the blind users.

Furthermore, the intended research covers the problem behind visually impaired users especially for those who can understand and speak the Amharic language. Currently, there is a significant number of Amharic documents on the Web. The software which allows the users to extract the

information from the web such as; screen reader presents web content visually, it converts text into 'synthesized speech' allowing the user to alternatively listen to content. However, the content is not localized, which can't be understood by users other than the one who can understand and speak the English language. Due to this problem usage of a web page can be difficult for the visually impaired users and illiterate that speak local languages, such as Amharic.

A Question answering (QA) system is a great solution to get essential information to the users' with the help of IE. Seid Muhie [8] done a research of Amharic factoid Question Answering System. This work was done using a rule-based approach to formulating natural language queries classification. Even though the traditional question answering systems developed in the local language, still, the contents are text based thus they cannot support all users such as visually impaired, aged, handicap and illiterate people.

Moreover, most of the mobile users prefer mobile devices to perform their tasks. Mobile devices are becoming the dominant mode of information access despite being troublesome to input text using small keyboards and browsing web pages on small screens [15]. The text based question answering system was time-consuming and users are not interested as compared to speech-based QA systems [15, 8]. The speech-based question answering systems have worked in different languages in the world and it solved the problems of text-based.

Various studies have been done on a voice-enabled search on different languages to address the issue particularly with disable people. The research done by Mishra and Bangalore [5] is based on "Speech-based Question-Answering system on Mobile Device" which is used to bootstrap methods to distinguish dynamic questions from static questions and he showed the benefits of tight coupling of speech recognition and retrieval components of the system. In addition to that, the research was done by Hu, Dan, and Wang [16] conducted an open domain question answering systems with a speech interface. The developed prototype is constructed with three separated modules; speech recognition, question answering (QA), and speech synthesis in which the speech interface improves the utility of QA system.

Belisty [15] come up with a prototype of "Towards Amharic Speech Based Factoid Questions Answering" that takes speech questions and provides answers in the text form. That means the system doesn't support visually impaired Amharic monolingual speakers. The speech recognition

training data was with a limited a speech corpus. Thus, it is difficult to recognize a speaker independent [17]. He had done using manual question classification. A rule based which is difficult for construing based on a rule to classification of a question. In a rule-based question classification attempts to manually that all rule must be exits otherwise, didn't working a QA systems.

This research, therefore, aims to design a prototype of Automatic question classification for speech based Amharic question answering and to reduce the challenge of visually impaired users for access information from large documents.

To this end, the following research questions are explored and answered in this study:

- How a question classification can be constructed for question answering?
- How a continuous Amharic speech recognition system can be constructed for factoid question processing?
- How a speech synthesis can be integrated for factoid question answering?
- To what extent a speech based Amharic question answering prototype is effective?

1.3 Objective of the study

The general and specific objectives of this study are discussed as follows;

1.3.1 General Objective

The main objective of this study is to design a prototype of Automatic question classification for speech based Amharic question answering that allows visually impaired users.

1.3.2 Specific Objectives

In order to achieve the general objective, the following step-by-step specific objectives are accomplished to complete the research in this study;

- To review the literature and related works so as to identify the gaps, suitable techniques, and approaches in the area.
- To analyze and identify the methods, techniques, and tools in the research components, such as speech recognition, speech synthesis, and question classification for QA system.

- To construct a model for speech recognition, QA, and speech synthesis with an Automatic question classification on Amharic speech question corpus, documented data, and phonemes recorded database.
- To integrate both Amharic speech recognition and speech synthesis within enabled Amharic question answering system.
- To develop a prototype of Automatic question classification for speech based Amharic question answering.
- To evaluate the performance and user acceptance of the proposed prototype.

1.4 Scope and Limitations of the Study

The scope of the study is to designing an Automatic question classification for speech based Amharic question answering that can retrieve and extract information using Amharic speech so as to get information from Amharic documents and produce in the form of audio. Hence, it integrates STT and TTS into question answering systems for converting speech to text for searching and text to audio for providing a search result.

The proposed systems are developed by the medium vocabulary that uses a Speaker independent continuous Amharic speech recognizer and a speech synthesizer. The collected corpus is prepared using recording techniques from different users living in Debre Berhan University staff and students for training and testing. In addition to Amharic question answering document with its corpus developed by [15], add more data in this study for training.

The major challenge of the system is due to structure matching algorithm, the questions are not clear and they don't have semantic similarity with WordNet. Wrong answer or no answer results in type's answer of the indirect question. It is better to use semantic similarity parsing analysis in terms of ontology concepts.

Furthermore, a question classification is focused only person and numeric data for the sake of data limitation. Because, it is difficult to gather corpus that means there is no standard Amharic corpus. In this study is not focused on Amharic dialects.

1.5 Methodologies of the Study

The term methodology is derived from a Greek word, denoting the practice of analyzing different methods, implying a set or system of methods, principles, and rules for regulating a given discipline [18]. As explained in [19], research methodology is a way to systematically answer the research problem. It is able to understand as a science of studying how research is done scientifically. Therefore, the following approach, methods, techniques, and tools were used with the aim of achieving the general objective of this study. The following methodologies were used in the course of this study.

1.5.1 Research Design

This study follows design science research approach, which is commonly used in design and development integration sciences, such as information technology, computer science, information sciences and software engineering etc. It is a collection of research designs which uses design and develop to understand fundamental processes.

The design science research methodology is worked by different researchers [20, 21, 22]. The DSRM presented here incorporates; principles, practices, and procedures required to carry out such study. It meets three objectives; it is consistent with prior literature; it provides a nominal process model for doing DS research; it provides a mental model for presenting and evaluating DS research in software engineering and information systems.

Therefore, in this study we focused on more related Design Science Research Methodology (DSRM) that was done by Peffers et.al [21]. Peffers et.al [21]DSRM process includes six steps:

1. Problem identification and motivation;
2. Definition of the objectives for a solution;
3. Design and development;
4. Demonstration;
5. Evaluation;
6. Communication.

The study is designing as a result of the researcher's activities. The artifact is then evaluated, giving feedback about what is good or bad about it, telling us what can be done to improve it, and gives further understanding of the problem.

1.5.2 Literature Review

Before build and evaluate the system, gathers a related literature from journal articles, books, conference papers, and the Internet have been reviewed for understanding concepts related to automatic speech recognitions, speech synthesis, question answering and Amharic language phonetics and writing systems. In addition, we reviewed also conducting to be similar to with the previous research for Amharic factoid QA system [8, 15]. Then, the best techniques and tools were analyzed, selected, adopted and adapted that modified with the intention of constructing a system.

1.5.3 Corpus Preparation

These study uses large number of Amharic documents and speech corpus for training and testing purpose. The first tasks are a voice or text corpus source data was prepared. The text corpus is collected from Seid and Belisty [8, 15] and from the internet, newspapers, fiction, books, etc. Then, these sentences are recorded while spoken by the target groups. Then the recorded data pass through segmentation and labeling steps for each sentence.

The researcher used 22600 news articles from different newspapers (Ethiopian News Agency, Ethiopian Reporter, and from the cloud) which are prepared in 4542 files. In this study, used 2,016 speech question sentences corpus by 24 (9 Female and 15 Male) different peoples, each having read 84 question spoken words and the questions are numeric and person related. The research design methods that are important for developing a speech based Amharic factoid question answering.

1.5.4 System Development Methodology

After prepared data, the prototyping approach is followed to build the speech-based Amharic question answering system. The components have to integrate and coordinated to construct the architecture of a speech based Amharic factoid question answering.

1.5.4.1 Automatic Speech Recognition

The first procedure is to design Amharic speech recognition that able to identify the incoming voice (question) from the user and convert into a written text form. Reviewed literature papers that used a speech recognition approaches (i.e. Acoustic Phonetic; Pattern Recognition Approach; Artificial Intelligence approach). Most research also used pattern recognition approach which is stochastic Hidden Markova Model. The adopted techniques have (i.e. speech signal analysis, feature extraction, speech capturing, acoustic front end, lexical model, language model and Decoder) process into the text-based question.

1.5.4.2 Question Answering

The second task is formulating an Amharic factoid question answering system. In QA system obviously, there are three fundamental processes [8]: question processing, document processing, and answer selection and extraction.

In the question analysis component responsible for determining question type (such as a person, time, place, or quantity) of the income user's requests. It is working by Automatic question classification using machine learning (statistical) approach. Support Vector Machine was using to classify the question. In this research, there are two class to classify by question classification thus are; person and numerical data.

Document retrieval is a component which receives generated queries and looks for documents containing information related to what the user asked. This searching is made in the index directory containing indexed documents. When relevant documents are found, they will be ranked and returned to the answer extraction component.

The answer extraction is a component that extracts facts which are believed to be an answer to the user's question.

1.5.4.3 Speech Synthesis

The last task in this study is a synthesizer convert resulting answers (strings character) into an audio waveform. In speech synthesis, there are several approaches and methods, such as acoustic synthesis, articulatory synthesis, formant synthesis, and concatenate synthesis (Domain synthesis,

Diaphone synthesis, and unit selection synthesis). But in this study, concatenate unit selector approach in phoneme based is used.

Finally, design and formulating of the proposed architecture of automatic question classification for speech based Amharic question answering. Besides, to design the study used to different implementation tools were used.

1.5.5 Implementation Tools

Software and tools were used to build the designing and constructing the system to accomplish the research tasks. Most of the techniques and tools are adopted and added some software tools from the previous researches'. To design a model of the study are used NetBeans 7.1.1 Programming language tools. Java is a robust and independent platform. In addition, Lucene was used for question answering.

The speech recognition was used these tools sphinx-3, sphinx-base and sphinx-Train, and sphinx4 for the purpose of training phase. Most of those tools have a clear procedure to construct STT and TTS in this system. LibSVM is used for Automatic question classification and prepared SVM model, which is written in java programming language. Wave Surfer also a speech analysis tools were used to analyze the speech and extract the formants and other relevant information from the speech file.

Based on in the above implementation tools, the researcher can be designed and constructed prototype with the three separate components which are integrated using java programming language NetBeans 7.1.1 in windows10 for the corei5 computer. Next, the implementation of the prototype is to demonstrate further system evaluation.

1.5.6 Evaluation Methods

The system evaluation procedures follows two general performance testing methods. The first method is evaluating the performance of the system and the next is user acceptance testing.

In the training part, 37 and 45, person and numerical question data types are used for question classification. Furthermore, the test dataset use of 23 question sentences with two speakers. In the

synthesis part, 10 users were used for testing a naturalness & intelligibility of the system, and user acceptance testing.

Initially, the accuracy of the system performance is measured based on the Amharic speech recognition, the Amharic question answering system and Amharic speech synthesis are tested independently. The evaluation of speech recognition in terms of the word or sentences that are accurate matches by using sphinx tools. And also, the accuracy of the question answering system model is evaluated based Precision, Recall, and F-measure. In addition to evaluating the question classification before and after the integration. In the speech synthesis part, two tested measurements were used those are correct pronounced, and based on MOS accuracy of intelligibility and naturalness.

Lastly, User acceptance testing (UAT) is the last phase of the software testing process**,** the aim of undertaking user acceptance testing is to make sure how well speech based Amharic question answering is performing on the users so as to make sure that the system is accepted and usable by users. Ten users were selected to test the system.

1.6 Significance of the Study

This research shows some interesting results in this direction. It is supposed that the STT and TTS, However, are mainly developed for a general audience, could be a great support to the visually impaired users, disable and non-literate communities of the social order who have been put aside in recent technology.

The study is able to find out the required information from larger documents. The most significance of the study reducing the gap between a physical challenge of the people and the World Wide Web. The SBAQA usually has different intentions than a person using traditional Amharic text-based retrieval. Similarly, while the fundamental search algorithm is the same for both voice and traditional search. The research provide a solution for visually impaired users who prefer Amharic speech based for a blind person to access Amharic document. And also, the importance of the study is uses to in different commerce application through speech based human-machine interaction.

The findings of this study could contribute to the understanding of speech recognition components, question classification techniques in QA systems and speech synthesis components that are important for the designing of the system. The study can serve as a reference tool for people in academia to know what accounts question classifier for speech based local language QA system and operations in both study areas.

The findings of the study will be of enormous benefit to different application area government, multi, and bilateral development partners, for education purpose and towards designing future studies, and Ethiopian disabilities peoples especially to visually impaired users' and illiteracy.

1.7 Thesis Organization

This thesis is organized into six chapters. The first chapter discusses the background of the study and statement of the problem. It also presents general and specific objectives of the study, the methodology of the study, scope, and limitation of the research and significance of the formulated results.

In **Chapter two**, literature is reviewed to get a good understanding of the topic and find gaps related to the topic. Comparison and contrast of information retrieval and information extraction are stated. Related to Automatic speech recognition, speech synthesis, and Amharic language towards QA system are also described in this chapter. This Chapter present what a question answering mean to have a good understanding of QA and details about factoid question. We present the high-level architecture of a typical question answering system and its components. ASR is also discussed in detail in a subsection of this chapter. Speech synthesis is also briefly discussed in this section. The tools and techniques used in designing speech based Amharic question answering are stated here. Finally, related works done on the study in different languages are presented by different scholars.

Chapter three discusses the basic components of the proposed architecture for designing a speech based Amharic question answering and their interaction. Different components of the proposed system were discussed. ASR component, Question answering components and speech synthesis used for proposed architecture are briefly discussed in this chapter.

Chapter four is about the detail designing (implementation prototype) of the system. It discusses the algorithms we used in achieving the goal of every component in the system.

Chapter five deals with the experiments done in every component and the results achieved together with explanations of how such results happen.

Finally, based on the findings of the study, conclusion, and recommendations for the research are stated in **chapter six**.

CHAPTER TWO
LITERATURE REVIEW

This study reviews the state of the art in designing an automatic question classification for speech based Amharic question answering especially to visually impaired users and handicap peoples to get relevant information from the large documents. It is a kind of information retrieval systems that retrieve precise information from large documents. Information retrieval (IR) is the way of finding documents of an unstructured nature (usually text) that satisfies users an information need from large collections (usually stored on computers) [3].

This chapter provides a review of the literature and related works so as to identify the gaps, approach, and techniques in five areas related to this study: question answering systems, Automatic speech recognition, speech synthesis, the Amharic language phonetics & writing systems and the integration of the systems.

2.1 Question Answering Systems

Question answering (QA) is the discipline of information retrieval that is characterized by information needs that are concerned with the retrieval of accurate answers to natural language questions from a textual corpus [7]. The purpose of a QA system is to find the correct answers to user arbitrary questions in both non-structured and structured collection of data [23], [24]. It takes the question of natural language as input search and extracts precise answers from the large collections of the document within the corpus. The previous information retrieval techniques display the ranked list of documents, not the precise answer to the users of questions. QA system is related to Natural Language Processing, Information Retrieval, Machine Learning, Knowledge Representation, Logic and Inference, Sematic Search [25].

In recent years, many international questions answering contests have been held at conferences and workshops, such as Text REtrieval Conference (TREC), Cross Language Evaluation Forum (CLEF) and NII Test Collection for IR Systems (NTCIR) [16]. QA system are classified into two main categories from the domain of the application (open domain or closed domain) [6].

Open domain question answering studies everything and rely on general ontology and world knowledge. Closed-domain question answering deals with questions under a specific domain (for

example, medicine, sport, weather forecasting, etc.) and can be seen as an easier task because NLP systems can exploit domain-specific knowledge frequently formalized in the ontology.

QA system attempts to deal with a wide range of question types including; fact, list, definition, how, why, hypothetical, semantically-constrained and cross-lingual questions [10]. The research attempt to an open domain of Factoid question answering since the research studied with such kind of questions. These include questions, such as "Who is the President of Ethiopia?"/"የኢትዮጵያ ፕሬዝዳንት ማን ነው?" and the expected answer is a person, and "How much does one kilo of sugar cost?" or "የአንድ ኪሎ ስኳር ዋጋ ስንት ነው?" The expected answer is numerical.

There are two types of semantic resources that have been utilized to answer factoid questions [7]: the first one is syntactic parsing techniques to analyze questions for syntactic structures and to find phrases in the knowledge source that match these structures. The second one is Ontologies to extract terms from questions and corpus sentences and to enrich these terms with semantically similar concepts [26]. This research, other than focusing on semantically similar ontology concepts, it focuses on syntactic parsing techniques which extract answers from a set of documents.

2.2 QA System Component

Figure 2.1 shows that QA system consists of three separate modules [6] [2]; question processing, document processing, and Answer Processing.

Question processing is the module which identifies the focus of the question, classifies the question type, derives the expected answer type, and reformulates the question into semantically equivalent multiple questions. Reformulation of a question into similar meaning questions is also known as question expansion and it boosts up the recall of the information retrieval system.

In Document Processing Information retrieval (IR) system can retrieve the top and related documents that will be subjected to the passage filtering which extracts passages that pinpoint possible answer strings. IR recall is very important for question answering because if no correct answers are present in a document, no further processing could be carried out to find an answer [8]. Precision and ranking of candidate passages can also affect question answering performance in the IR phase.

Answer Processing is one of the question answering system model, which is a distinguishing feature of question answering system and the usual sense of text retrieval system. This module is responsible for identifying answers, and then answer extraction will be applied to get the exact answer then finally validating the answer will be done. Answer extraction technology is becoming an influential and decisive factor on question answering system for the final results.

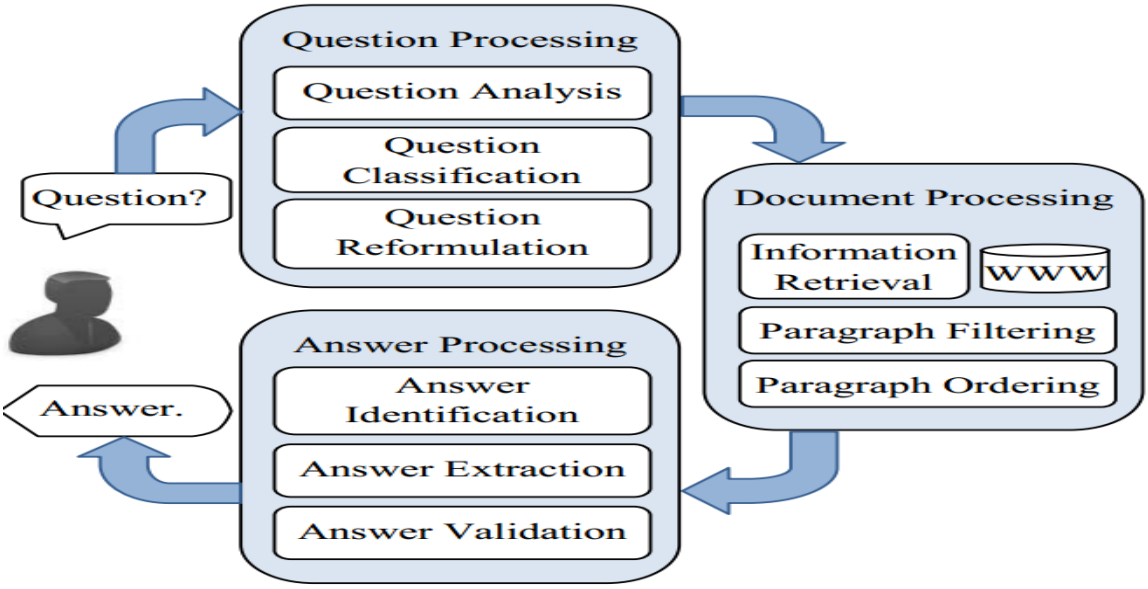


Figure 1-1 Architecture of an Open Domain QA System (adopted [6])

2.2.1 Question Processing Module

First, the ASR system posts a question as an input to the QA system, and the overall function of the question processing module is to analyze and process the question by creating some representation of the information requested. Therefore, the question processing module is required for [6], [2] question analysis, question type classification and question reformulation.

2.2.1.1 Question Analysis

Question analysis is also called “Question Focus”. For some questions, classifying the question and knowing its type is enough to find answers to the given question. The question formulated by “Who” obviously indicate the person. But, for example, a question was written by “What” is ambiguous and it doesn’t tell about the information of the question [2]. To address this ambiguity, an additional component which analyzes the question and identifies its **focus** is necessary. Question types can be broadly categorized as location, person/entity, definition, numeric,

explanation/list, true/false, time/event, choose, and so on. The focus of a question has been defined by [27] to be a word or sequence of words which indicate what information is being asked in the question. For instance, the question “What is the longest river in Ethiopia?” has the focus “longest river”. If both the question type (from the question classification component using SVM) and the focus are known, the system is able to more easily determine the type of answer required. Identifying the focus can be done using pattern matching rules, based on the question type classification [8].

2.2.1.2 Question Type Classification

In order to answer the question correctly, it is required to understand what type of information the question asks. Because knowing the type of a questionable to provide constraints on what constitutes relevant data (the answer), which helps other modules to correctly locate and verify an answer.

The question type classification component is, therefore, a useful, if not essential, a component in a QA system as it provides significant guidance about the nature of the required answer. Therefore, the question is first classified by its type: what, why, who, how, when, where questions and used pattern matching techniques. For instance, an example of a Question is በአፍሪካ ትልቁ አገር ማን ነው? (What is the largest country in Africa?), the question focus is አፍሪካ, አገር (Africa: Country) and the expected type of answer is ትልቁ (Country).

The SVM model based question classification takes benefit of flexibility over the Rule-Based question classification. It needs to modify hard-coded rules to handle new cases while the language model can be automatically maintained [28]. In Rule-Based, a pattern matching techniques if didn’t get the wanted keyword now cannot predict the expected answer types.

Question classification

Support Vector Machine (SVM) Classifier was developed by [29] in 1995 based on classification and regression theory. SVM is a supervised learning algorithm typically used for classification problems like prediction, text categorization, handwritten character recognition, image classification, facial expression classification, and so on. This means if we have some sets of things

classified but we know nothing about how we classified it, or we don't know the rules used for classification and when a new data comes, SVM can predict which set the data should belong to.

SVM is basically a binary classifier and it's worked using the numerical statically number. It takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. 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. There are other supervised learning algorithms like Naïve Bayes, Neural Networks, and Decision Trees. But for text classification purposes, SVM is proved to give better performance [29].

Support Vector Machine (SVM)

A support vector machine (SVM) is a supervised machine learning algorithm. It can be used for classification and regression. An SVM performs classification by finding the hyperplane [30]that separates between a set of objects that have different classes. This hyperplane is chosen in such a way that maximizes the margin between the two classes to reduce noise and increase the accuracy of the results. Figure 2-2 shows how an SVM classifies objects:

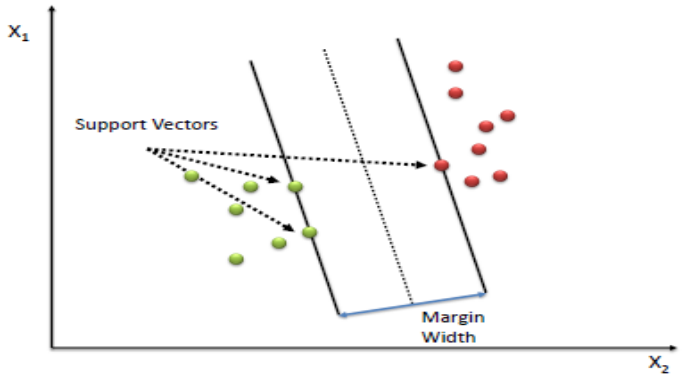


Figure 2-2 Classifying objects with a support vector machine

The perception behind SVM that to classification and regression [3]:

- The vectors that are on the margins are called support vectors. Support vectors are data points that lie on the margin, points (instances) are like vectors p=(x_1, x_2, x_3, x_4……....x_n)
- SVM finds the closest two points from the two classes (see figure 2-2), that support (define) the best separating line or plane.
- Then SVM draws a line connecting them
- After that, SVM decides that the best separating line is the line that bisects and is perpendicular to the connected line.

The data sets represented as D= {(x_1, y2),(x_2,y_2),……(X_n, Y_n)} n is the number of instances, xi is the i^{th} point (example of vector) and i is the class associated with the point y_i can either -1 or +1 [3]. A key concept required here is the dot product between two vectors.

A= [a1, a2, a3, …….. an], B= [b, b2, b3, …….. bn]

A.B = a1.b1+a2.b2+a3.b3……. +an.bn

It can define separating (decision) hyper plane in terms of an intercept term b and a normal vector $\vec{w}$ which is perpendicular to the hyper plane (commonly referred to as the weight vector). Then to choose among all the hyper planes that are perpendicular to the normal vector, we specify the intercept term b. all points vector $\vec{x}$ on the hyper plane satisfy $\vec{w}^T \cdot \vec{x} = -b$ as the hyper plane is perpendicular to the normal vector. Then represent the training data set as D= {($\overrightarrow{x1} \cdot \overrightarrow{y1}$)} as a pair of a point and a class label corresponding it. Finally, the linear classifier becomes f($\vec{x}$)=sign($\vec{w}^T \cdot \vec{x} + b$).

There are different types of SVM [31]; C-SVC (multi-class classification), nu-SVC (multi-class classification), one-class SVM, epsilon-SVR (regression), nu-SVR (regression). Furthermore, different kernel functions in SVM [31]; linear: u'*v, polynomial: (gamma*u'*v + coef0) ^degree, radial basis function: exp (-gamma*|u-v|^2), sigmoid: tanh (gamma*u'*v + coef0), precompiled kernel (kernel values in training set file). In this study used one-class SVM and LibLinear kernel type.

The most challenges of SVM question classifier are taken a huge amount of time to train a large dataset and doesn’t work well when there is no clear separation between target classes.

2.2.1.3 Expected Answer Type Classification

Answer type classification is a subsequent and related component to question classification. Answer type classification is based on a mapping of the predict question classification. Once a question has been classified, a simple SVM classifier mapping would be used to determine the potential answer types. Again, because question classification can be ambiguous and match word missing, the system should allow for multiple answer types. For example, for the question (በአፍሪካ ትልቁ አገር ማን ነው?) the expected answer classification will be country (አገር) but the country answered should be ትልቁ.

2.2.1.4 Question Reformulation

Once the "focus" and "question type" are identified, the module forms a list of keywords to be passed to the information retrieval component in the document processing module. The process of extracting keywords could be performed with the aid of standard techniques such as named-entity recognition, stop-word lists, and part-of-speech taggers, etc.

2.2.2 Document Processing Module

The document processing module in QA system is also commonly referred to as paragraph indexing module, where the reformulated question is submitted to the information retrieval system, which in turn retrieves a ranked list of Top relevant documents. The document processing module usually depends on one or more information retrieval system to gather information from a collection of document corpora which almost always involves the WWW as at least one of these corpora. The documents returned by the information retrieval system is then filtered and ordered. Therefore, the main goal of the document processing module is to create a set of candidate ordered paragraphs that contain the answer(s), and in order to achieve this goal, the document processing module is required to information retrieval, paragraph filtering then final paragraph ordering [6].

2.2.2.1 Information Retrieval (IR)

The goal of the information retrieval system is to retrieve accurate results in response to a question submitted by the user and to rank these results according to their relevance.

QA systems do not depend on IR systems which use the cosine vector space model for measuring the similarity between documents and queries**.** This is mainly because a QA system usually wants documents to be retrieved only when all keywords are present in the document. This is because the keywords have been carefully selected and reformulated by the Question Processing module. IR systems on the other hand based on cosine similarity often return documents even if not all keywords are presented.

2.2.2.2 Paragraph Filtering

Paragraph filtering can be used to reduce the number of candidate documents and to reduce the amount of candidate text from each document. The concept of paragraph filtering is based on the principle that the most relevant documents should contain the question keywords in a few neighboring Paragraphs, rather than dispersed over the entire document. Therefore, if the keywords are all found in some set of N consecutive paragraphs, then that set of paragraphs will be returned, otherwise, the document is discarded from further processing.

2.2.2.3 Paragraph Ordering

The aim of paragraph ordering is to rank the paragraphs according to an acceptability degree of containing the correct answer. Paragraph ordering is performed using standard radix sort algorithm. The radix sort involves three different scores to order paragraphs [6]:

- **Same word sequence score:** the number of words from the question that are recognized in the same sequence within the current paragraph window.
- **Distance score**: the number of words that separate the most distant keywords in the current paragraph window.
- **Missing keyword score:** the number of unmatched keywords in the current paragraph window. The purpose of minimizing for shortening documents into paragraphs is making a faster system.

2.2.3 Answer Processing Module

As the final phase in the QA architecture, the answer processing module is responsible for identifying, extracting and validating answers from the set of ordered paragraphs passed to it from the document processing module [6].

2.2.3.1 Answer Identification

The answer type which was determined during question processing is crucial to the identification of the answer. Since usually, the answer type is not explicit in the question or the answer, it is necessary to depend on a parser to recognize named entities (e.g. names of persons and organizations, monetary units, dates, etc.). And also, using a part-of-speech tagger (e.g., Brill tagger) can help to enable recognition of answer candidates within identified paragraphs. The recognition of the answer type returned by the parser creates a candidate answer. The extraction of the answer and its validation are based on a set of heuristics.

2.2.3.2 Answer Extraction

The parser enables the recognition of the answer candidates in the paragraphs. So, once an answer candidate has been identified, a set of heuristics is applied in order to extract only the relevant word or phrase that answers the question. Extraction can be based on measures of distance between keywords, numbers of keywords matched and other similar heuristic metrics. Commonly, if no match is found, QA systems would fall back to delivering the best-ranked paragraph. Unfortunately, given the tightening requirements of the TREC QA track, such behavior is no longer useful.

2.2.3.3 Answer Validation

Confidence in the correctness of an answer can be increased in a number of ways. One way is to use a lexical resource like WordNet to validate that a candidate response was of the correct answer type. Also, specific knowledge sources can also be used as a second opinion to check answers to questions within specific domains.

2.3 Automatic Speech Recognition

Automatic Speech Recognition (ASR), as described by [32], is the "decoding of the information conveyed by a speech signal and its transcription into a set of characters". Nowadays, Automatic speech recognition applications are becoming more and more useful. ASR [33] system allows a computer to identify the words that a person speaks into a microphone or telephone and decoding that convert it into written text.

Automatic speech recognition systems can be separated into several different classes by describing the type of speech utterance, type of speaker model, type of channel and the type of vocabulary that they have the ability to recognize [33]. The types of speech categories' are speech utterance, speaker model, and types of vocabulary [15, 33, 34].

2.3.1 Types of Speech Utterance

Isolated or continuous speech: both are types of a speech utterance. In isolated word recognizers, a user must pause between words, which mean artificial pause should be inserted before and after each word, whereas in continuous speech recognizers, a user speaks naturally [35].

Read or spontaneous speech: a speech recognizer may be designed to recognize only read the speech or to recognize spontaneous speech. Spontaneous speech can be characterized by varied speaking rate, filled pauses, corrections, hesitations, repetitions, partial words, dis-fluencies, "sloppy" pronunciation but read speech requires prepare the reading material, ready for reading and reading properly. The major differences between spontaneous speech and read speech are (1) readers tend to make the boundaries between tone units at different points, (2) the position of the stresses differs, and (3) there are fewer pauses in reading speech, and the location of these differs between readers [33].

2.3.2 Types of Speaker Model

Speaker dependent or independent: Speaker-independent systems are capable of recognizing speech from any new speaker. Speaker dependent systems, on the other hand, recognize speech from those people whose speech is used during the development of the recognizer [17].

2.3.3 Types of Vocabulary

The size of the vocabulary of an automatic speech recognition system affects the complexity, processing requirements and the accuracy of the system. Some applications only require a few words (e.g. numbers only), others require very large dictionaries (e.g. dictation machines).

Small, Medium or Large Vocabulary system: small vocabulary recognition systems are those which have a vocabulary size of 1 to 99 whereas medium and large vocabulary system have a vocabulary size of 100 - 999 and 1000 or more words, respectively [34, 36].

As mentioned above part of speech constraints, and the environment variability, channel variability, speaking style, sex, age, the speed of speeches makes the ASR system more complex. This study is now focusing on ASR systems that incorporate three features: medium vocabularies, continuous speech capabilities, and speaker independence [34].

2.3.4 Approach of Automatic Speech Recognitions

For developing ASR system, four basic approaches such as; acoustic phonetic, template matching (pattern recognition), stochastic approaches, and Artificial Intelligence should be considered.

2.3.4.1 Acoustic Phonetic Approach

It is based on the theory of acoustic phonetics that postulates there exists finite, distinctive phonetic units in spoken language and that the phonetic units are broadly characterized by a set of properties that are manifested in the speech signal or its spectrum over time. Even though the acoustic properties of phonetic units are highly variable, both with speakers and with neighboring phonetic units (the so-called co-articulation of sounds), it is assumed that the rules governing the variability are straightforward and can readily be learned and applied in a practical situation [37].

In this approach, the research of [38] stated four stages. The first stage is the speech spectral analysis that parameterizes the speech signal into feature vectors; the most popular set of them is the Mel Frequency Cepstral Coefficients (MFCC) [39]. The next stage is a feature extraction stage that converts the spectral measurements into a set of features that describe the acoustic properties of the different phonetic units. The third stage is segmentation and labeling stage in which the speech signal is segmented into isolated regions, followed by attaching one or more phonetic labels

to each segmented region. The last stage finds a valid word from the phonetic label sequences produced by the segmentation to labeling. The acoustic-phonetic approach has not been widely used for Automatic speech recognition [40].

2.3.4.2 Pattern Recognition Approach

According to [37], Pattern recognition approach is used to extract patterns based on certain conditions which separate one class from the others. It has four stages i.e. feature measurement, pattern training, and pattern classification and decision logic. In order to define a test pattern, a sequence of measurements is made on the input signal. The reference pattern is created by taking one or more test patterns corresponding to speech sounds. This can be in the form of a speech template or a statistical model i.e. HMM and can be applied to a sound, a word, or a phrase. In the pattern classification stage of the approach, a direct comparison is made between the unknown test pattern and class reference pattern and a measure of distance between them is computed. Decision logic stage determines the identity of the unknown according to the integrity of match of the patterns.

There is two type of pattern methods [38]; i.e., Template and stochastic method. In template method, unknown speech is compared against a set of templates in order to find the best match. A template method provides good Automatic speech recognitions performance for a variety of application. Stochastic methods depict the use of probabilistic models to deal with incomplete information. In ASR, incompleteness arises from various sources. For example, speaker variability, contextual effect, etc. The most popular stochastic approach nowadays is hidden Markov modeling (HMM) (see section 2.3.4.3 for a brief description).

2.3.4.3 Artificial Intelligence (AI) Approach

The Artificial Intelligence approach is a mixture of the acoustic-phonetic approach and pattern recognition approach. Most of the time, some scholars developed recognition system by using acoustic-phonetic knowledge to construct classification rules for speech sounds. While template based methods provided little insight into human speech processing, these methods have been very effective in the design of a variety of ASR systems. On the other hand, linguistic and phonetic literature provided insights about human speech processing. However, this approach had only partial success due to the complicatedness in quantifying expert knowledge. Another problem is

the integration of levels of human knowledge; i.e., phonetics, lexical access, syntax, semantics, and pragmatics [38]**.**

Stochastic modeling involves no direct matching between stored models and the input. Instead of that, it is based on complex statistical and probabilistic analyses to extract patterns inductively from the given speech data. A well-known and most widely used stochastic model is the Hidden Markov Model (HMM) [34]which is also used in this study unlike template matching, stochastic modeling.

2.3.5 Hidden Markov Model

Hidden Markov Model (HMM) which is also called Markov sources or probabilistic functions of Markov Chains is one of the key technologies developed in the 1980s at Princeton University [41]. It uses a stochastic approach to recognizing speech. HMM has been applied with great success in the field of ASR during the last three decades [42]. The widespread use of HMM modeling for ASR is due to the availability of efficient algorithms for parameter estimation procedures that involve maximizing likelihood (ML) given the model. HMM provides a simple and effective framework for modeling time-varying spectral vector sequences [43]. Moreover, Gales and Young have indicated that HMM is the best and more appropriate approach to develop speaker independent medium vocabulary continues word recognition. Supporting this idea, Huang, and Deng [44] indicated that HMMs have improvement popularity because of their ability to estimate parameters from a large amount of data automatically, their simplicity as well computational feasibility. Thus, it was used to develop the required prototype Amharic (አማርኛ) recognizer for converting speech question into the text to search for relevant element document that contains answers.

HMM is defined by the following elements [44]:

- N, the number of states in the model. Individual states are labeled as {1, 2… N} and the state at time t are denoted by qt.
- T, the number of distinct observation symbols per state. Observation symbols correspond to the physical output of the system being modeled. Individual observation symbols are denoted as Y = {y1, y2, y3 … , yT}

- The state transition probability distribution A={ ***a**ij* } where

  $\alpha_{ij}=p(q_{t+1} = j/q_t = i)$, $1\leq i, j\leq N$
- The observation symbol probability distribution, B={ ***b**j*(k)} in which

  bj (k) = P (Ot = yk / qt = j), $1 \leq k \leq M$

  bj(k) defines the symbol distribution in state i, j=1,2,…,N
- The initial state distribution π {πi} in which

  $\Pi_i = p(q_{1 = i)}$, $1 \leq i \leq N$

2.4 Speech Synthesis Methods

A speech synthesis is an application that converts text into spoken word, by analyzing and processing the text using Natural Language Processing (NLP) and Digital Signal Processing (DSP) technology [45]. There are two main approaches to speech synthesizers [46]: rule-based and data-driven methods. Rule-Based speech synthesis approach aims at re-creating the same sounds as those created by human speech system in modeling the human speech production system. In contrast, data-driven synthesizers generate a speech by arranging various segments of prerecorded speech in a specific order.

The TTS system gets the text as input and then analyses, pre-processes and speech synthesis the text. The TTS engine usually generates sound data in an audio format as the output [45].

The major purposes of speech synthesis techniques are to convert a chain of phonetic symbols into artificial speech that transforms a given linguistic representation and to generate speech automatically with information about intonation and stress i.e. prosody [45].The research of TTS is studied in a different field: from acoustic phonetics (speech production and perception) over morphology (pronunciation) and syntax (parts of speech, grammar), to speech signal processing (synthesis) [47].

2.4.1 Architecture of TTS

Basically, a speech synthesizer has two major measurement points; natural and intelligent. The text to speech conversion involves the following steps: Text pre-processing, text analysis, text phonetization, prosody generation and then the speech synthesis using various algorithms [48]. The steps followed to convert text to speech are described in Figure 2.3.

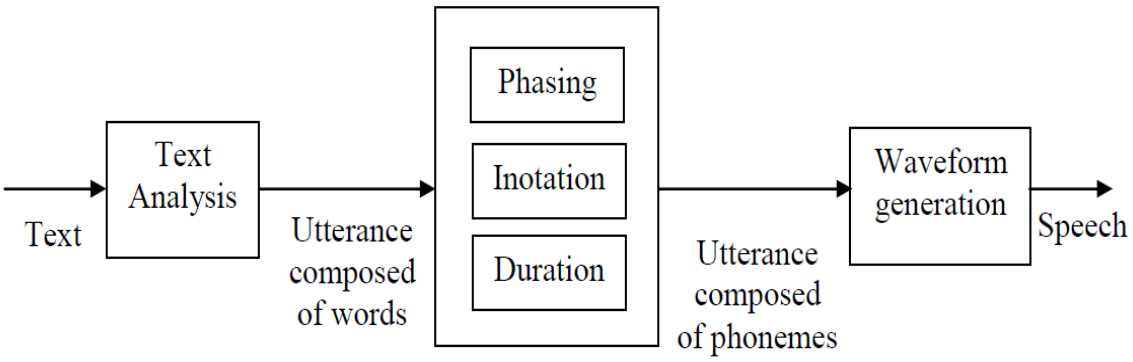


Figure 2-3 Steps in TTS System adopted from [49]

Figure 2.3 depicted that, TTS system contains two components: the Natural Language Processing (NLP) and the Digital Signal Processing (DSP) [45].

2.4.2 Natural Language Processing (NLP)

The NLP component consists of three processing stages: text analysis, automatic phonetization and prosody generation [49].

A. Text Analysis

It is the first stage of natural language processing by which it consists of four modules: pre-processing module, a morphological analysis module, the contextual analysis module and syntactic-prosodic parser [49].

In a **pre-processing module**, the input sentences are organized into lists of words. The system must first identify these words or tokens in order to find their pronunciations. In other words, the first step in the text analysis is to make chunks/split out of the input text - tokenizing it. There are many tokens in a text that appear in a way where their pronunciation has no obvious relationship with their appearances such as abbreviations, acronyms, and numbers. Apart from tokenization, normalization is needed where a transformation of these tokens into full text is done.

In a **morphological analysis module**, all possible part-of-speech categories for each word are proposed on the basis of their spelling. For example, inflected, derived and compound words are decomposed into their morphs by simple grammar rules. The token-to-word conversion creates the

orthographic form of the token. For the token “Mr.” the orthographic form expand the token and formed “Mister”, the token “12” gets the orthographic form “twelve” and “1997” is transformed to “nineteen ninety-seven” [50].

A **contextual analysis module** considers the word in their contexts. This is important to reduce possible part-of-speech categories of the word by simple regular grammars using lexicons of stems and affixes.

In a **syntactic-prosodic parser**, the search space is examined and the text structure is found. The parser organizes the text into clause and phrase like constituents. Subsequently, the parser tries to relate these into their expected prosodic realization [49].

B. Phonetization

The second module is the Letter-to-Sound module (LTS), where the words are phonetically transcribed. In this stage, the module also maps sequences of graphoneme into sequences of phoneme with possible diacritic information, such as stress and other prosodic features that are important to fluency in naturally sounding speech

C. Prosody Generator

The last module, the prosody generator, is where certain properties of the speech signal, such as pitch, loudness, and syllable length are processed. Prosodic features create a segmentation of the speech chain into groups of syllables. This gives rise to the grouping of syllables and words in larger chunks.

2.4.3 Digital Signal Processing (DSP)

The second TTS component is DSP [50] that enables after the preprocessed and analyzed the phonetic representation of phonemes with correct intonation, duration, and stress is used to generate speech. Speech sound is finally generated with one the methods: articulatory synthesis, formant synthesis, concatenation synthesis, Hidden Markov Model (HMM) and Unit Selection Synthesis.

The speech synthesis [51] techniques can be classified by generation, The First Generation techniques includes Formant Synthesis and Articulatory Synthesis. Second Generation Techniques incorporates concatenate synthesis and Sinusoidal synthesis. The third Generation includes Hidden Markov Model (HMM) and Unit Selection Synthesis. These have been classified on the basis of how they parameterize the speech for storage and synthesize [47]. Even if it is hard to find the best method that satisfies the naturalness and intelligibility for each method fall into one of the above categories.

2.4.3.1 Formant Synthesis

Formant Synthesis works by using individually controllable formant filters, which can be set to produce correct estimations of the vocal track transfer function [51]. In at runtime formant synthesis does not use human speech samples [52]. A formant synthesis system synthesizes speech using acoustic models of speech production and it is also called parametric based synthesis. Formant synthesis uses a relatively simple system to select from a small number of parameters, to control a mathematical model of the speech sounds. A set of parameters is picked for each speech sound and they are then joined up to make the speech. This stream of parameters is turned into synthetic speech using the model [53]. Formant synthesis generates highly intelligible, but not completely natural sounding speech [52] [47]. A formant synthesis uses a smaller amount of memory and microprocessor power requirement compare with other techniques; therefore it is suitable for the limited devices [54].

2.4.3.2 Articulatory Synthesis

The idea in articulatory synthesis is also mathematically modeled speech production, but models the speech production mechanism itself use a complex physical model of the human vocal tract. Articulatory synthesizers, rather than modeling the sound, they model human speech production mechanisms; in some cases, they might give a more natural sounding speech than formant synthesis. They classify speech in terms of movements of the articulators, the tongue, lips and velum, and the vibrations of the vocal cord. Text to be synthesized is converted from a phonetic and prosodic description into a sequence of such movements and the synchronizations between their movements calculated. A complex computational model of the physics of a human vocal tract

is then used to generate a speech signal. Articulatory synthesis is a computationally intensive process and is not widely available outside the laboratory [53].

2.4.3.3 Concatenate Synthesis

Concatenate synthesis generates speech output by concatenating the unit selector segments of recorded speech from diaphone database.

Generally, concatenate synthesis generates the natural sounding synthesized speech [51]. This speech synthesis technique uses actual snippets of recorded speech that were cut from -+ recordings and stored in an inventory called voice database, either as waveforms (uncorded) or encoded by a suitable speech coding method. Elementary units i.e., speech segments are, for example, phones (a vowel or a consonant), or phone-to- phone transitions (diaphones) that encompass the second half of one phone plus the first half of the next phone (e.g., a vowel-to-consonant transition). Concatenate synthesis itself then strings together (concatenates) units selected from the voice database, and, after optional decoding, outputs the resulting speech signal. Because concatenate systems use snippets of recorded speech, they have the highest potential for sounding "natural" [48]. However, differences between natural variations in speech and the automatic segmentation of the waveforms can cause audible glitches in the output and can disturb the intelligibility [54]. Unit selection is part of concatenating approach. The unit selection paradigm is a cluster based technique where units of the same type are clustered based on the acoustic differences. The clusters are then indexed based on higher level phonetic and prosodic context. During synthesis, an appropriate unit is chosen from multiple instances of that unit based on minimization of joining cost and concatenation cost. [55]

2.4.3.4 HMM-Based Speech Synthesis

Hidden Markov Model (HMM)-based speech synthesis can be generated from a concatenation of mathematical models that represent the statistics of the acoustic parameters of each phonetic unit. Furthermore, unlike the acoustic models of formant synthesis, these statistical acoustic models have created automatically from real speech data. As a result, HMM-based speech synthesis has gained the advantage from both the flexibility of formant methods and the quality of concatenating synthesis [56]. Knife [57] worked on Sub-word based Amharic word recognition and for an experiment using Hidden Markov Model (HMM).

Hidden Markov Models have proven to be efficient parametric models for the scientific study of speech in the framework of speech synthesis [58]. It is a familiar model which can capture the dynamics of the training data by using states where each state has the capability to generate all possible observable values [23]. HMM attempts to model the probability distribution of a feature vector according to its actual state, and it also models the dynamics of vector sequences with transition probabilities between states.

The aim of this research has been to produce acceptably natural sounding speech, for the speech synthesis component developed by using unit selector concatenate synthesis approach.

2.5 Amharic Writing Systems and its Phonetics

Amharic is the working language of the government of Ethiopia that can be a written language for at least 500 years. In 1998, there were 17.4 million (17 million in Ethiopia and 400,000 in other countries) first language speakers worldwide and estimated 5 million as second language speakers. It is the second most spoken Semitic language after Arabic. There are five dialects: Gondar Amharic, Gojam Amharic, and Showa Amharic (standard written and spoken Amharic) [59]. Amharic is one of the languages which has its own script called Fidel.

2.5.1 Amharic Writing Systems

As mentioned in the above topic, Amharic is one of the languages that have its own writing system. It was written with a version of the Ge'ez script known as ፊደል (Fidel) [60]. Amharic has its own characterizing phonetic, phonological and morphological properties. For example, it has a set of speech sounds that is not found in other languages. For example, the following sounds are not found in English: [p`], [tS`], [s`], [t`], and [q] [60]. Amharic writing phonetics allows anyone to write Amharic texts if s/he can speak Amharic and has the knowledge of the Amharic alphabet. The Amharic writing system has the following limitations, however [61],

1. There are several duplicate characters that are normally used interchangeably.
2. It lacks a mechanism to mark germination of consonants
3. There exists ambiguity between sixth-order symbols with and without a vowel in different contexts.

These problems need to be properly addressed to realize an ASR system that accurately recognizes and transcribes Amharic Speech. The intuitive solution for the first problem is to use one form of the duplicate characters. For instance (ሰ, ሠ), (ፀ, ጸ), (ሃ, ሓ, ሀ, ኀ, ኃ), (ዐ, ዓ, አ, ኣ) are duplicates. These symbols which are used for writing are not adequate to denote the frequent germination encountered in Amharic speech. For instance, the Amharic word "ገና" gives completely different meanings based on whether the sound "N" is geminated or not. "ገና" gEna (meaning Yet, Still), and the second meaning is "ገና" gEn-na (meaning Christmas).

The Amharic orthography, as represented in the Amharic character set also called Fidel consists of 276 distinct symbols. In addition, there are 20 numerals and 8 punctuations marks. It is characterized by a quite homogeneous phonology distinguishing between 234 distinct Consonant-Vowel (CV) syllables [36].

Grammatical Structure

According to Getahun [62], the Amharic language has word categories in its grammar as ስም (noun), ግስ (verb), ቅፅሌ (adjective), ተውሳከ ግስ (Adverb), መስተዋዴዴ (preposition), and ተውሊጠ ስም (pronoun). Those categories can be combined and construct statement, phrase or integrative sentence using question marks.

Factoid questions need answers that are nouns most of the time and understanding the structure of a sentence in which a noun can be easily indicated to extract the correct answer is very important. The Amharic basic sentence is constructed from noun phrase (ስማዊ ሀረግ) and a verb phrase (ግሳዊ ሀረግ) [8].

Integrative Words

There are many integrative words to construct a question in the Amharic language. Seid and Mulugeta [8] have collected sample questions from Radio and Television puzzle, television Question and Answer Game and Linguistic resources. As shown in Table 2:1. These Amharic question particles are classified into WH question types in consultation with language professionals [15]. It is Rule-Based question classifier. But, in this study used independent question classifier model using SVM (Support Vector Machines) is used.

Table 2-1 Sample Classification of Amharic Question Particles in Rule-Based

What	Why	Where	Which	When	Who/whom	How
ምን, ምንድን , የምን	ለምን	የት,በየት, የየት, ወደት,ከየት, እስከየት, ወደየት, ወደምን,	ከየትኛው, የትኛው, ምኑን, በምን,	መቸ, እስከመቸ, መቼ, በመቸ, ለመቸ, ምንጊዜ, እስከምን,	ማን, በማን ማነው, እነማን, ማንን, እነማንን, ማንማን, የቱ, ማንኛይቱ, የቷ, ማንኛዋ, የቲቱ, የማን, ለማን, የቶቹ, የትኛዋ, የትኞቹ, የየትኛው, በየትኞቹ, ምንህን, ምንሽን, ምናችሁን, ምኔ,	ምንያህሌ, ስንት, ስንቱ, ከስንት, እንዳት, እንደ,

2.5.2 Amharic language Grapheme to Phoneme

In the case of the TTS and STT systems, a language correlated between grapheme to phoneme means to convert a target word from its written form (grapheme) to its pronunciation form (phoneme). The conversion prearrangement is designed based on the orthographic ordering of the writing and the acoustic similarity of the letters. There is a translation of Amharic symbols to use in English symbol characters. Solomon and Wolfgang [60] noted that in Amharic orthographic structure a one-to-one correspondence amongst the sounds and the graphic symbols exist, except for the germination of consonants and some redundant symbols. Amharic languages is a tone (or tonal) language, where similarly written words have different meanings based on the tone (high or low) with which they are pronounced.

The phone set of the language is defined with the corresponding features like voicing, tongue position, tongue height, place of articulation, and manner of articulation. From the studies reported in [62], the researcher derived a set of phonetic features for the 39 phones. A set of 38 phones, 7 vowels, and 31 consonants have seven orders. Six of them are CV combinations while the seventh is the consonant itself, makes up the complete inventory of sounds for the Amharic language. One additional consonant /v/ included summing up to a total of 39 phonemes [62]. Amharic has a rich variety of vocalic sounds. The two major categories of Amharic phones are consonants and vowels [36]. This representation of each character in 7 different forms makes the number of core characters to be 231 (33 * 7), shown in **appendix I**, also called syllographs (ፊደል).

Consonants

Amharic consonants are generally classified as stops, fricatives, nasals, liquids, and semi-vowels [36]. Table 2:2 shows the phonetic representation of the consonants of Amharic as to their manner of articulation, voicing, and place of articulation.

Table 2-2 categories of Amharic consonants

		Labials		***Alveolar***		***Palatals***		***Velars***		***Labio-velar***		***Glottals***	
Stops	Voiceless	p	ፐ	T	ት			K	ከ	Kx	ኳ	Ax	ዕ
	Voiced	b	ብ	D	ድ			G	ግ	Gx	ጓ	H	ሀ
	Glottalized	px	ጱ	Tx	ጥ			Q	ቅ	Qx	ቋ	Hx	ኋ
Affricatives	Voiceless	F	ፍ	X	ስ	Sx	ሽ						
	Voiced	V	ቭ	Z	ዝ	Zx	ዥ						
	Glottalized			Xx	ጽ								
Affricatives	Voiceless					C	ች						
Nasals	Voiced					J	ጅ						
	Glottalized			N	ን	Cx	ጭ						
	Voiced	M	ም	L	ር	Nx	ኝ						
Liquids	Voiced			R	ል								
Glides		W	ው			Y	ይ						

Vowels

In Amharic phonetics, there are 7 vowels. These are ኧ [A], ኡ [u], ኢ [i], ኣ[a], ኤ [e], እ[I], ኦ[o]. Those seven vowels are categorized according to the place of articulation as shown, in figure 2.4.

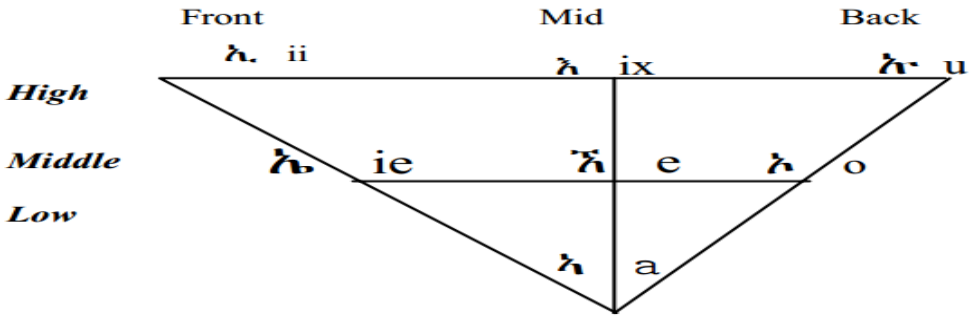


Figure 2-4 Vowels with their features (mainly adopted from [15])

According to figure 2-4 above, the high vowels are ኢ [I], ኧ [I], ኡ [u], low vowels is ኣ[a] and ኧ [A], ኤ [e], ኦ [o] are mid vowels. In combination, CV and C of the phoneme in ASR and speech synthesis are used.

2.6 Software Tools

There are different tools available for constructing speech based Amharic question answering, such as sphinx-4 tools for speech recognizing and decoding, Lucene for question answering system & LibSVM for question classification and general, integrating all modules are used NetBeans.

2.6.1 Sphinx Tools

The Sphinx ASR system is an open source, high-performance ASR system developed by the CMU Sphinx project that can be used in building ASR applications [63]. It also includes related resources, such as acoustic model trainer, and language model builder. Therefore, Sphinx is available as a complete ASR trainer and decoder. It has different versions available for training and decoding. Here are the different sphinx versions and their difference below [64].

Sphinx-2 is a high-speed large vocabulary speech recognizer. It is usually used in dialogue systems and pronunciation learning system.

Sphinx-3 is a slightly slower but more accurate Large Vocabulary ASR System. It is used as a server implementation of Sphinx for evaluation. It uses HMMs with continuous output PDFs. It supports several modes of operations. The more accurate model, “known as the flat decoder”, comes from the original Sphinx-3 release.

Sphinx-4: is completely written in Java. It provides high accuracy and speed performance comparable to the state of the art. It uses HMMs with continuous output PDFs. Sphinx-4 uses models trained by the sphinx-3 trainer.

PocketSPhinx: is speech recognizer which can be used in embedded devices. It is highly optimized for CPUs, such as ARMs. PocketSphinx is CMU's fastest speech recognizer system. It uses HMMs with semi-continuous output PDFs. Even though it is not accurate as Sphinx-3 or Sphinx-4, it runs at real time, and therefore it is good for live applications.

SphinxTrain: is a suite of tools which carry out acoustic model training. It is CMU sphinx training package. It trains continues or semi-continues models with vector quantization for SPHINX decoder versions 3 or 4, which is also used by PocketSphinx. The sphinx-3 format can also be converted to Sphinx-2 format under some conditions related to sphinx-2 limitations. SphinxTrain supports MFCC and PLP coefficients with delta or delta-delta features.

CMU-Cambridge Language Modeling Toolkit is a suite of tools which carry out the language model training.

Sphinxbase provides a common set of the library used by several projects in CMU sphinx. According to the tests performed at CMU which compares Sphinx-4 with 33 Sphinx-3 (flat decoder) and to Sphinx3.3 (fast decoder) in several different tasks, ranging from digits recognitions to medium-large vocabulary, Sphinx-3 (flat decoder) is often the most accurate, but Sphinx-4 is faster and more accurate than Sphinx-3 in some of these tests [64]. The decision about which version to use depends on the type of application being developed.

Concerning Programming language**,** Sphinx-2 and sphinx-3 are written in C and Sphinx-4 written fully in Java programming. Accuracy and Speed of the different sphinx versions are reported by different scholars [63]. The accuracy can be summarized as Sphinx-4 ≥ Sphinx-3>Sphinx-2; and in terms of processing time, Sphinx-4 ≥ Sphinx-3 ≥ Sphinx-2. Therefore, Sphinx-4 used as a complete ASR decoder for this study.

2.6.2 Lucene Indexing

Lucene is free open source implemented in java. It is a member of the Apache Jakarta group of projects and it is under the license of the liberal Apache Software License. Lucene can index any text-based information we like and then find it later based on various search criteria. Although Lucene only works with text, there are other add-ons to Lucene that allow us to index Word documents, PDF files, any text files, XML, or HTML pages. It is with high performance and scalable IR library which can help to add indexing and searching functionalities to different applications. In this study, the document retrieval prototype will be implemented using Lucene API. Related to IR the basic functionalities of Lucene include Indexing, Question parsing, and Searching are discussed [65].

Lucene is not an application but it is specifically an Application Programming Interface (API). This means that all the hard parts have been done, but the easy programming has been left to us to develop towards our application needs. The following explanation is partly adopted from the Lucene tutorial in [65]: Let's take a look basic Lucene classes that are useful to build a search engine.

Document - The Document class represents a document in Lucene. We index Document objects and get Document objects back when we do a search.

Field - The Field class represents a section of a Document. The Field object will contain a name for the section and the actual data. For example, (E:\AmharicSVMspeechQA_1\datadir\textdata) is one Field object for a document representing.

Analyzer - The Analyzer class is an abstract class that is used to provide an interface that will take a document and turns it into tokens that can be indexed. There are several useful implementations of this class but the most commonly used is the Standard Analyzer class.

Index Writer - The Index Writer class is used to create and maintain indexes. We put data into the Index, and then do searches on the Index to get results out. To build the Index, we use an Index Writer object. Document objects are stored in the Index, but they have to be put into the Index at some point. We have to select what data to enter in, and convert them into Documents.

Index Searcher - The Index Searcher class is used to search through an index which was created by the Index Writer.

Question Parser - The Question Parser class is used to build a parser that can search through an index. Queries can be quite complicated, so Lucene includes a tool to help generate Question objects, called a Question Parser. The Question Parser takes a question string, much like what we would put into a search engine, and generates a Question object.

Question - The Question class is an abstract class that contains the search criteria created by the Question Parser. The search itself is a Question object, which we pass into IndexSearcher.search(). IndexSearcher.search() returns a Hits object, which contains a Vector of Document objects.

Hits - The Hits class contains the Document objects that are returned by running the Question object against the index.

Generally, Lucene handles the indexing, searching and retrieving, but it doesn't handle:

- Managing the process (instantiating the objects and hooking them together, both for indexing and for searching)
- Selecting the data files, parsing the data files
- getting the search string from the user
- displaying the search results to the user

2.6.3 LIBSVM

LibSVM is an integrated LibSVM classifier software to support vector classification, (C-SVC, nu-SVC), regression (epsilon-SVR, nu-SVR) and distribution estimation (one-class SVM). It supports multi-class classification. LIBSVM provides a simple interface where users can easily link it with their own programs. Main features of LIBSVM include [66]:

- Different SVM formulations.
- Efficient multi-class classification.
- Cross-validation for model selection.
- Probability estimates.
- Various kernels (including precompiled kernel matrix).

- Weighted SVM for unbalanced data.
- Both C++ and Java sources.
- GUI demonstrating SVM classification and regression.
- Python, R, MATLAB, Perl, Ruby, Weka, Common LISP, CLISP, Haskell, OCaml, LabVIEW, and PHP interfaces. C# .NET code and CUDA extension are available. It's also included in some data mining environments: RapidMiner, PCP, and LIONsolver.
- The automatic model selection which can generate a contour of cross-validation accuracy.

2.6.4 Wave Surfer

It is an open source tool which allows users to edit the Audio file. It includes the usual actions in this kind of software: convert, amplify, normalize, echo, invert, fade in, fade out, and more. It supports many kinds of formats, like WAV, AU, AIFF, MP3, CSL, SD, Ogg/Vorbis, or NIST/Sphere.

2.7 Review of Related Works

In this section, a related literature is reviewed that discussed the Amharic question answering and Speech Interface Integration within Question Answering.

2.7.1 Text Based Amharic Question Answering

Different researches were done on question answering, the research of [9] attempts to construct an Amharic question answering for definitive questions. The researcher proposed Question Answering which deals with Amharic definition question by applying surface text pattern method. The researcher achieved nugget precision of 85.6 %, nugget recall of 73% and F-measure of 78.8%. Usage of surface patterns is effective to answer Amharic definition questions.

Seid and Mulugeta [8] studied on Amharic question answering system that identified a distinct technique used to determine the question types, the possible question focuses, and expected answer types as well as to generate proper Information Retrieval question, based on our language specific issue investigations. An approach in document retrieval focuses on retrieving three types of documents (Sentence, paragraph, and file). The algorithm has been developed for sentence/paragraph re-ranking and answer selection. The named-entity-(gazetteer) and pattern-

based answer pinpointing algorithms developed help to locate possible answer particles in a document. The Rule-Based question classification module classifies about 89% of the question correctly. The document retrieval component shows greater coverage of relevant document retrieval (97%) while the sentence based retrieval has the least (93%) which contributes to the better recall of our system. The gazetteer-based answer selection using a paragraph answer selection technique answers 72% of the questions correctly which can be considered as promising. The file based answer selection technique exhibits better recall (91%) which indicates that most relevant documents which are thought to have the correct answer are returned.

2.7.2 Speech Interface Integration within Question Answering

Various researchers have been held with reference to question answering system which is integrated with speech interface in different languages.

Until the work of Schofield and Zheng [67], no research work is done on speech based question answering system. They have been described a multimodal interface to an open domain QA system designed for rapid input of questions using a commercial dictation engine, and indicate that speech can be used for automatic QA system by their evaluation. He recommended for modifying a question-answering system for improving robustness to spoken input and spoken output.

The scholars of Mishra and Bangalore [5] stated that Mobile devices are becoming the dominant mode of information access despite being cumbersome to input text using small keyboards and browsing web pages on small screens. The research of Qme, A speech-based question-answering system that allows for spoken queries and retrieves answers to the questions instead of web pages. They presented bootstrap methods to distinguish dynamic questions from static questions and showed the benefits of tight coupling of speech recognition and retrieval components of the system.

Moreover, the researchers of Hu, Dan, Liu, and Wang [16] have developed a new and valuable research task: open domain question answering system with speech interface and first prototype (SpeechQoogle) is constructed with three separated modules: ASR, QA, and speech synthesis. There have been 600,000 QA pairs that are collected. Then corresponding acoustic model and language model is particularly developed for speech recognition module which promotes the character ACC to 87.17%. Finally, in open-set testing, the integrated prototype successfully answers 56.25% spoken questions.

Belisty [15] proposed a prototype of towards speech based Amharic question answering system for open domain factoid questions. [15] Used to Sphinx tool for speech recognition, Lucene tool for question answering and NetBeans to integrate speech recognition and question answering. The experimented result shows that the performance of continuous Amharic speech recognition developed for question corpus registered 4.5% WER using development testing and 84.93% recognition performance used live speech input data. The performance of question answering is 76% average precision in retrieving correct answers. After integrating the speech recognition and question answering, the performance registered used speech based question answering system is 75% average precision in the retrieving correct answer of a given question. The study of [15] done on speech to text limited training data, those are difficult to recognize for speaker independent. And also, used manual question classification for identifying question types.

Therefore the study attempts to designing a prototype of an automatic question classification for speech based Amharic question answering that to support visually impaired Amharic monolingual speakers. We are going to integrate ASR and TTS with Question Answering using Amharic language.

CHAPTER THREE
AUTOMATIC QUESTION CLASSIFICATION FOR SPEECH BASED AMHARIC QA

The aim of this study is to design and construct an automatic question classification for speech based Amharic question answering. This is accomplished by integrating machine learning question classification, automatic speech recognition, question answering, and speech synthesis systems. The proposed system is advancing toward such that attempts to reduce the societal gap in which the tradition document retrieval techniques do not support all users. Speech technology is the most relevant ways for human-computer communication environments.

3.1 System Design and Architecture for Integration

Figure 3.1 shows a number of modules those which are connected to each other. In the processing part of the system, the output of one module becomes an input to another module. The major three core modules are ASR, QA system, and speech synthesis. Each module has a number of components and they have their own tasks.

The first task is ASR module contains such as; Speech Capturing, Acoustic front-end, Acoustic Model, Language Model, and Speech recognizer (decoder). The ASR component is converted speech question into text and passing to QA system modules.

The second task is a question answering module which contains five major components, such as Documents pre-processing, Question processing, Searching, Ranking and Answer Selection [8]. Our study Modified independent question classifier component using automatic SVM.

The last process is speech synthesizer which that integrates contains three major sub-modules, such as Text Analysis (Tokenization, Symbol expansion (Numbers, acronyms), Homograph disambiguation), Linguistic Analysis (Pronunciation (lexicon/letter to sound rules), Prosody includes phrasing, intonation, duration) and Waveform Synthesis (Unit concatenation, prosodic modification, HMM-based speech synthesis). Amharic speech synthesis system that has been developed using a concatenated speech synthesis and unit selection method. The reason we select concatenated speech synthesis approach is to keep the naturalness of the wave [68].

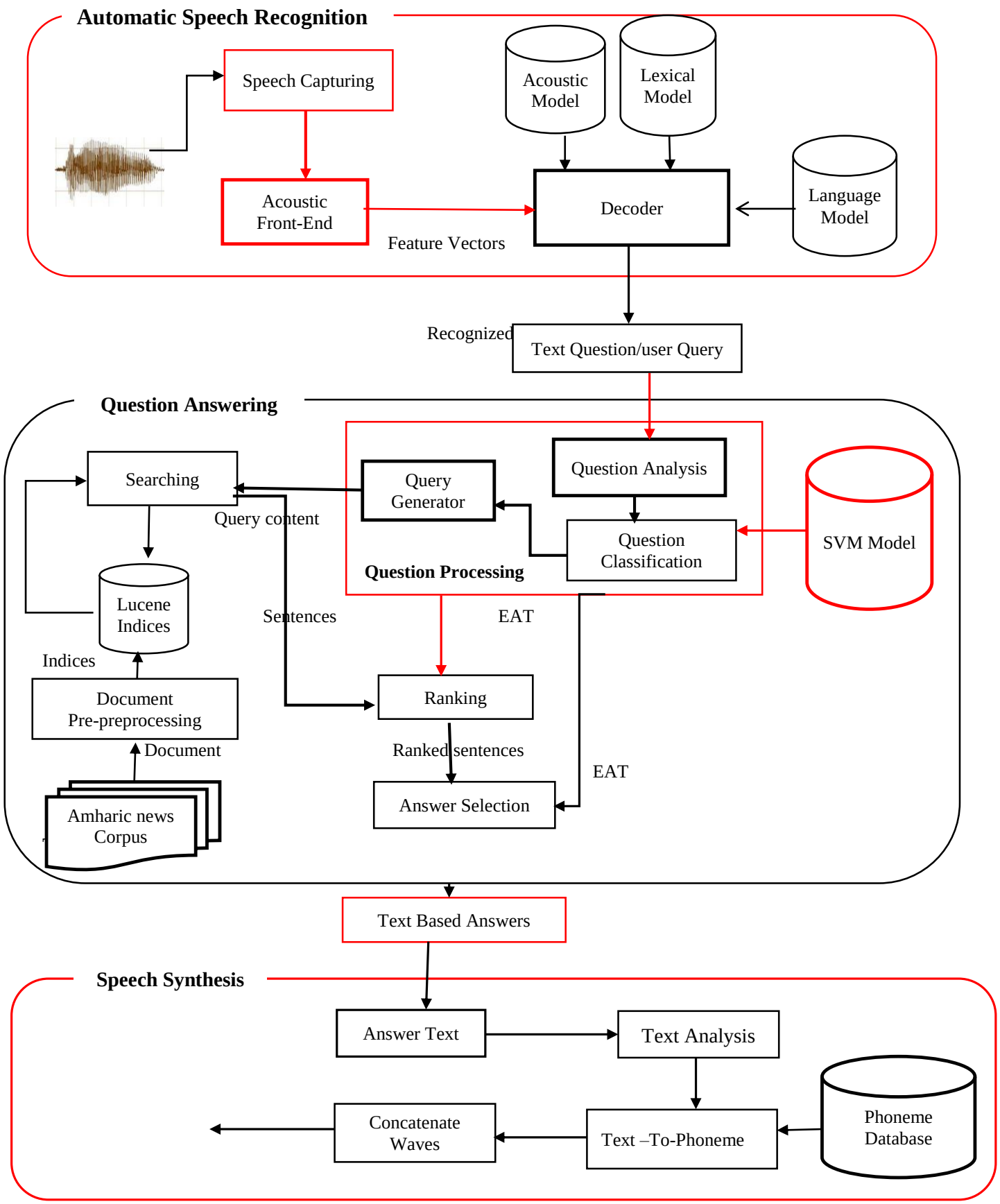


Figure 3-1 Architecture of Automatic Question Classification for Speech Based Amharic QA

The above Figure shows the architecture of a speech based Amharic question answering. In this study, we identify the required algorithm and methods to create an interface between the user and the system.

The system accepts input question from the users and processes it by using recognized that converts speech into text question. Then, the question answering system analyzes the text question and process using suitable techniques to question classification and retrieve the precise answer. Finally, the speech synthesizer, using Amharic voice reads the answer to the target users. The detail description of how the system work will be described as follows.

3.2 Amharic Speech Recognition

The ASR component takes an acoustic waveform as input and it was processed and produced an output as a sequence of texts. The ASR systems have components such as speech capturing, feature extraction, acoustic model, language model, and decoder (search). Figure 3.2 shows statistical speech recognizer components [43].

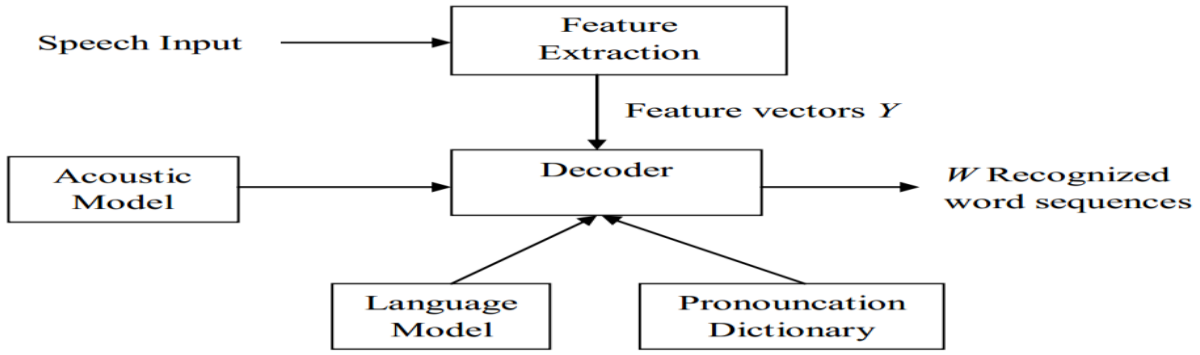


Figure 3-2 Components of ASR adopted from [44]

3.2.1 Speech Capturing

An utterance is any stream of speech between two periods of silence. Utterances are sent to the Acoustic front-end for digitizing and extracting features for further process. Silence, in speech recognition, is almost as important as what is spoken, because silence delineates the start and end of an utterance. Here's how it works. The speech recognition engine is "listening" input speech. When the engine detects speech signal, the beginning of an utterance is signaled.

3.2.2 Feature Extraction

The microphone accepts a voice question from users and translate into a speech signal. Feature Extraction is the process of converting the speech waveform to some type of parametric representation to extract features of the speech signal [43]. Feature extraction requires much attention because recognition performance relies heavily on the feature extraction phase. Different techniques for feature extraction are [38], LPC, MFCC, AMFCC, RAS, DAS, ΔMFCC, Higher lag autocorrelation coefficients, PLP, MF-PLP, BFCC, RPLP etc.

The waveform data were parameterized into sequences of feature vectors and Mel-Frequency Cepstral Coefficients (MFCCs) which are derived from Fourier Transform based log spectra was used in this study [43]. The difference from the real cepstral is that a nonlinear frequency scale is used, which approximates the behavior of the auditory system. Additionally, these coefficients are robust and reliable to variations according to speakers and recording conditions. MFCC is an audio feature extraction technique which extracts parameters from the speech similar to ones that are used by humans for hearing speech, while at the same time, deemphasizing all other information [69]. Figure 3.3 shows the way MFCC works in the representation of the speech signal.

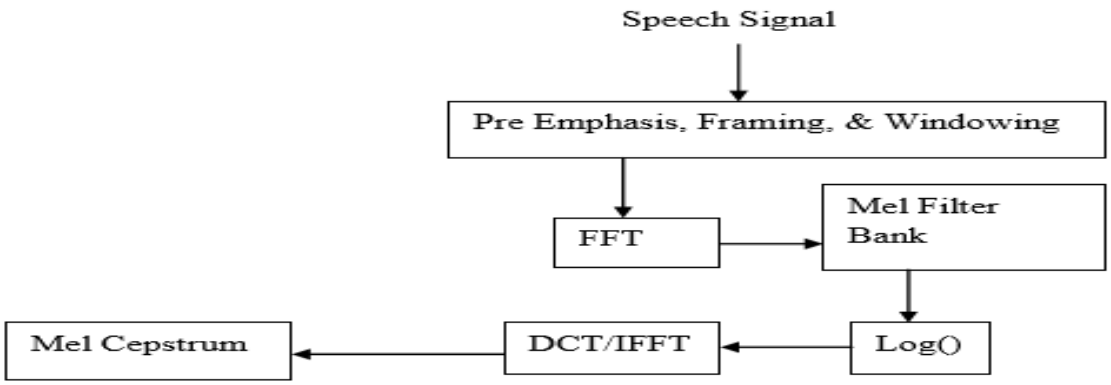


Figure 3-3 MFCC Derivation

The speech signal is first divided into time frames consisting of an arbitrary number of samples. In most systems, overlapping of the frames is used to smooth transition from frame to frame. Each time frame is then windowed with Hamming window to eliminate discontinuities at the edges [69].

Afterward, the windowing, Fast Fourier Transformation (FFT) is calculated for each frame to extract frequency components of a signal in the time-domain.

3.2.3 Acoustic Model

The acoustic model is the main component for an ASR which accounts for the most of a computational load and performance of the system. The acoustic model establishes a mapping between phonemes and their possible acoustic manifestations. It is developed for detecting the spoken phoneme. Its design involves the use of audio recordings of speech, their text scripts and compiling them into a statistical representation of sounds which make up words [38]. The acoustic model training process takes four main types of data as input, such as a set of speech recorded files, phonetic dictionary, a text file containing a parallel transcription of these speech files, and a list of phones.

Recorded Speech Files establishes question recording data are required to be in NIST or, WAV format. For building question speech, questions which are identified for the training and recording of those texts have to be prepared. For the duration of the recording period, the following parameters of the wave file have been maintained: Sampling rate of the audio: 16kHzBit rate (bits per sample): 16 (16 bit per sample will divide the element position into 65536 possible values.) Channel: mono (Single channel).

Dictionary Files establishes the word to phonetic transcription mappings. There are two types of the dictionary in training process. The first one is transcription dictionary which contains a list of words followed by phones per a single line for each word.

The second dictionary file is Filler dictionary that contains filler words. For example, the words for mentioning silence and special sounds like a cough, breathe, the chair creaking, door closing, etc. The researchers have defined non-speech utterances. In the filler file the start of the utterance silence <s>, the end of the utterance silence <\s> and the middle of the utterance silence <sll>.

Phone File: defines all the phones used in the dictionary file including the silence SIL. There should be no empty lines in this file. E.g. ʋ, ሃ, ሁ, ፕ ……., ፖ, ቨ, SIL.

File IDs Definition file: in the training and testing files, all audio file ids without extension with references to the root folder defined. For examples:

Wav/s1_test/sp11_file1
Wav/s1_test/sp11_file2
Wav/s1_test/sp11_file3

Transcription File: establishes sentence to utterance mappings in each file. Each line starts with <s> and ends with </s> followed by speech fields (filed file) in parentheses. These are not phone to audio mappings but mapping between the words in the left column of the dictionary file and the audio files. The files should be in the same order as described in the training file ids file.

3.2.4 Lexical Models

Lexical Models provide the pronunciation of each word in a given language. Lexicon is developed by various combinations of phones which are defined through lexicon model to give valid words for the recognition [38].

3.2.5 Language Model

The language model is the key element functioned on millions of words, consisting of millions of parameters and with the help of pronunciation dictionary, developed word sequences in a sentence. An ASR system uses n-gram language models which are used to search for correct word sequence by predicting the likelihood of the nth word on the basis of the n−1 preceding words. According to [38] the probability of occurrence of a word sequence W is calculated as:

P (W) =P (w1, w2, ..., wm-1, wm)

=P(w1).P(w2|w1). P(w3|w1w2)..... P(wm |w1w2w3.....w m-1).....(1)

3.2.6 Decoder

In the part of developing the speech recognition system, we have used of Amharic words and speech capturing. There are different ASR tools such as; HTK, Sphinx and etc. The HTK tools don't support direct Amharic Unicode scripts. HTK is worked or selected Amharic words, then

transliterate to Amharic and Latin. But sphinx tools support Amharic Unicode without transliterating the language so due to this reason we used, sphinx-4 for decoding.

Afterward, we are follow-up using the above-required data to training was prepared using the three sphinx tools, such as sphinx3, sphinxbase and sphinxtrain were put in the same folder. After all the system setup and data preparation are done, the next step is to train the system. Initial configured the training setting and to start the training and testing, the following command was used in command prompt and it went through all the required stages.

Python.../sphinxtrain/scripts/sphinxtrain run

After training, showed the seven files, are converted into a jar file. To implement in NetBeans and sphinx-4 used decoding. We have copied all the training files and model files to a folder called WSJ_8gau_13dCep_16k_40mel_130Hz_6800Hz and also dict folder which contains the dictionary (bekeledb.dic) file. Then, the following command was used for converting files to the jar file.

JAR cf WSJ_8gau_13dCep_16k_40mel_130Hz_6800Hz.jar / WSJ_8gau_13dCep_16k_40mel_130Hz_6800Hz

The Decoder used sphinx-4 tool because it has been written in java programming language and simple to import the .jar training file on the NetBeans environment.

3.3 Question answering Systems

QA is concerned with a building system that automatically answers questions arranged by speech recognition. This study uses a syntactic analysis for processing a question and retrieving the expected answer. In QA system, different tasks such as question analysis, question classification, the question generated, searching, indexing, answers extraction, and answers retrieve.

3.3.1 Amharic Natural Language Analysis

The Amharic text analyzer is the principal section in indexing Amharic documents using Lucene. The following are the tasks performed in the Amharic analyzer in their order of use;

Tokenizing

The Amharic documents in the repository are tokenized before going to the indexing process. Tokenization breakdowns the stream of characters into raw terms or tokens. This process detects word boundaries of a written text. In Amharic, words can be separated using a whitespace, tab, Amharic punctuation marks, or by a new line. Therefore, using tokenization, we identify separate terms from the document. In the process of tokenization, alternative Amharic characters found in the current token normalized to a unique character. The known Amharic alternative alphabets are then replaced by a normal one throughout the documents in the repository. The normalization is similar to the one in [8]. Example: characters "ሀ", "ኀ", and "ሐ", are normalized into "ሀ". "ሰ" & "ሠ" are normalized into "ሰ".

Stop words Removal

Stop words commonly occur in a document. As stop words have no document discrimination value for information retrieval purpose, they have to be removed. A new list is also added to include question particles like ማን (who), መቼ (when), የት (where), and ስንት (how much) so that they are not removed as a stop word.

Stemming

Stemming is a technique whereby morphological variants are reduced to a single stem. It is language dependent and should be tailored for each language since languages have a varying degree of differences in their morphological properties. As Amharic is a morphologically rich language, designing a perfect stemmer is not an easy task. So far, different efforts have been made to come up with a full-fledged stemmer for the Amharic language. For the question answering task, we should keep proper nouns, dates, and numerals un-stemmed because these are the terms with a potential to bear an answer. So, for this purpose, the stemming algorithm developed in [8].

We have used the stemming technique for identifying stem/ root of a word. The technique first identifies the prefix list and the suffix list from a word, through the identification process of the prefix and the suffix word, stem/root is identified. For example, if we have a word "የኢንጅነሮች" the stemming technique identifies "ኢንጅነር" as stem/root of a word. The prefix word is "የ" and the

suffix word is “ኦች”. Furthermore, the stemming technique is used for improving the effectiveness of IR and text mining and reducing indexing size.

3.3.2 Question Processing

In the process of question processing, four steps are followed to build a sub-component, such as question analysis, question classification, question reformulation and question generation. Amharic natural language analysis has used Amharic tokenizing, stop words removal, and stemming.

3.3.2.1 Question Classification

In order to define the question and identify the expected answer, language analysis should be developed and classifies a question based on training data. To determine answer types (person, time, place, and number), the previous researchers [8] [15] use rule-based question classification. However, this study uses automatic question classification.

The first step we took to determine the expected answer were understanding the question. The type of queries is a person, number value, time or place. Then, we designed decision Question Model based on the answer type.

The decision model is constructed from question classification using the SVM (Support Vector Machines) algorithm with a dataset of 80 questions. The aim of this classical model is to return the answer type of a question. For instance, if the system gets the question “ግንቦት 8 2001 በተካሄደው የጠቅላላ ጉባዔ ከፊፋ የመጡት ተወካዮች ስንት ናቸው” and the expected answer type of this question is “ቁጥር [numerical]” which goes to the class “number”. Therefore, the system would search for a “ቁጥር [numerical]” as an answer to the given question. The question classifier and decision model generation are ensured by two components. The first component is the question classifier which pre-processes the question dataset and trains it by the SVM algorithm. And the second component is the decision model generator which establishes the models that we use in the proposed question.

An important module of any Question Answering system is question classifier. The question classification significance is worked based on the expected answer types. The significance of the

answer type; decreases processing and effort and also offers a possible way to select correct answers from among the possible answer candidates.

In the literature review, many methods and methodologies have been used in question classification. Based on the literature review, SVM model for question classifier is constructed. However, machine learning approaches (especially SVM) provided to get to the closest answer type of a question.

Table 3-1 Coarse-grained and Fine classes

Coarse classes	**Fine classes**
Person, place (ስም)	ቦታ [አገር(ከተማ, ክ/ከተማ, ወረዳ, ቀበሌ, ወንዝ,ተራራ, ሀይቅ, ውቅያኖስ, ሆቴል,ክልል, ዩኒቨርሲቲ, ዋሻ, ሐውልት, ድልድይ, ደሴት, ት/ቤት, መስጅድ, ቤተ-ክርስቲያን, ሌላ), ፕላኔት, ድርጅት], ከለር/መልክ, ድረ-ገፅ, ሀይማኖት, ቋንቋ, ሥራ/ሙያ, ሰው [ባለሌጣን፣ፕረዝዳንት, ተጨዋች, ደራሲ, ሀላፊ, ቅጥያ ስም, ከንቲባ/አስተዳዳሪ, ሌላ], በሽታ, መጠጥ, ምግብ, እንስሳት, እቃ [መኪና, አውሮፕላን, ሌላ], ሌላ
Person, place ስፖርት)	ተጨዋች, አትሌት, ስታድየም, ቡድን, አሰልጣኝ, እግር ኳስ አምበል
Person [ሙዚቃ]	መሣሪያ, ስም, ዘፋኝ, አርቲስት, ደራሲ,ፊልም
Number, time (መጠን)	ጊዜ,ርቀት,ገንዘብ,ኮምፒዩተር,ፍጥነት,ሙቀት,አመት, ቁጥር፣ ሌላ

For the highest accuracy/precision, we use the SVM algorithm to train the system question classification. And also, we have developed the coarse-grained classes and the fine-grained classes of answers types. We used four coarse-grained classes and sixty-five fine-grained classes. All classes are listed in the above Table 3.1.

The Question classification predicts the expected class based on the training data. For instance, the following are examples of Amharic Factoid question answering system (see appendix III). The first question is የኢትዮጵያ እግር ኳስ ፌዴሬሽን ፕሬዝዳንት ማን ይባላል Root Class: a person (ሰም) [ሰው]; Fine Class: president (ፕሬዝዳንት), the second examples are የኢትዮጵያ የአየር ትራፊክ ተቆጣጣሪዎች መሀበር ስንተኛ አመቱን አከበረ? Root Class: number **(መጠን)** Fine Class: year (አመት). LibSVM classifier tool to identify a question type and generate a decision model based on the training data.

As we discussed above, the system finds the question from a given question. However, knowing the question type only is not enough to find the appropriate answer. For example, the questions can be relatively unclear (ambiguous) for the classifier in terms of the required answer. Therefore, to discuss using that kind of ambiguity and reduce the search non-stop words based on **keywords**, further explanation techniques that are used which extract the question focus is crucial. The question focus in our situation is represented by **Keywords**. From this time, we build keywords which are extracted from the question.

To support the above statement let us see this example, the question "ጠቅላይ ሚኒስትር መሆን የሚቻለው ከስንት አመት በላይ ነው?"Root Class: number (መጠን); Fine Class: year (አመት)**.** This information is not sufficient, therefore, we need to find the focus of the question which is "መሆን የሚቻለው አመት በላይ" and "ጠቅላይ ሚኒስትር" as the keyword.

Keywords Extraction

Keyword extraction involves Amharic text tokenizing, part of speech tagging, chunking, stop word removal and resources extraction for a question.

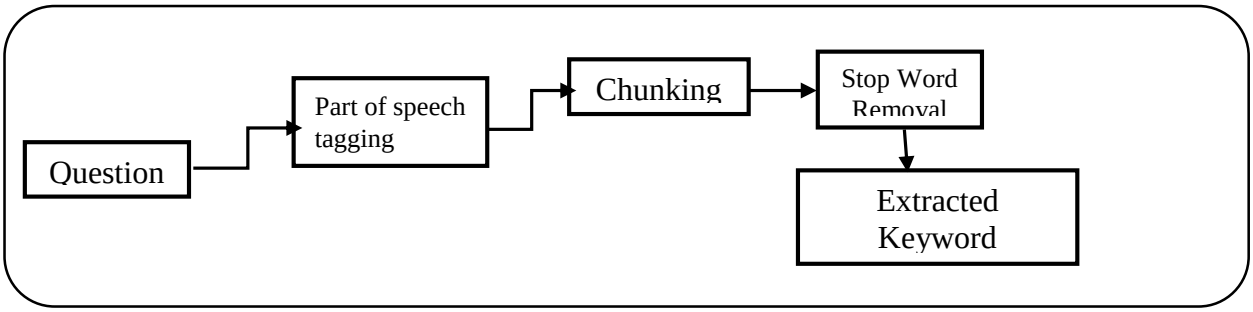


Figure 3-4 Keyword content extraction

In the extraction of keywords from the question, the system chooses any word in the question that satisfies one of the following seven conditions [70]:

1. All non-stop words in quotations.
2. Words recognized as proper nouns.
3. Complex nominal and its adjective modifiers.
4. All another complex nominal
5. All nouns and their adjectival modifiers
6. All other nouns
7. All verbs.

3.3.3 Training Data for Question Classification

To fill the gap of rule-based question classification, we used machine learning algorithm. Moreover, LibSVM linear library is used because [66] it is faster and more accurate classifier. The first procedure is preparing data in the training.

Training the classifier SVM, initially designed for binary classification (classification into one of the two classes). Different approaches have been proposed to use the binary classifier for multi-class classification tasks. The Windows version of the command SVM classifier, LibSVM, were developed by [66], and those are used for our question classification task. It is freely available for research purpose and can be smoothly integrated with our system. The more training data prepared for the classifier, the more accurate the classifier becomes.

SVM, as a statistical approach, needs a large amount of training data so as to train the machine it is preferable to make accurate classifications for the unseen test data. For this, we prepared 82 questions from the two classes of factoid questions ("person", and "number") for training the classifier for further clarification see appendix V. LibSVM data format is not supported non-numerical data's, so we have changed a question text form to numeric within corresponding training vocabulary. The general step we took to classify the question is, listing the words of the training question and giving them index number for more see appendix IV. Then we continue to classify the question in the format of the LibSVM which is stated below.

<Classes> <index1: label1> <index2:label2>…….

We assign class 0 for the 35 question which is person type and class 1 for the 47 question which is a numeric type. For examples, the expected answer type of "የኢትዮጵያ እግር ኳስ ፌዴሬሽን ፕሬዝዳንት ማን ይባላል" are a person. Therefore, the training data's are look like; 0 1:1 2:1 3:1 4:1 5:1 6:1 7:1. Zero (0) indicates person class and 1:1 indicates index and label of the term "የኢትዮጵያ" etc. The finally, result in the training model, libSVM generated statically data 0. That means a question is a person. If the question is numeric, 1 is generated as the LibSVM model.

3.3.4 Searching

The generated question is then passed to the document retrieval component for searching documents related to the question generated and believed to contain candidate answers. QA system usually wants documents to be retrieved only when all existence keywords are present in the document. Filtering a paragraph of the highest question keyword matching to document and paragraph ordering based on the three criteria, same word sequence score, distance score and missing keyword score.

3.3.5 Indexing

Indexing is defined as the processing of the original data into a highly efficient cross-reference lookup (index) in order to facilitate searching [71]. To search a large amount of data rapidly, it requires index terms that can represent each document in an index in a suitable format. Lucene is an open source software used to index term from large data. Lucene uses an indexing mechanism called inverted index. The purpose of an inverted index is to allow a fast full-text search, at a cost of increased processing when a document is added to the database. In this study, file, paragraph, sentences based indexing level were used. So as to do the indexing, the Lucene API uses five basic classes: Index Writer, Directory, Analyzer, Document, and Field [71].

To index term using Lucene, first, we add Document(s) containing Field(s) to IndexWriter which analyzes the Document(s) using the Analyzer and then creates/open/edit indexes as required and store/update them in a Directory. IndexWriter is used to update or create indexes. It is not used to read indexes.

3.3.6 Answer Selection

After retrieving documented index, it will reduce paragraph filtering and paragraph ordering by using distance vector. The answer extraction method is based on hybrid approach pattern matching and gazetteers from the most simple and commonly used group of answer extraction methods. Retrieving top ranked answer is based on counts the number of expected answer type in a sentence and the number of terms matches a question did with the document terms. Answer extractions extract relevant word based on measuring a distance between keywords.

3.4 Speech synthesis

As we illustrated on section 3.3, the questions were processed and retrieve expected answer using the Amharic factoid question answering system. Text based answers can be produced in the form of audio. As shown in figure 3.1, various input speech units are recorded in the Amharic language, and the quality of the synthetic speech produced when using the speech units are presented. The speech units of Amharic language are taken into phonemes. However, diphones, syllables, Words, phrases, and sentences are not taken because there are a lot of such units in the Amharic language. From this time, storing numerous unit's needs a vast amount of storage space.

The basic goal of speech synthesis is to produce natural sounding and intelligible speech for any input. And also, the most challenging things of speech synthesis are naturalness, intelligibility, cost-effectiveness, and expressivity. The previous researcher [72] used Rule-Based and corpus-based methods. This research used concatenation synthesis because it solves the above problems [55]. Concatenation of stored speech units is common of speech synthesizer that is in use for any applications [73].

Therefore, selects the phonemes as a unit for concatenation, especially in an environment where resource (processing and storage capacity) is limited, there come issues like how much of the unit are there in the language (in this case Amharic language), how much storage space is needed to store the units, and also how CPU intensive the synthesis process (generation of the waveform). All these issues should be considered keeping in mind the quality of the spoken word which is the design goal of a speech synthesizer and produced after the synthesis process has taken place.

3.4.1 Preparation of Phoneme Wave File

As it is previously discussed, the main goal of speech synthesis is producing natural sounding and intelligible speech output. An Amharic phoneme speech has two types, context-dependent, and context-independent phonemes. As stated in section 2.5.2, there are 233 Amharic phonemes. In speech database building, a Recording for all Amharic phoneme is done using Wave surf speech synthesis analysis tool at DBU (Debre Berhan University) office environment See appendix VI.

The major reason for the selection of phonemes as basic speech unit is that phoneme is smaller than word and syllables. So, phoneme gives less number of speech sound as compared to words and syllables, thus requiring less storage space.

The context independent phoneme directly recorded Amharic phoneme and saved with the name of Amharic phoneme stored in the same folder. In context, a dependent phoneme is recorded both Amharic phonetic list and IPA equivalence and its ASCII Transliteration equivalence See appendix II. In the recording phase, designing the input wave parameters add and remove wave files based bit size, sample rate, channels, data headers and the file of phoneme wave files.

3.4.2 Input Wave Parameters

The creation of a wave audio file includes bit size, sample rate, channels, data, headers and the file. A WAV (RIFF) file is a multi-format file that contains a header and data parameters. A WAV file contains a header and the raw data, in time format. The wave file have different factors to record a phoneme (size, sample rate, channels, data, header, etc.)

Bit size determines how much information can be stored in a file. Most of the bit size has 16 bit and 8-bit files are smaller but have less resolution. Bit size deals with amplitude. In 8 bit recordings, a total of 256 (0 to 255) amplitude levels are available. In 16 bit, a total of 65,536 (-32768 to 32767) amplitude level is available. The greater the resolution of the file is, the greater the realistic dynamic range of the file. The recording Audio files have used 16-bit samples.

The sample rate is the number of samples per second. The recording Audio files have a sample rate of 44,100. This means that 1 second of audio has 44,100 samples. When seeing at frequency response, the highest frequency can be measured to be 1/2 of the sample rate.

Channels are the number of separate recording elements in the data. For a real quick example, one channel is mono and two channels are stereo. In this document, both single and dual channel recordings were used.

The header is the opening of a WAV file is used to provide specifications on the file type, sample rate, sample size and bit size of the file, as well as its overall length. Examples of one wave files include the following header format:

Table 3-2 Wave Input Parameters

Positions	Sample Value	Description
1 - 4	"RIFF"	Marks the file as ariff file. Characters are each 1 byte long.
5 - 8	File size (integer)	The size of the overall file - 8 bytes, in bytes (32-bit integer). Typically, you'd fill this in after creation.
9 -12	"WAVE"	File Type Header. For our purposes, it always equals "WAVE".
13-16	"fmt "	Format chunk marker. Includes trailing null
17-20	16	Length of format data as listed above
21-22	1	Type of format (1 is PCM) - 2 byte integer
23-24	2	Number of Channels - 2 byte integer
25-28	44100	Sample Rate - 32 byte integer. Common values are 44100 (CD), 48000 (DAT). Sample Rate = Number of Samples per second, or Hertz.
29-32	176400	(Sample Rate * BitsPerSample * Channels) / 8.
33-34	4	(BitsPerSample * Channels) / 8.1 - 8 bit mono2 - 8 bit stereo/16 bit mono4 - 16 bit stereo
35-36	16	Bits per sample
37-40	"data"	"Data" chunk header. Marks the beginning of the data section.
41-44	File size (data)	Size of the data section.
Sample values are given above for a 16-bit stereo source.		

In the intended study, recorded and used as the researcher voice speaker of Amharic recorded that the selected phoneme. The quality of audio depends upon the quality of recorded sound and also the quality of extracted audio units from this recorded audio. The recording was done in the WaveSurfer with the following characteristic: Sample rate: 44,100Hz, Bit Depth: 16 bit, Channels: Mono.

3.4.3 Text –to- Phoneme

In our study, we used unit selection synthesis in which the prerecorded words are stored with corresponding names and preserved in a database. Based on the input text from the user interface, the phonemes wave files are selected from the database and concatenated audio wave files using

java code to create the resulting output. Compare text input phoneme within recorded phoneme and select the required wave files.

3.3.3 Concatenate Wave Files

The input text of any length is segmented into Amharic phonemes by using substring java code. After phonemes are searched in the prepared database and corresponding instant's time location is retrieved. After, these second's location search in audio wave file is stored in the resources folder. Finally, the concatenate word wave files produce in the form of wave format.

3.5 Integration of speech recognition and Speech Synthesis for QA

Speech based Amharic question answering offers the important potential for finding information on a laptop and mobile phone. Figure 3.5 shows that the code snapshot that instantiates the recognizer to accept speech from the user using a microphone and then tries to recognize as per the configuration and grammars defined.

Integration module of the existing system is used to integrate the Amharic ASR with question answering. It takes recognized utterances from the Amharic speech recognition as input. The input needs further processing due to the speech recognition performance that results in insertion, substitution and deletion errors. Inserted words removed, deleted words are replaced and substituted words are mapped using a dictionary. Correct questions are automatically passed to the question answering module using QAnswering().

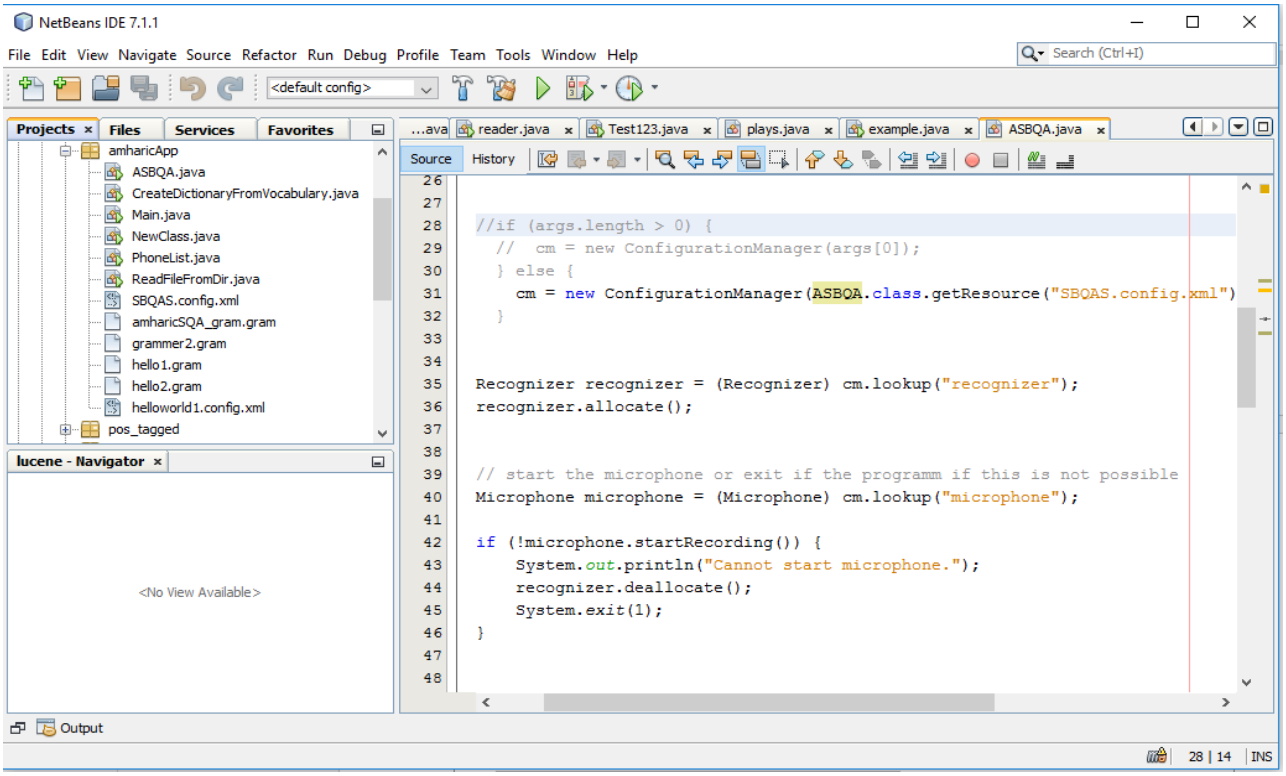


Figure 3-5 Speech recognition integrated with QA system

A QAnswering () function received input text question from ASR. The input texts are searched the right answer with further processing due to Amharic Natural language analysis, SVM classifier and question generated using both IE and IR techniques. QAnswering () function retrieved list of answers with their source document.

The codes on figure 3.6 show how a retrieved answer automatically passed to the speech synthesis module using ASynthesis () function. ASynthesis () function produces top five answers in the form of an audio.

```
        public ArrayList<String> list = new ArrayList<String>();
418  public void QAnswering (String question) {
419      ASynthesis As=new ASynthesis();
420
421            jTextArea1.setText(question);
422             jTextArea2.setText("");
423             jTextArea3.setText("");
424             TAmarachmelis1.setText("");
425             TAmarachmelis2.setText("");
426             TAmarachmelis3.setText("");
427             TAmarachmelis4.setText("");
428              search(indexDir, q);
429             list.add(jTextArea2.getText());
430             list.add(TAmarachmelis1.getText());
431             list.add(TAmarachmelis2.getText());
432             list.add(TAmarachmelis3.getText());
433             list.add(TAmarachmelis4.getText());
434
435              for (int i=1; i<5;i++)
436              {
437        ArrayList<String> x=     As.ASynthesis(ArrayList<String> list.get(i));
438        System.out.println(x.get(i));
439              }
```

Figure 3-6 QA system integrated with Speech synthesis

3.6 Performance Evaluation

Evaluation measures that are used for measuring the performance of the system are targeted to the aims of the research. The concept of evaluation for question answering requires the creation of questions and correct answers for each question, given a corpus and some pre-defined criteria for judging correctness in the question answering part.

The Amharic speech recognition, the Question Answering system, and the speech synthesis are evaluated independently. The Amharic speech recognition evaluated based on the rate of word errors (WER), which is computed using the following formula [74].

$$\text{WER (Word Error Rate)} = \frac{\text{total Error}}{total\ words\ uttered}$$

$$\text{Accuracy} = \frac{total\ correcly\ recognized}{total\ words\ uttered}$$

The Question Answering performance is evaluated using precision and recall. Precision is calculated as the number of correctly answered documents over the total list of answers (correct (correct and not exact answer), wrong, and No Answer). The recall is also calculated as a number of correctly answered questions among the list of expected answer sets where documents are first analyzed for the presence of correct answers.

$$\text{Precision} = \frac{\text{Correct answers}}{Total\ answer\ retrieved}$$

$$\text{recall} = \frac{Correct\ answers}{correct\ answer + missed\ answer}$$

The most evaluation measured in question answering is F-Measure which is a utility class for evaluators which measure precision, recall, and the resulting f-measure. Evaluation results are the arithmetic mean of the precision scores calculated for each reference sample and the arithmetic mean of the recall scores calculated for each reference sample [26].

$$F - measure = 2 * \frac{(Recall * Percision)}{recall + percision}$$

The performance of the speech synthesis system is evaluated in two steps. First, the accuracy of the system is tested to observe its prediction capability of a given text. Then, the overall quality or acceptability of synthetic speech can be measured. Literature suggests three techniques: Mean Opinion Score (MOS), Modified Rhyme Test (MRT), and Diagnostic Rhyme Test (MRT) [75]. Intelligibility and naturalness are factors which measure the overall quality of synthetic speech. The synthesized Amharic speech is tested based on its intelligibility and naturalness. Among the identified testing systems, MOS is preferred in this study to test the prototype TTS for the Amharic language because it is the most widely used and the modest method to evaluate the overall speech quality [75]. MOS is a five-level scale: bad (1), poor (2), good (3), very good (4), and excellent (5). For the evaluation of the system using MOS, ten native listeners of the language are asked to give their opinion on the naturalness and intelligibility of the synthesized sentences with respect to scales used in MOS.

Finally, the overall performance of automatic question classification for speech based Amharic question answering is measured based on user acceptance testing of the systems based on criteria.

CHAPTER FOUR
IMPLEMENTATION OF SPEECH BASED AMHARIC QUESTION ANSWERING

In this chapter, construct a prototype of automatic question classification for speech based Amharic question answering. The system recognized a question and convert it into text form and pass into the question-answering module. Question answering module classifier the question type and predict the expected answer. Then, the system received top-ranked answers in the form of text. To achieve the objectivity of the study, we used different models as well as software tools. In the previous chapter, we covered the architecture and basic explanations about the components of the system. This chapter basically deals about the algorithm which we use to develop the system.

Following, we discussed the setting up system environment and a detail view of the sub-parts in the system and discuss how these parts are formulated with the following order; speech recognition, question answering process (indexer, question analysis, question classification, question reformulation and question generated, document retrieval, answer extraction) and finally, speech synthesis (text analysis, phonetic analysis, unit selector, concatenate wave and audio wave out).

4.1 Development Environment

This study is written in java programming language. The implementation and experimentation of the prototype have been taken place on a Toshiba Laptop with the windows-7 32-bit platform in a core i5laptop with CPU speed 2.5GHz, 6GB RAM capacity. There are different softwares;

Netbeans7.1.1: is used to write the java codes and building GUI for the whole system.
Lucene: an open source tool that is used for a searching and indexing.
SphinxTrain1.0.8: is a set of tools used for acoustic modeling.
Sphinxbase0.8: is a common set of the library used by several projects in CMU Sphinx.
Sphinx4: adjustable, modifiable recognizer written in Java.
Sphinx 3 trainer: - to build the Amharic acoustic model.
Active Perl: to run Perl scripts while the training was performed.
Active Python: to run python scripts during the training session.
Cygwin: is free software that provides a Unix-like environment and software tool set to users of

any modern x86 32-bit and 64-bit versions of MS-Windows. We have used it to setup the configuration file and run Sphinx-Train to start the training.

Visual studio 10: to compile Sphinxbase0.8, SphinxTrain1.0.8.

Wavesufer 1.8.5: is a really good open source tool which allows you to edit as if you were a professional any audio file.

Libsvm*:* used question classifier and it supports multi-class classification.

4.2 Prototype system Components

Figure 4.1 shows three major components of the sentence level speech-based factoid question answering system retrieval. The prototype integrates speech recognition, question analysis using natural language processing, machine learning for classification, information retrieval techniques for answer selection and speech synthesis for produce an audio waveform.

Pseudo code

1. Start
2. User asking a question
3. Speech recognition recognize a user question
4. Subsequently, recognize text question call Question Answering function
5. Question answering system processing
6. Selecting for answer of a specific question
7. Synthesizer can produce answer in voice and back to Step two
8. End

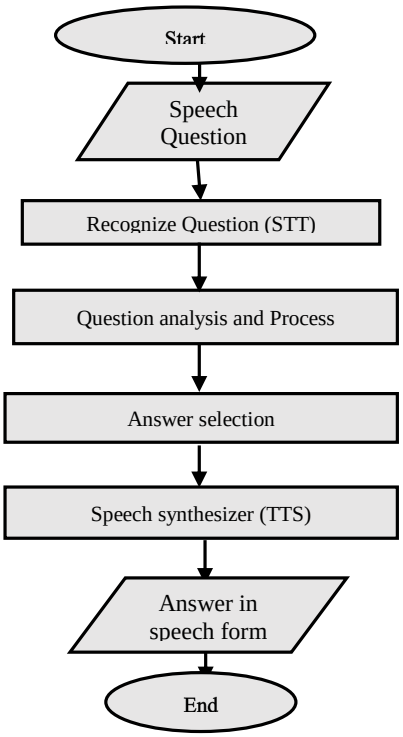


Figure 4-1 system flow chart

4.2.1 **Recognize Question**

The first step to constructing a speech input interface requires initialization. It took steps to make the recognizer to recognize the speech uttered by the system user. The microphone accepts speech and convert into a speech signal. The extract features of the speech signal and waveform data were parameterized into sequences of feature vectors. Based on the probability and statistics which depends on language, dictionary, and lexical model the decoder decodes the utterances and converted it into texts.

```
23
24      public static void main(String[] args) {
25          ConfigurationManager cm;
26              cm = new ConfigurationManager(ASBQA.class.getResource("ABQAS.config.xml"));
27               Recognizer recognizer = (Recognizer) cm.lookup("recognizer");
28          recognizer.allocate();
29          Microphone microphone = (Microphone) cm.lookup("microphone");
30          if (!microphone.startRecording()) {
31              System.out.println("Cannot start microphone.");
32              recognizer.deallocate();
33              System.exit(1);
34          }
35          System.out.println("Say: With Amharic Language");
36          while (true) {
37              System.out.println("Start speaking. Press Ctrl-C to quit.\n");
38              Result result = recognizer.recognize();
39              if (result != null) {
40                  String resultText = result.getBestFinalResultNoFiller();
41                      System.out.println("You said: " + resultText + '\n');
42                     speechq=resultText;
43                     QAS(speechq);
44              } else {
45                  System.out.println("I can't hear what you said.\n");
46              }
47          }
48      }
```

Figure 4-2 Recognized for a Voice question

The codes on figure 4.2 show how the user asked a speech question, and how the system recognized the question based on the model constructed during the training phase and converts the question into texts. The first steps are Configuration which is used to supply required and optional attributes to recognize using ABQAS.config.XML. The XML files included those parameters, such as Sphinx-4, frequently tuned properties, word recognizer, Decoder, linguist, Grammar,

Dictionary, acoustic model, unit manager, front end, live frontend and front-end pipelines are configured for the understanding of a user voice question.

The system accepts user's specific question using a microphone. Then, the system recognized the voice question and converted into the text-based question. After, recognizing a complete text question, the text question passes to question answering systems using QAS (speechq) function.

The most importance is to recognize question and converted the speech to text. As the search engine only accepts input text, the user's speech based question is converted into text. The system works well when there is no interruption such as noise. If the system won't recognize the word due to some interruption, based on the utterance the system automatically replaces, add and modify words which fit the unrecognized word.

4.2.2 Question Analysis and Processing

Figure 4.3 shows how the QA system receives the text question from the speech recognition component. QAnswering (String question) before generated answer, processed a question normalized (), analyzequestion(), generated SVM() model to identify answer based on question type, DocumentNormalization() and finally extract and retrieve the expected answers. The importance of the code is extracting answers from the corpus.

```
121  public void QAnswering (String question) {
122                jTextArea1.setText(question);
123              jTextArea2.setText("");
124              jTextArea3.setText("");
125              TAmarachmelis1.setText("");
126              TAmarachmelis2.setText("");
127              TAmarachmelis3.setText("");
128              TAmarachmelis4.setText("");
129              File indexDir = new File("SEIndexes");
130             QueryGenerator qgen = new QueryGenerator();
131              AnalyzeQuestion an = new AnalyzeQuestion();
132              SVMmodel sv=new  SVMmodel();
133              DocumentNormalization dn = new DocumentNormalization();
134              if (jTextArea1.getText().toString().equals("")) {
135                  JOptionPane.showMessageDialog(null, "Please Enter a Question!!!.");
136              } else {
137                  questiontype = an.SVM(dn.(jTextArea1.getText().toString()));
138                  if (questiontype == null) {
139                      JOptionPane.showMessageDialog(null, "Not Factoid Question!!!.");
140                      jTextArea1.setText("");
141                  } else {
142                  String q= qgen.QueryExpand((jTextArea1.getText().toString())); //
143                                          Queryword = q;
144                      try { search(indexDir, q);
145                           System.out.println(Queryword);
146                      }   }  }
147
```

Figure 4-3 question answering process and answer extraction

In question answering system, the system has used information extraction and information retrieval which it is to get top-ranked answers and appeared on the user interface. The system retrieved one accurate answer, four alternative answers, and the resources of the document See figure 4.4.

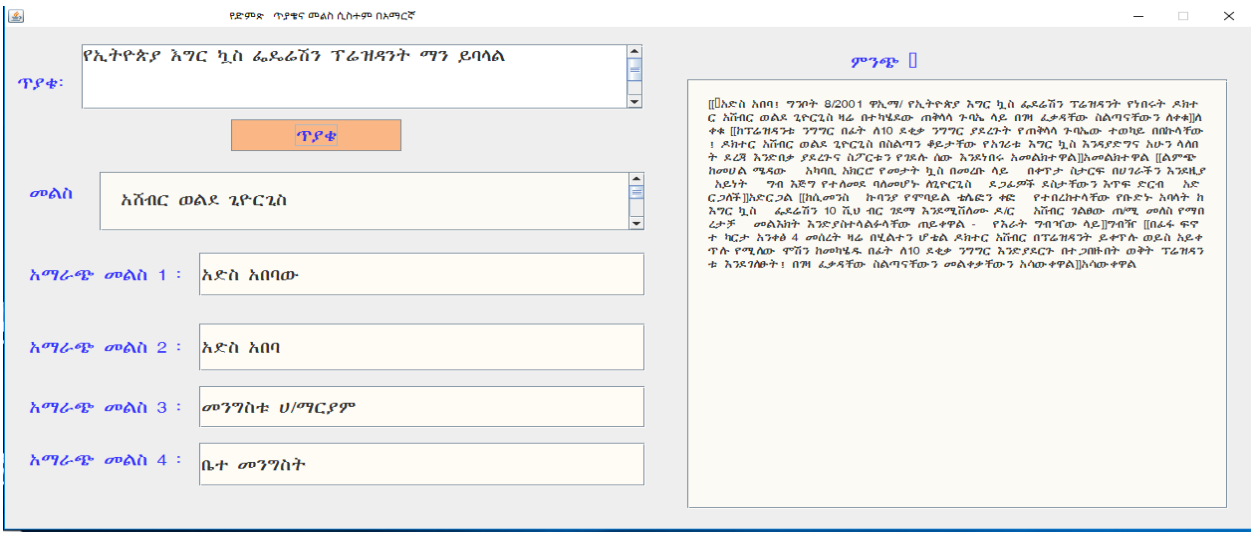


Figure 4-4 QA System User Interface

Figure 4-4 shows the user interface of the system; the first text field accepts a question from the user in the form of voice, the second text field shows the exact answer for the question and the four text fields fetches alternative answers for the question.

The QA system generates proper answers for a specific question. Figure 3.6 shows that the QA system is passed to speech synthesis module.

4.2.3 Speech synthesizer

For the development of Amharic speech phoneme database, an Amharic language containing the usable phonemes with the combination of vowels (V) and consonant-vowel (CV) and their equivalent audio positions are stored in the database file.

The moment's initial and ending position for V and CV are marked from the recorded wave file. The input text of any length is segmented based on Amharic phonemes, later phonemes are searched in the prepared database and the corresponding position is retrieved. Next, it will concatenate these position based on each input header and the sound wave file and stored it into the resources folder. Then, finally, produce the wave files.

4.2.3.1 Text Analysis

The text analyzer accepts text data from the user interface and splitters words into character based on Amharic dictionary. The importance of the code is to identify Amharic character and corresponding phoneme wave file search from the database.

```
public ArrayList<String>  ASynthesis(ArrayList<String> txt) throws FileNotFoundException, IOException{
        ArrayList<String> re = txt;
                    for (int i = 0; i < re.lastIndexOf(i); i++)
                    {
                        String a = re.substring(i, i);
                         for(String prime : list) // Loop through List with foreach
                        {
                            if (prime == a)
                            {  single_letter.add(prime);
                                break;
                            }
                            else
                            {  continue;
                            }
                        }
                        }
                    for(String sample : single_letter)
                     {
                      //(Iterator<String> sample = list.iterator(); sample.hasNext(); ) {
                        //  String item = sample.next();
                         String k = sample;
                        String sa = k + exten;
                        String red = sMediaPath + sa;
                        result.add(red);
                     }
                  WaveIO wa=new WaveIO();
                   wa.Merge(result, "F:\\"+txt+".wav");
```

Figure 4-5 snapshots of Convert Text to Phoneme

Figure 4.5 shows how Synthesis (ArrayList<String> text) function accept input text which is passed from the question answering using ASynthesis() function. The texts then convert to phoneme using substring (i, i+1) java method, and it will add into array list and iterate until finished. Example that shows, splitting words into phonemes are illustrated as follow;

አሸብር ወልደ ጊዮርጊስ ------> አ ሸ ብ ር ወ ል ደ ጊ ዮ ር ጊ ስ

4.2.3.2 Text to Phonemes

The database required for a character to phoneme wave file is recorded alphabets (ሀ-ፐ), digits (0-9) in the form of wave files (.wav). For each and every phoneme, it will search corresponding wave (.wav) files from the database. After finding each phoneme wave, all phoneme wave files are stored in array list “result” and calling the merging function using wa.Merge(result, "F:\\"+txt+".wav").

Examples አ ሸ ብ ር ወ ል ደ ጊ ዮ ር ጊ ስ search and select the unit recorded from database አ.wav, ሸ.wav, ብ.wav, ር.wav, ወ.wav, ል.wav, ደ.wav, ጊ.wav, ዮ.wav, ር.wav, ጊ.wav, ስ.wav. The next step is margining those wave files. All the data is stored on the resources folder in the name of incoming text like E:\ Bekele\reading\አሸብር ወልደ ጊዮርጊስ .wav.

4.2.3.3 Concatenate Wave Files

For the concatenate approach less memory and less run time were used. Therefore, the separate wave phoneme files are merging or concatenating in order to incoming text sequences, and pass to audio to play function.

```
80                  public final void Merge(ArrayList<String> files, String outfile)
81                          throws FileNotFoundException, IOException
82                  {
83                          WaveIO wa_IN = new WaveIO();
84                          WaveIO wa_out = new WaveIO();
85                          wa_out.DataLength = 0;
86                          wa_out.length = 0;
87                  //Gather header data
88                          for (String path : files)
89                          {
90                                  wa_IN.WaveHeaderIN(path);
91                                  wa_out.DataLength += wa_IN.DataLength;
92                                  wa_out.length += wa_IN.length;
93                          }
94                          //Recontruct new header
95                          wa_out.BitsPerSample = wa_IN.BitsPerSample;
96                          wa_out.channels = wa_IN.channels;
97                          wa_out.samplerate = wa_IN.samplerate;
98                          wa_out.WaveHeaderOUT(outfile);
99
100                         for (String path : files)
                            {java.io.FileInputStream fs = new java.io.FileInputStream(path);
102                           byte[] arrfile = new byte[path.length() - 44];
103                                 fs.getChannel().position(44);
104                                 fs.read(arrfile, 0, arrfile.length);
105                                 fs.close();
106                                 java.io.FileOutputStream fo = new java.io.FileOutputStream(outfile, true);
107                                         //BinaryWriter bw = new BinaryWriter(fo);
108                                          BufferedOutputStream bw = new  BufferedOutputStream(fo);
109                                         bw.write(arrfile);
110                                         bw.close();
111                                         //fo.Dispose();
112                                         fo.close();
113                         }
114
```

Figure 4-6 Concatenate Wave files

Figure 4.6 show, the code which indicates concatenated phoneme wave files. There were two tasks that are done for merging a single phoneme to the word. The first task is gathering header data based on the recording phone wave files. The second task is reconstructing new header based on the input header. Finally, concatenate the wave files in terms of a new header.

4.2.3.4 Audio Generation

The system reads each sentence corresponds to the wave. Then the .wav files concatenated and played according to the input header of the waves. The text file contains a number of lines. Line by line, the words are concatenated and played. For instance, figure 4.5 are shown as played አሸባር ወልዶ ጊዮርጊስ .wav audio utterances files. For example, the system produced the retrieved answer in the form of a wave in below figure 4.7 the answer is አሸባር ወልዶ ጊዮርጊስ.

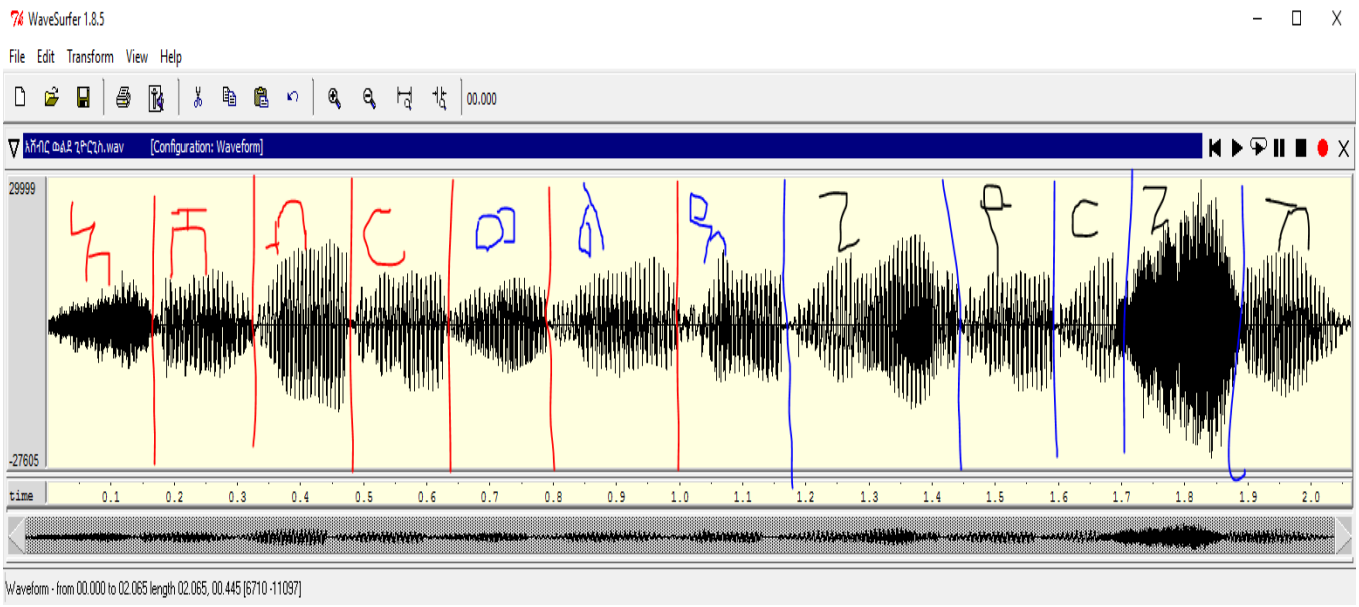


Figure 4-7 speech utterance each phoneme

Figure 4.7 shows how the phoneme wave files are concatenated the appropriate text and produce speech utterances. The wave files are extracted from the database and merged until the input text finished. At that point, the systems that produce the text will concatenate the audio wave format.

To this end, speech recognition and speech synthesis tasks are integrated for Amharic question answering with an automatic question classification.

CHAPTER FIVE
EXPERIMENTATION AND EVALUATION

The experiments of the system were conducted in order to evaluate an automatic question classification for speech based Amharic question answering system. The system can produce the retrieved answer in the form of Audio-based. The reason behind taking performance evaluation is to identify the extent to which the study work and for the further recommendation. This chapter presents the detailed experimentation criteria for the subsystems of speech recognition, question answering system, and speech synthesis. Those components are independently tested.

A prototype system based on the proposed method has been constructed and an experiment on 84 natural language questions, 2016 corresponding speech corpus, and 233 phoneme & digits wave files was prepared. Then, we discussed experimental results and findings of the study.

5.1 Experimentation and Evaluation for Speech Recognition

Primarily, the speech recognizer accepts speech inputs and recognize sentences. A speech recognition systems are composed of speech database, Amharic dictionary, language model, lexical model, and acoustic model that are used to transfer voice into sequence text using sphinx decoder.

For this study, the recording has been taken place using 24 different speakers. These speakers were from Debre brehan whose first language is Amharic. Table 5.1 shows a training speaker information. Three different data sets were taken from the database as a training set, a development set, which was used for testing and fine tuning during development of the system and an evaluation test set to evaluate the final performance of the system. Table 5.1 below summarizes the type and size of data set used for this work.

Table 5-1 Dataset used for training

No	No of Speakers	Age Range	Sex
1	8	15-30	M
2	7	15-30	F
3	9	31-50	M
Total	24		

The speech corpus included 2,016 question sentences spoken by 24 (9 Female and 15 Male) different speakers, each speaker reads 84 questions utterance for the training purpose. A speech corpus was generated by recording and cutting, the sentences. The prepared and parameterized speech samples are then recognized using wave surfer tools. For testing and training speech recognition, 84 questions were classified into two parts, such as person and numeric.

In this study, training of a speech recognition system depends on the collected data (text corpus and speech corpus). The data we used for the system were collected by the previous researchers. Additionally, five speaker's speech corpus data's were added to the training. The result of the training shows that 90% is error sentences and 10% is word error. From the training, we observed that the researcher used manual word aligner because there was no any Amharic (Unicode) aligner that modifies and delete the words. Hence, during the training we measured the accuracy speech recognition with 91.65% and word error rate, 6.35% is registered.

Table 5-2 Experimental result of speech recognition

Speaker	Number of questions	Number of Speech Word utterances	Number of recognized words	Number of substituted word	Number of deleted word	Number of inserted word	Accuracy
1	23	191	167	19	10	14	87.4%
2	23	191	160	24	8	12	83.7%
Average accuracy= 85.58%							

Table 5-2 shows the experiment result of speech recognition. The evaluation is measure based on how many words are recognized over total words (see section 3.6).

23 questions were asked by two speakers. The first speaker has trained and achieved the performance of 87.4% accuracy and 12.6% word error rate. And also the second speaker is non-trained and achieved the performance of 83.7% and 16.3% word error rate. The average accuracy of speech recognition system performs 85.58%.

The major challenges of the system not to perform more than 85.58% for speech recognition was due to the noise environment, the age of the participant different accents and the distance between speaker and microphone which reduced the system performance.

5.2 Experimentation and Evaluation for QA System

The previous' study used rule-based question classification approach. When the user asks any question, the system automatically classifies a question based on expected keyword like (**ማን**). In this classification, the word **ማን** can be "person" or "place". In rule-based classification, the system does not accurately classify the question due to the wrong classification. For example, when the user asks "**የኢትዮጵያ አግር ኳስ ፌዴሬሽን ፕሬዝዳንት ማን ይባላል**" the expected answer should be "person". But, the ASR system is recognized a voice question and convert into the text as "**የመንገድ ኳስ ፌዴሬሽን ፕሬዝዳንት ማን ይባላል**" the first answered will be "place" as shown in figure 5.1. However, the system mistakenly classifies a question, which means it outputted "place" as the expected answer rather than "person". And also the speech recognition system has not recognized all question keyword, in this moment the question classification is based on keyword. These things are the main gap we identified from the rule-based question classification.

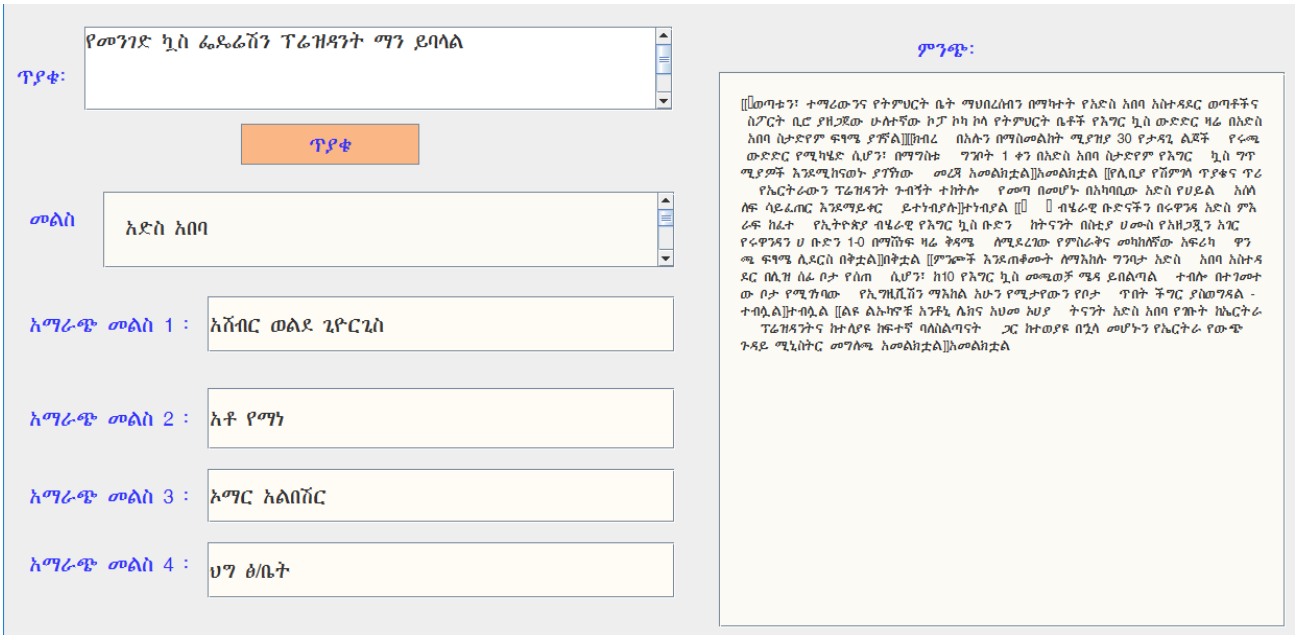


Figure 5-1 Rule-Based Question Classifier

The correct answer should be the one which is displayed on the second text field, but the system displays the first answer as if it is the correct one. In rule-based question classification, if the speech recognition systems are not uttered the classifier keyword, the system will not retrieve the correct answer.

We have evaluated the performance of the question answering system based on rule-based and SVM in speech recognition. There are 87 training questions for two types of question "person" and "numeric " and for testing, we use 23 questions shows as in appendix III.

Table 5-3 Speech recognition retrieval on Rule-Based and SVM question classifier

Algorithm	Questions	Correct answered	Not Answered	Not Exactly Answered	No Answers	precision	Recall	F-measure
Rule-based	23	17	2	2	2	82.60%	95%	88.36%
SVM based	23	15	5	2	1	73.91%	94.44%	82.92%

The performance of question classification for speech based Amharic question answering system retrieved 82.60% precision and 95% recall on Rule-Based question classifier. Also, the study of the system retrieved 73.91% precision and 94.44% recall using SVM classifier.

The challenge of SVM question classification is a query is exist in different class so, SVM is not clear classifies a question and also, some question didn't have an answer, due to pattern matching algorithms challenges and their questions have unclear and indirectly there haven't semantic similarity WordNet. For example, as Figure 5.2 depicted the QA system did not generate an answer for the question "የአደራ ፊልም ዋና ገጸ ባህሪ ማን ናት.?" For the next investigation, we recommend using semantic ontology analyzer.

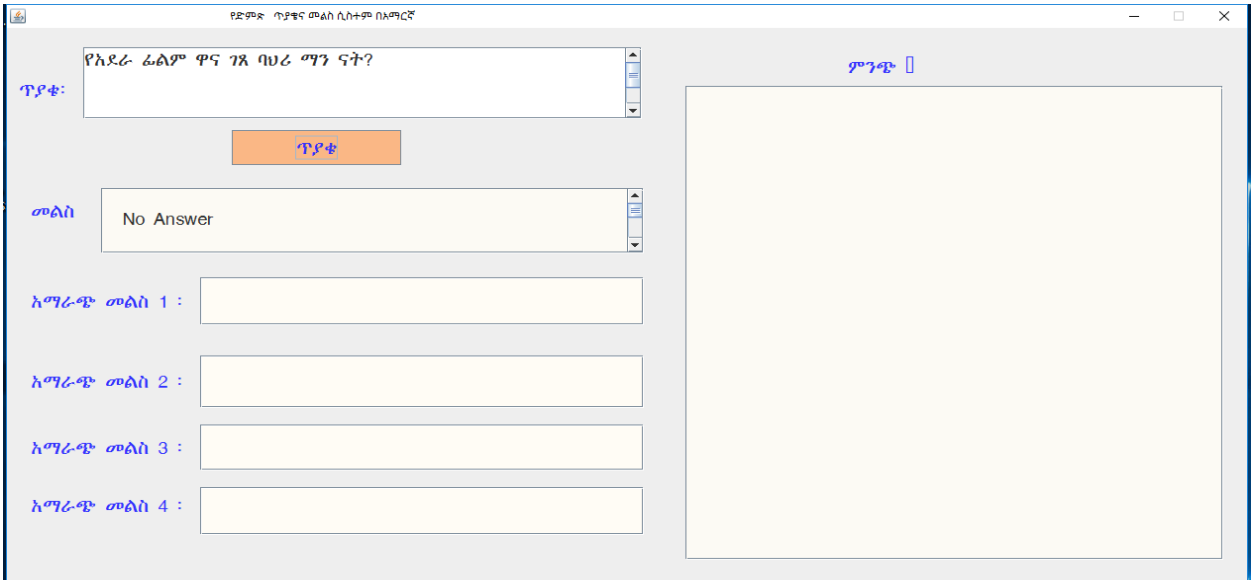


Figure 5-2 No answer of specific question

5.3 Experimentation and Evaluation for Speech Synthesis

In the part of speech synthesis, concatenate unit selector approach were used. The speech synthesis performance was measured in two ways. The first of these tests was performed to see how many words are pronounced and not.

Table 5-4 speech synthesis performance measure

	Performance Measure on the Test Data				
	Total words	Correctly Pronounced	Partially Pronounced	Not Correctly Pronounced	Not answer and Not pronounced
No. of Words	47	38	5	2	2
Accuracy %		80.86	10.64	4.25	4.25

The performance measurement of the speech synthesis on the test data achieves 80.86% correctly pronounced, 10.64% partially pronounced, 4.25 % not correctly pronounced and 4.25% results as not answer and not pronounced.

The purpose of TTS system is producing the audio of the input text with less response time. Furthermore, to get quality output with naturalness and intelligibility and to concatenate speech synthesizer, we use unit selection approach. To measure the overall ineligibility and naturalness

of the system we have used MOS (Mean Opinion Score method). Performance evaluation scale is given as follow: **1 = bad, 2 = poor, 3 = good, 4 = very good and 5 = excellent.** The testing evaluation takes five queries and five answers for each question.

There are ten listeners who are used to test the synthesizer. The synthesizer scores averaged on MOS rating. The synthesizer is rated as: excellent if it has an MOS value greater than or equal to 4.5, very good if it has an MOS value greater than or equal to 3.5 and less than 4.5, good if it has an MOS value greater than or equal to 2.5 and less than 3.5, poor if it has an MOS value greater than or equal to 1.5 and less than 2.5, or bad otherwise.

Table 5-5 Experimental results of speech synthesis for Naturalness

Listeners	***Test Data***																									
	Question 1					***Question 2***					***Question 3***					***Question 4***					***Question 5***					***Scores***
	Answer 1- 5					***Answer 1- 5***					***Answer 1- 5***					***Answer 1- 5***					***Answer 1- 5***					
1	***3***	***3***	***3***	***2***	***2***	***3***	***3***	***3***	***3***	***3***	***3***	***3***	***3***	***2***	***2***	***3***	***2***	***3***	***3***	***2***	***3***	***2***	***2***	***2***	***2***	***2.6***
2	***3***	***2***	***3***	***2***	***3***	***2***	***3***	***3***	***3***	***3***	***3***	***3***	***2***	***1***	***3***	***2***	***3***	***2***	***4***	***3***	***3***	***3***	***2***	***3***	***2***	***2.64***
3	4	4	5	2	3	4	3	5	3	3	4	5	5	3	4	5	5	4	4	3	5	4	5	4	5	***4.04***
4	2		2	2	3	3	2	3	2	2	3	4	3	3	1	3	3	3	3	3	3	3	3	2	2	***2.62***
5	3	3	3	3	1	2	3	2	3	2	3	3	3	2	3	3	2	3	4	3	2	3	3	3	3	***2.72***
6	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	3	2	3	3	3	3	2	3	3	***2.88***
7	1	5	5	2	4	4	3	4	4	3	4	4	5	2	4	4	4	2	4	3	5	4	5	4	5	3.76
8	1	4	4	2	3	3	4	4	4	3	4	5	4	4	5	4	3	2	3	3	4	5	5	5	5	3.72
9	1	5	5	3	4	4	3	4	5	4	3	3	5	4	4	2	2	5	4	3	5	4	4	4	5	3.8
10	4	4	5	2	3	4	3	4	4	4	4	4	5	5	4	4	4	4	5	4	4	4	4	5	5	4.08
***Average* scores %**																										3.45

Table 5-6 Experimental results of speech synthesis for intelligibility

Listeners	*Test Data*																									
	Question 1					*Question 2*					*Question 3*					*Question 4*					*Question 5*					*Scores*
	Answer 1- 5					*Answer 1- 5*					*Answer 1- 5*					*Answer 1- 5*					*Answer 1- 5*					
1	4	4	3	2	5	3	3	4	4	3	5	3	3	2	3	3	4	4	3	5	3	4	4	4	5	3.6
2	2	2	3	3	5	2	3	3	3	3	3	3	3	3	2	3	2	3	3	4	3	4	3	2	4	2.96
3	2	1	2	4	3	3	2	4	2	2	3	2	3	2	3	2	3	2	3	4	3	3	3	3	3	2.68
4	3	3	5	2	5	3	2	4	3	3	5	4	3	2	5	4	3	3	3	5	3	4	4	4	5	3.6
5	3	2	2	2	2	3	2	2	3	3	2	3	2	3	3	3	2	3	4	2	3	3	3	3	3	2.64
6	2	3	3	3	3	4	3	3	4	4	3	4	4	3	3	4	4	4	4	3	4	3	3	4	5	3.48
7	1	2	3	3	3	2	3	3	2	2	3	3	3	2	2	2	3	2	3	4	5	3	3	3	3	2.72
8	4	3	5	2	5	3	3	4	4	3	5	4	3	2	5	4	5	3	3	5	3	4	4	4	3	3.72
9	1	2	2	4	1	3	3	2	3	3	3	2	3	3	3	3	3	3	3	3	2	3	2	3	3	2.64
10	5	4	4	2	4	3	2	4	4	3	5	4	3	2	4	3	4	4	3	5	3	4	4	3	5	3.64
***Average* accuracy%**																										3.17

The complete result of speech synthesizer achieved 80.86% accuracy using concatenate approach. The general intelligibility and naturalness of the system. Out of five questions, ten listeners perform 3.17% and 3.45% accuracy. Which means the goodness of the synthesizer achieves based on the scale of MOS.

To produce understandable and natural sounding synthesis speech, connecting the prerecorded natural utterances is enhanced. However, we observed that the output 19.14 % quality becomes in a weak position due to the problems occurred by concatenation of phonemes and words in the concept of Stops, Affricatives, Nasals, and Liquids. And also the recording speech tools and noise environment in the recording time and a variety of phoneme recording time problems.

5.4 User Acceptance Testing (UAT)

User acceptance testing (UAT) is the last phase of the software testing process**,** the aim of task user acceptance testing is to make sure how well an automatic question classification for SBAQA is performing on the users to make sure that the system is accepted and usable by users. Ten visually impaired users were selected to test the system. The technology acceptance model (TAM)

is an information systems theory that models how users come to accept and use a technology. The model suggests that when users are presented with a new technology, a number of factors influence their decision about how and when they will use it [76]. The evaluators evaluated an automatic question classification for SBAQA by using the following standards [76];

1. Using an automatic question classification for SBAQA system enable to better informed more quickly
2. Do you find the automatic question classification for SBAQA system useful for receiving knowledge
3. Using an automatic question classification for SBAQA system would improve access to question answering service performance
4. It would be easy to get an automatic question classification for SBAQA system to do what I want to get answer
5. It is easy to use an automatic question classification for SBAQA system with training
6. Do you like the idea of using an automatic question classification for SBAQA system
7. will you use an automatic question classification for SBAQA system in the future

Evaluation of the system using user acceptance testing is done to make sure how the users 'view the system on the base of the above-mentioned evaluation standards. Different researchers have used different types of user acceptance testing evaluation criteria. But for this study, the evaluation criteria suggested by [76] is customized and used to ease the evaluation process. The weights scale Suggested [76] has been used such that strong agree = 4, agree=3, neutral=3agree =2 and disagree=1 Table 5-7 depicts that summary of user acceptance evaluation of the system [76].

Table 5-7 summaries of Users' evaluation of the system

No-	Criteria of evaluation	disagree(1)	neutral (2)	Agree (3)	Strong agree (4)	Average
1	Using an automatic question classification for SBAQA system enable to better informed more quickly	2	4	4	0	2.8
2	Do you find the automatic question classification for SBAQA system useful for receiving knowledge	1	3	4	2	2.7
3	Using an automatic question classification for SBAQA system would improve access to question answering service performance	2	2	4	2	2.6
4	it would be easy to get the automatic question classification for SBAQA system to do what I want to get an answer	1	3	4	2	2.7
5	It is easy to use the an automatic question classification for SBAQA system with training	0	3	4	3	3.0
6	Do you like the idea of using the SBAQA system	0	2	4	4	3.2
7	will you use the SBAQA system in the future	0	2	3	5	3.4
						Total Average=2.91

5.5 Discussion

As table 5-7 shows, 20% of evaluators replied disagree for "**using an automatic question classification for SBAQA system enable to better inform more quickly"**. 40% of respondents replied neutral and agree for it. With regard to the second question, which is "**Do you find the automatic question classification for SBAQA system useful for receiving knowledge"** 10%, 30%, 40%, 20% of evaluators replied disagree, neutral, agree and strong agree. The third question is about **Using an automatic question classification for SBAQA system would improve access to question answering service performance** 20%, 20%, 40%,20% of evaluators replied disagree, neutral, agree and strong agree

The fourth evaluation criteria are about **it would be easy to get the automatic question classification for SBAQA system to do what I want to get the answer** 10%, 30%, 40%, 20% of evaluators replied disagree, neutral, agree and strong agree. The fifth question is about **it is easy to use the automatic question classification for SBAQA system with training** 30% of evaluator scored natural and 40% of them as agree. Whereas 30% of the evaluator scored the system as strong Agree.

The other evaluation criterion is **Do you like the idea of using the automatic question classification for SBAQA system.** Among the evaluators**,** 20% replied neutrally for SBAQA and 40% of them replied agree and 40% are strong agree.

The last criterion is **will you use the automatic question classification for SBAQA system in the future** 20% of evaluators replied neutral, 30% replied agree and 50% replied strong agree. This implies that the contribution of a constructing system like SBAQA is important in the area of question answering. Finally, according to the evaluation filled by target users, SBAQA has registered 2.91 out of 4(72.75%) which is taken as agree on achievement.

Weakness

- Some user thinks using **automatic question classification for SBAQA** system not helping them to better inform them quickly
- It is the rigid amount of data they get from the system. They think it's better they get more information rather than the available data in the system (need to receive more knowledge)
- They don't believe that the **automatic question classification for SBAQA** system would improve access to question answering service performance.

Strength

- The users surely believe that they can use the system with training. (They think the system is usable).
- As the system is not available here in our country Ethiopia the users are more satisfied and like the idea of using the system
- The users are willing to use the system for the future due to system availability and accessibility and performance.

5.6 Findings and Challenges

In this research, we attempt to integrate Amharic speech recognition and speech synthesis with Amharic question answering system using automatic question classification that enables the question answering to accept speech query and retrieves precise textual answers and then produce answers in the form of audio to the users for a specific query.

The key findings in this research are developing Amharic continuous speech recognizer and synthesis, using question classification techniques of the question answering by applying Support Vector Machines (SVM) Algorithm based on the performance achieved, and integrating the speech synthesis with the question answering.

Experimental result shows that, the Amharic speech recognizer for question corpus resisted 85.58% recognition rate. Compared with other related works such as Amharic Speech recognizer for question answering question rate of 84.93% [15], SpeechGoogl with recognition rate of 87.17% [16] and Ikkyu with recognition rate of 84.5%, the performance is almost appreciable.

The other finding is related to the Question classification with question answering system. The experiments are done on rule based and SVM based question classification techniques with a specific query. Consequently, SVM based question classification technique that registered 73.91% precision is selected. It also shows less performance result compared with the previous research's [15]that is registered 76%.

The speech synthesis are used a concatenate approach and better performance registered 80.86% word pronounced and naturalness (3.17) and ineligibility (3.45) comparing with the other system.

This study achieved good results in speech based Amharic question answering. Some challenges have encountered that delay the system from scoring a better achievement. The limitation of this study is no semantic similarity parsing. Speech recognition and speech synthesis problems are one challenge of the study.

CHAPTER SIX
CONCLUSION AND RECOMMENDATION

6.1 Conclusion

This study show the possibility of constructing a prototype of automatic question classification speech based Amharic question answering explored. The user provides the question in the form of Amharic natural language using STT. The user questions are compared with the documents to extract the required answer from question answering systems and produce the answer in the form of audio waveform (TTS). The first input processing was done by speech recognition that used speech corpus for training and testing. This study uses speech recognition method. This method includes different components like feature extraction, acoustic model, language model, dictionary model and decoding. In preparing the language model, we used online tools and sphinx3, sphinx-base, sphinx-train for training tools and also sphinx4 are used for the decoder. Finally, the question answering systems integrated using NetBeans java programming language.

The second task was designing factoid question answering systems for information retrieval and information extraction. The question answering part has three main components, namely, question analysis, document retrieval, and answer extraction. The question analysis component takes in user's request, identifies the type of fact the user needs as a person, time, place, or quantity. This is called question classification and it is done in a machine learning (statistical) approach used in the training and testing LibSVM library tools easily and fast classifier support in java platform. Document retrieval is the component which receives generated queries from the preceding question analysis component and looks for documents containing information related to what the user has asked. This searching is made in the index directory containing indexed documents. When relevant documents are found, they will be ranked and returned to the answer extraction component. Finally, the answer extraction is a component responsible for digging out the fact which is believed to be an answer to the user's question and to return these facts in a certain order back to the user.

Speech synthesis is a continuously developing technology since many years. This paper has presented various aspects of speech synthesis, such as methods, tools, techniques, applications which are currently available in various languages. Speech synthesis is producing natural sounding and intelligible speech output. The study used Amharic phoneme units which have recording and

processing and the approaches are concatenate speech synthesis approach to constructing and concatenate the phonemes into words. In question answering retrieved ranked top, five answers and TTS components can produce in the form of audio to users. In the implementation, TTS is done by java programming language. Finally, we integrated all components to fulfill the gaps using those methodological and archived their objectivities.

The performance resulted in shows that the ASR is 85.58% accurate. The first challenge is very high, noise environment and speaker distance. In the testing part, we used two users; trained person and non-trained person. Trained person has a high performance and non-trained is less because of this research does not deal with the Amharic five dialects. If the speech corpus is more focused on five dialects, the performance of the speech recognized would increase.

The performance of question answering system retrieved 73.91% precision and 94.44% recall on SVM-based question classifier. Also, the system retrieved 82.60% precision and 95% recall on rule based classifier. Finally, we tested the speech synthesis and the performance measurement of the speech synthesis on the test data achieves 80.86% correctly pronounced, 10.64% partially pronounced, 4.25% not correctly pronounced and 4.25% results as not answer and not pronounced. Based on the scale of the MOS test, the intelligibility is 3.17. Which means the synthesizer is "good" which is based on the scale of the MOS test and naturalness of the synthesizer found to be 3.45 which also approach to "Good" MOS scale.

The system has register 72.75% overall accuracy according to the system performance and user acceptance tests using ten visually impaired users were used. The performance analysis shows that automatic question classification for SBAQA registers acceptable performance. The QA based system also provides precise information to the target groups. This system is limitation on retrieves the expected answers in the case of a question keyword is no semantic similarity with the document.

In the speech recognition dialects problem, a distance of speaker and noise environment were challenging. In the speech synthesis, the difficult is germination come across in Amharic speech when the low tone and high tones. However, further exploration and study have to be done to refine and yield a better in this system which can be deployed more focused visually impaired users and provide information for a specific question, so that they can take timely and appropriate actions.

6.2 Recommendation

Based on the experimental results and what has been learned throughout the research the following recommendations are forwarded:

- We recommend semantic similarity praising ontology-based domain-specific natural language question answering should be used that applies semantics and domain knowledge to improve both question construction and answer extraction.
- Working on other question types. Our research aims at factoid questions answering of type "Person", and "Numerical". But, "Place", "Time", in factoid question and there are other question types like "Definition", "List", "How", "Acronym", and so on. So, we believe they are potential concepts for research in having a complete question answering system.
- Noises: There are no perfect noise detection techniques, which are applicable to natural language processing in part of speech technologies for question answering. So for the next investigation, we recommend removing noises when question speech are greater than noise.
- The system could be customized in different ways; as a Web service, mobile application, web application or even as a java API, robotics dialog systems will be better for to alleviate the problems.
- More information can be utilized in question answering module, especially, the answer part of QA pair. Besides, deeper syntactic and semantic parser, such as question-type and expected answer prediction also can be considered in the ranking function of QA pairs using different machine language model like artificial neural network, Naïve Bayes, and Decision Trees etc.
- The system will be deployed with different areas likes; online learning application, Ethiopia Aircraft Corporation, in the agriculturalist area etc. to alleviate the real life of visually impaired users.
- In question classification for speech based Amharic question answering system is fixed based on question corpus. We recommend using different methods or techniques to distinguish dynamic questions from static questions not depend on training question.
- The research is done on illiterate and visually impaired users for monolingual Amharic language speakers. But, in Ethiopia, there are more than 80 languages so the next researchers' will propose a multilingual audio interface.

References

[1] J. Daniel and H. M. James , "Question Answering and Summarization," in *An introduction to natural language processing, computational linguistics, and speech recognition*, Speech and Language Processing, October, 29, 2007, p. 1004.

[2] A. Lampert, "A quick introduction to question answering," *Citeseer*, 2004.

[3] C. Manning , P. Raghvan and H. Schutze , "Introduction to information retrieval," vol. 1, Cambridge, Cambridge university press, 2008, p. 496.

[4] J. Liu, "An Intelligent FAQ Answering System Using a Combination of Statistic and Semantic IR Techniques.," *Department of Computer Science and Electronics at Malardalen University*, 2007.

[5] T. Mishra and S. Bangalore, "Qme! : A Speech-based Question-Answering system on Mobile Devices," in *Association for Computational Linguistics*, Los Angeles, California, 2010.

[6] A. M. N. Allam and M. H. Haggag, "The question answering systems: A survey," *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, vol. 2, 2012.

[7] N. Schlaefer, "Deploying semantic resources for open domain question answering (Doctoral dissertation,)," *Diploma Thesis. Language Technologies Institute School of Computer Science Carnegie Mellon University*, 2007.

[8] M. Seid and . L. Mulugeta, "TETEYEQ: Amharic Question Answering For Factoid Questions," *IE-IR-LRL*, vol. 3, p. 17, 2009.

[9] W. Teshome, "Designing Amharic Definitive Question Answering," *AAU*, 2013.

[10] D. Demner-Fushman, "Complex Question Answering Based on Semantic Domain Model of Clinical Medicine," *IEEE*, 2006.

[11] J. Parikh and M. N. Murty, "Adapting Question Answering Techniques to the Web," in *IEEE*, 2002.

[12] W. Nemera, "light for the world in ethiopia," world , 2012. [Online]. Available: http://www.light-for-the-world.org/ethiopia/. [Accessed 17 july 2016].

[13] M. Fantahun and P. Korpinen, "Inclusion of People with Disabilities in Ethiopia," 2013.

[14] C. R. Murphy, "Computers Assisting The Handicapped," 2007. [Online]. Available: http://courses.cs.vt.edu/cs3604/lib/Disabilities/murhpy.AT.html. [Accessed 17 july 2016].

[15] Belisty.Y, "(TASBFQA): Towards Amharic Speech Based Factoid Questions Answering," 2014.

[16] G. Hu, Dan, Q. Liu and R. Wang, "SpeechQoogle: An Open-Domain Question Answering System with Speech Interface," *Presented at International symposium on Chinese spoken language processing*, 2006.

[17] H. Seid and B. Gamback, "A speaker independent continuous speech recognizer for Amharic," 2005.

[18] M. Berndtsson and J. Hansson, Thesis Projects: A Guide for Students in Computer Science and Information Systems:, London: Springer Science \& Business Media, 2007.

[19] C. R. Kothari, Research Methodology: Methods and Techniques, 2. Ed., Ed., New Delhl: New Age International, 2004.

[20] S. T. March and G. F. Smith, "Design and natural science research on information technology," *Decision support systems*, vol. 15, pp. 251--266, 1995.

[21] K. Peffers, T. Tuunanen, M. Rothenberger and S. Chatterjee, "A design science research methodology for information systems research," *Journal of management information systems*, vol. 24, pp. 45--77, 2008.

[22] R. H. Von Alan, S. T. March, J. Park and S. Ram, "Design science in information systems research," *MIS quarterly*, vol. 28, pp. 75--105, 2004.

[23] A. M. Moussa and R. F. Abdel-Kader, "Qasyo: A question answering system for yago ontology," *International Journal of Database Theory and Application*, vol. 4, pp. 99-112, 2011.

[24] S. Schlobach, M. Olsthoorn and R. De , "Type checking in open-domain question answering," in *ECAI*, vol. 16, 2004, p. 398.

[25] U. Shah, T. Finin, A. Joshi, R. S. Cost and J. Matfield, "Information retrieval on the semantic web," in *Proceedings of the eleventh international conference on Information and knowledge management*, ACM, 2002, pp. 461--468.

[26] M. Ehrig, S. Staab and Y. Sure, "Bootstrapping ontology alignment methods with APFEL," in *International semantic Web conference*, Springer, 2005, pp. 186--200.

[27] D. I. Moldovan and S. Harabagiu, "Lasso: A Tool for Surfing the Answer Net," pp. 65-73, 1999.

[28] W. Li, "Question classification using language modeling," *Center of Intelligent Information Retrieval (CIIR). Technical Report*, 2002.

[29] S. R. Gunn and others, "Support vector machines for classification and regression," *ISIS technical report*, vol. 14, 1998.

[30] "Hyperplane," 12 october 2015. [Online]. Available: https://en.wikipedia.org/wiki/Hyperplane. [Accessed 17 july 2016].

[31] C. Chih-Chung and L. Chih-Jen , "LIBSVM -- A Library for Support Vector Machines," [Online]. Available: https://www.csie.ntu.edu.tw/~cjlin/libsvm/. [Accessed 19 Febrary 2016].

[32] J.-C. Junqua and J.-P. Haton, Robustness In Automatic Speech Recognition: Fundamentals and Applications., vol. 341, Boston: Springer Science \& Business Media, 1996.

[33] V. Radha and C. Vimala, "A review on speech recognition challenges and approaches," *doaj. org*, vol. 2, pp. 1-7, 2012.

[34] A. T. Solomon and Y. T. Martha , "Amharic speech recognition: past, present and future," in *16th Int. Conf. Ethiopian studies*, 2006.

[35] B. Solomon, "Isolated Amharic Consonant-Vowel syllable recognition: An Experiment using the Hidden Markov Model," *MSc Thesis, School of Information Studies for Africa*, 2001.

[36] A. T. Solomon , M. Wolfgang and T. Bahiru , "An Amharic speech corpus for large vocabulary continuous speech recognition," *In Proceedings of 9th Europ. Conf. onSpeech Communication and Technology*, 2005.

[37] L. Rabiner and B.-H. Juang, "Fundamentals of Speech Recognition.," *Prentice hall*, 1993.

[38] P. Saini and P. Kaur, "Automatic speech recognition: A review," *International journal of Engineering Trends & Technology*, pp. 132-136, 2013.

[39] N. Alcaraz Meseguer, "Speech analysis for automatic speech recognition," *Institutt for elektronikk og telekommunikasjon*, 2009.

[40] S. K. Gaikwad, B. W. Gawali and P. Yannawar, "A review on speech recognition technique," *International Journal of Computer Applications*, vol. 10, pp. 16--24, 2010.

[41] M. Anusuya and S. K. Katti , "Speech recognition by machine, a review," *arXiv preprint arXiv:1001.2267*, 2010.

[42] M. Frikha, S. Ben Massaoud, M. A. Kammoun, D. Gargouri, M. Lahyani and A. Ben Hamida, "Optimizing Some HMM Model Parameters in an Isolated Speech," *In IEEE conference SSD05,Sousse, Tunisia*, 2005.

[43] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," *Foundations and trends in signal processing*, vol. 1, pp. 195--304, 2008.

[44] X. Huang and . L. Deng, "An Overview of Modern Speech Recognition," *An Introduction to Language Processing with Perl and Prolog*, 2010.

[45] T. Dutoit, "High-quality text-to-speech synthesis: an overview," *Journal of Electrical & Electronics Engineering, Australia: Special Issue on Speech Recognition and Synthesis*, vol. 17, pp. 25-33, 1996.

[46] T. K. Dagba and C. Boco, "A text to speech system for Fon language using multisyn algorithm," *Procedia Computer Science*, vol. 35, pp. 447--455, 2014.

[47] A. Balyan, S. Agrawal and A. Dev, "Speech Synthesis: A Review," in *International Journal of Engineering Research and Technology*, 2013.

[48] M. Singh, "Text to Speech Synthesis for numerals into Punjabi language," *THAPAR UNIVERSITY PATIALA*, no. Doctoral dissertation, , 2013.

[49] T. Dutoit, "An introduction to text-to-speech synthesis," *Springer Science & Business Media.*, vol. 3, 1997.

[50] I. Isewon, J. Oyelade and O. Oladipupo, "Design and Implementation of Text To Speech Conversion for Visually Impaired People," *International Journal of Applied Information Systems,* vol. 7, pp. 25-30, 2014.

[51] A. Indumathi and E. Chandra, "Survey on speech synthesis," *Signal Processing: An International Journal (SPIJ),* vol. 6, p. 140, 2012.

[52] E. Keller, Fundamentals of speech synthesis and speech recognition: basic concepts, state-of-the-art and future challenges, John Wiley and Sons Ltd, 1995.

[53] P. Nugues, "An overview of speech synthesis and recognition," *An Introduction to Language Processing with Perl and Prolog,* pp. 1-22.

[54] "Speech Synthesis," [Online]. Available: http://en.wikipedia.org/wiki/Speech_synthesis. [Accessed 23 auguest 2015].

[55] S. H. Mariam, S. P. Kishore, A. W. Black, R. Kumar and R. Sangal, "Unit selection voice for amharic using festvox," in *IEEE*, 2004.

[56] T. Yoshimura, "Simultaneous modeling of phonetic and prosodic parameters, and characteristic conversion for HMM-based text-to-speech systems," 2002.

[57] T. Kinfe, "Sub-word based amharic word recognition: An experiment using hidden markov model (HMM)," no. Doctoral dissertation, M. Sc Thesis., 2002.

[58] B. Kasaye and others, "Developing a Speech Synthesizer for Amharic Language Using Hidden Markov Model," *Addis Ababa University,* 2008.

[59] b. T. Solomon and . M. Wolfgang, "Automatic speech recognition for an under-resourced language-amharic," in *INTERSPEECH*, 2007, pp. 1541--1544.

[60] A. T. Solomon and M. Wolfgang , "Syllable-based speech recognition for Amharic," *Association for Computational Linguistic,* pp. 33-40, 2007.

[61] M. L. Bender, S. W. Head and R. Cowley, "The Ethiopian Writing System," *Language in Ethiopia. Edited by ML Bender, JD Bowen, RL Cooper, and CA Ferguson. London: Oxford University press,* pp. 120-130, 1976.

[62] A. Getahun, "Modern Amharic Grammar in a simple approach," *AAU,* 1996.

[63] "Sphinx-4 Speech recognizer written entirely in Java," Carnegie Mellon University, 12 octoober 2015. [Online].

[64] "Training Acoustic Model For CMUSphinx," Carnegie Mellon University, [Online]. [Accessed 2015].

[65] S. J. Owens, "http://darksleep.com/lucene/," Lucene Tutorial . [Online]. [Accessed 16 Auguest 2015].

[66] C. Chih-Chung and . L. Chih-Jen, "LIBSVM -- A Library for Support Vector Machines," [Online]. Available: https://www.csie.ntu.edu.tw/~cjlin/libsvm/. [Accessed 19 Febrary 2016].

[67] E. Schofield and Z. Zheng, "A speech interface for open-domain question-answering," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, vol. 2, Association for Computational Linguistics, 2003, pp. 177--180.

[68] X. Huang, A. Acero, H.-W. Hon and R. Reddy, Spoken language processing: A guide to theory, algorithm, and system development, Prentice hall PTR, 2001.

[69] D. Namrata , "Feature Extraction Methods LPC, PLP and MFCC In Speech Recognition," *jaret*, vol. 1, july 2013.

[70] Q. Guo and M. Zhang, "Question answering system based on ontology and semantic web," in *In Proceedings of the 3rd international conference on Rough sets and knowledge technology*, 2008.

[71] O. Gospodnetic and E. Hatcher, "Lucene," Manning, 2005.

[72] S. Zegaye, "HMM based large vocabulary, speaker independent, continuous Amharic speech recognizer," *MSc Thesis, School of Information Studies for Africa*, 2003.

[73] P. Rutten, G. Coorman, J. Fackrell and B. Van Coile, "Issues in corpus based speech synthesis," in *State of the Art in Speech Synthesis (Ref. No. 2000/058), IEE Seminar on*, IET, Ed., 2000, pp. 16--1.

[74] B.-H. Juang and L. R. Rabiner, "Automatic speech recognition--a brief history of the technology development," *Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara*, vol. 1, p. 67, 2005.

[75] A. Tewodros, "Text-to-Speech (TTS) system for the Wolaytta language," *addis abeb university informatics department MSc thesis*, 2009.

[76] Davis and D. Fred, "percived usefulness, user acceptance of information technology," *MIS quartly*, pp. 319--340, 1989.

Appendices

Appendix I

The Amharic Phonemes, Unicode representation

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

Appendix II

Amharic Phonetic List, IPA Equivalence, and it's ASCII Transliteration.

IPA	Transcription Amharic	Equivalence
Consonants		
[p]	[p]	ፕ
[t]	[t]	ት
[k]	[k]	ክ
[ʔ]	[ax]	ዕ
[b]	[b]	ብ
[d]	[d]	ድ
[g]	[g]	ግ
[p’]	[px]	ጵ
[t’]	[tx]	ጥ
[c’]	[cx]	ጭ
[q]	[q]	ቅ
[f]	[f]	ፍ
[s]	[s]	ስ
[ʃ]	[sx]	ሽ
[h]	[h]	ህ
[s’]	[xx]	ጽ
[tʃ]	[c]	ች
[g’]	[j]	ጅ
[m]	[m]	ም
[n]	[n]	ን
[n’]	[nx]	ኝ
[l]	[l]	ል
[r]	[r]	ር
[j]	[y]	ይ
[w]	[w]	ው
[v]	[v]	ቭ
[z]	[z]	ዝ
[z’]	[zx]	ዥ
Vowels		
[E]	[e]	ኧ
[u]	[u]	ኡ
[r]	[ii]	ኢ
[a]	[a]	አ
[e]	[ie]	ኤ
[i]	[ix]	እ
[o]	[o]	ኦ

Appendix III

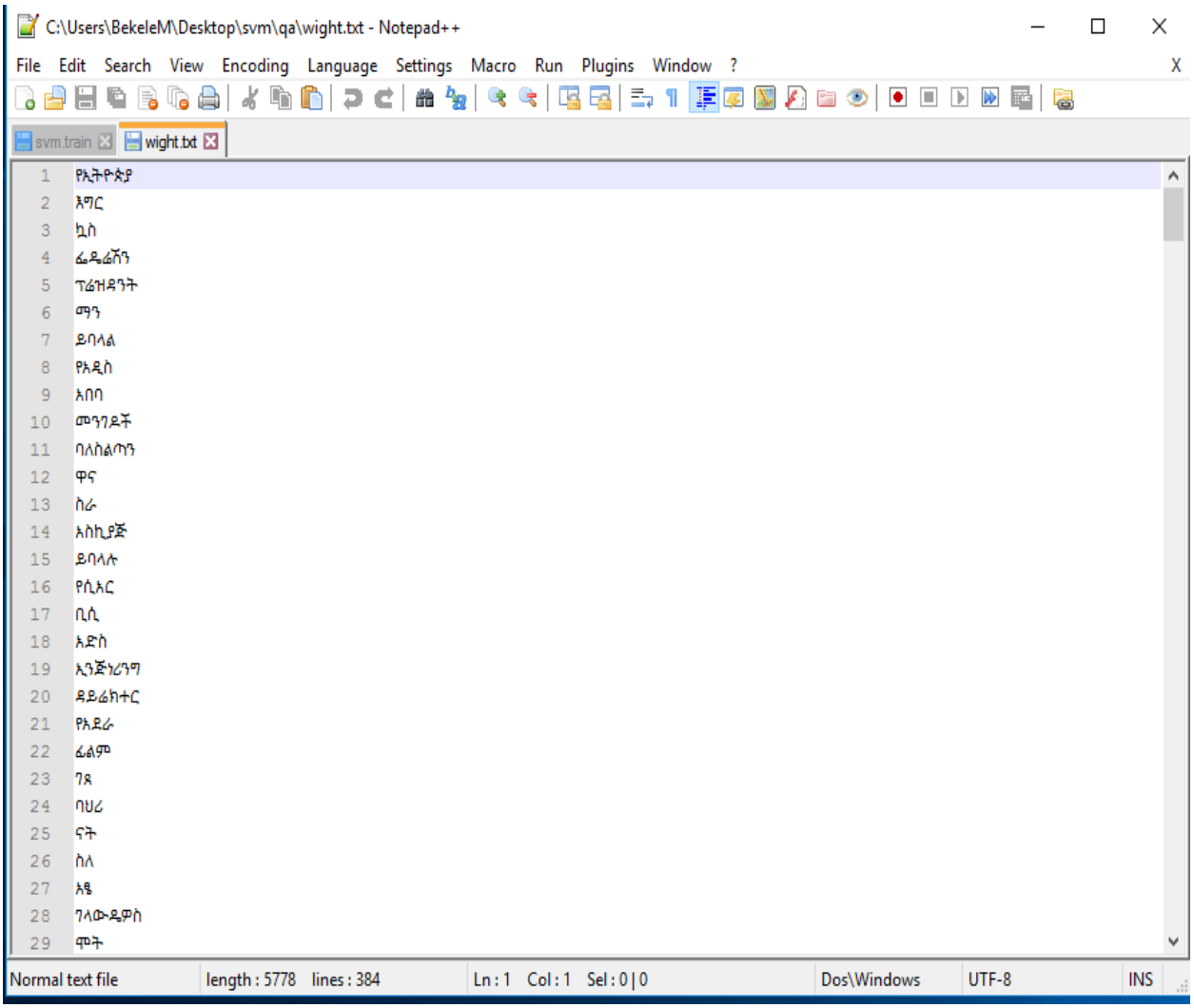
List of Question Used: This appendix shows factoid questions from the two question types person and numerical.

	Questions	Question Types
1	የኢትዮጵያ እግር ኳስ ፌዴሬሽን ፕሬዝዳንት ማን ይባላል ?	Person
2	የአዲስ አበባ መንገዶች ባለስልጣን ዋና ስራ አስኪያጅ ማን ይባላሉ ?	Person
3	የሲአር ቢሲ አድስ ኢንጅነሪንግ ዋና ዳይሬክተር ማን ይባላል ?	Person
4	የአደራ ፊልም ዋና ገጸ ባህሪ ማን ናት ?	Person
5	ስለ አፄ ገላውዴዎስ ሞት በግእዝ የተፃፈውን ወደ አማርኛ የተረጎመው ማን ነው ?	Person
6	የኢትዮጵያ የአየር ትራፊክ ተቆጣጣሪዎች መህበር ያዘጋጀውን ፊልም የመረቁት ማን ናቸው ?	Person
7	የኢትዮጵያ የአየር ትራፊክ ተቆጣጣሪዎች መህበር ፕሬዝዳንት ማን ነው ?	Person
8	በደቡብ ክልል የትምህርት ልማት እቅድ ዝግጅት ክትትልና ግምገማ የስራ ሂደት ባለሙያ ማን ይባላሉ ?	Person
9	የቶፕ ኮንስትራክሽን አመራር አባል ማን ይባላሉ ?	Person
10	የአዲስ አበባ ከተማ አስተዳደር ምክትል ከንቲባ ማን ይባላሉ ?	Person
11	የድሬደዋ ከተማ ከንቲባ ማን ይባላሉ?	Person
12	ከመገናኛ እስከ አያት ያለውን የመንገድ ግንባታ ለማጠናቀቅ የሚያስችል የሁለት መቶ ሃያ አራት ሚሊዮን ብር ስምምነት የተፈራረመው ማን ነው?	Person
13	የአዲስ አበባ ዋና ሥራ አስኪያጅ ማን ናቸው?	Person
14	የኢንቨስትመንት ኤጀንሲ ዳይሬክተር ማን ናቸው?	Person
15	የኦፌዴን ዋና ፀሀፊ ማን ናቸው?	Person
16	የትምህርት ሚኒስቴር የህዝብ ግንኙነት ሀላፊ ማን ናቸው?	Person
17	የአፍዴሀግ ተቃዋሚ ግንባር ሊቀመንበር ማን ናቸው?	Person
19	የቶፕ ኮንስትራክሽን ሀላፊነቱ የተወሰነ የህብረት ማህበር አመራር አባል የሆኑት ማን ናቸው?	Person
20	የደቡብ ክልል የትምህርት ልማት እቅድ ዝግጅት ክትትልና ግምገማ የስራ ሂደት ባለሙያ ማን ይባላሉ?	Person
21	በሶማሌ ክልል ደገሀቡር ዞን የጋሻሞ ወረዳ አስተዳዳሪ ማን ይባላሉ?	Person
22	የሾላ ምግብ ዝግጅት ባልትና ማህበር ሊቀመንበር ማን ይባላሉ?	Person
23	የጅንካ ከተማ ከንቲባ ስማቸው ማን ይባላል?	Person
24	የደሴ ከተማ ፖሊስ መምሪያ የህብረተሰብ አቀፍ ወንጀል መከላከል ዋና የስራ ሂደት ባለቤት ማን ነው?	Person
25	ንብ ኢንሹራንስ ካሳ የከፈለው ለማን ነው?	Person
26	የንብ ኢንሹራንስ ቦርድ ሰብሳቢ ማን ይባላሉ?	Person
27	ከ51 ነጥብ አምስት ቢሊዮን ብር በላይ ካፒታል ላስመዘገቡ ፕሮጀክቶች ፈቃድ የሰጠው ማን ነው?	Person
28	የኢትዮጵያ ኢንቨስትመንት ኤጀንሲ ፕሮሞሽንና የህዝብ ግንኙነት መምሪያ ዳይሬክተር ማን ይባላሉ?	Person
29	ለምርጫ ዴሞክራሲ የለም ያለው ማን ነው?	Person
30	የኢትዮጵያ ስኳር ኤጀንሲ ማን ይባላል?	Person
31	በንግድና ኢንዱስትሪ ሚኒስቴር የማስታወቂያና የህዝብ ግንኙነት ጽህፈት ቤት ሀላፊ ማን ነው?	Person
32	የ3ኤም ኢንጅነሪንግ ኮንስትራክሽን ፒኤልሲ ዋና ስራ አስኪያጅ ማን ይባላል?	Person
33	ፍሬሰላም የወተትና ወተት ተዋጽኦ ማህበር ሊቀመንበር ማን ይባላሉ?	Person
34	የሶማሊያ ፕሬዝዳንት ማን ይባላሉ?	Person
35	የኮሪያ አለም አቀፍ የልማት ትብብር ኤጀንሲ ዋና ተወካይ ማን ይባላሉ?	Person
36	የህብረት ባንክ ፕሬዝዳንት ማን ይባላል?	Person
37	ፕሬዝዳንት ዶክተር አሸብር ለስንት ደቂቃ ንግግር እንዲያደርጉ ተጋበዙ?	Numerical

39	የዶክተር አሸብርን የመልቀቂያ ንግግር ተከትሎ የዶክተሩ የውሳኔ እርምጃ ያልጠበቁት መሆኑን የገለጹት ከፊፋ የተወከሉት ሰዎች ስንት ናቸው?	Numerical
41	የአሜሪካ አለም አቀፍ የልማት ድርጅት ያቀረበው አስቸኳይ ጊዜ እርዳታ ምን ያህል ነው?	Numerical
42	ደረጀ ኤቢሳ እድሜው ስንት ነው?	Numerical
43	ቶፕ ኮንስትራክሽን ለስንት ስራ አጥ ወጣቶች የስራ እድል መክፈት ቻለ?	Numerical
44	ቶፕ ኮንስትራክሽን ስንት አባላት አሉት?	Numerical
45	ባለፉት ሁለት አመታት የነበረው መምህር ተማሪ ጥምረት ስንት ነበር?	Numerical
46	ባለፉት ዘጠኝ ወራት ስንት የአንደኛ ደረጃ መጻህፍት ታተሙ?	Numerical
47	የኢትዮጵያ የአየር ትራፊክ ተቆጣጣሪዎች መህበር ስንተኛ አመቱን ነው ያከበረው?	Numerical
48	ስለ አፄ ገላውዴዎስ ሞት በግእዝ የተፃፈው በስንተኛው ክፍለ ዘመን ነው?	Numerical
49	ዶክተር አሸብር ወልደ ጊዮርጊስ ስልጣናቸውን ሲለቁ ለምን ያህል ደቂቃ ንግግር አደረጉ?	Numerical
50	በፊፋ ፍኖተ ካርታ መሰረት ዶክተር አሸብር ይቀጥሉ አይቀጠሉ የሚለው ሞሽን አንቀፅ ስንትን መሰረት ያደረገ ነው?	Numerical
51	ግንቦት 8 2001 በተካሄደው የጠቅላላ ጉባዔ ከፊፋ የመጡት ተወካዮች ስንት ናቸው?	Numerical
52	የአሜሪካ አለም አቀፍ የልማት ድርጅት ለጥያቄው ምን ያህል ግምት ያለው እርዳታ አቀረበ?	Numerical
53	የአሜሪካ አለም አቀፍ የልማት ድርጅት ምን ያህል የምግብ እርዳታ ሰጠ?	Numerical
54	ሲአር ቢሲ የሚገነባው መንገድ ርዝመቱ ምን ያህል ነው?	Numerical
55	ሲአር ቢሲ የሚገነባው መንገድ ስፋቱ ምን ያህል ነው?	Numerical
56	ለብሔራዊ ፈተና ስንት የመፈተኛ ጣቢያዎች አሉ?	Numerical
57	ለስንት የውጭና የሀገር ውስጥ ኢንቨስትመንት ፕሮጀክቶች ፈቃድ ሰጠ?	Numerical
58	በየአመቱ ስንት ቤቶች ለመገንባት እቅድ ወጥቶአል?	Numerical
59	ስንት ፓርቲዎች ህጋዊ እውቅና አግኝተዋል?	Numerical
60	ስማቸው በምርጫ ቦርድ የሰፈረው ፓርቲዎች ስንት ናቸው?	Numerical
61	የቶፕ ኮንስትራክሽን ሀላፊነቱ የተወሰነ የህብረት ማህበር የአባላት ቁጥር ስንት ነው?	Numerical
62	የደቡብ ክልል ትምህርት ቢሮ ምን ያህል የ1ኛ ደረጃ ተማሪዎች መጽሀፍ አሳተመ?	Numerical
63	የኢሳቅ ጎሳ አባላት አካባቢያቸውን ለስንት ሰአታት ይጠብቃሉ?	Numerical
64	የጋሻሞ ወረዳ ከሶማሌ ላንድ በስንት ኪሎ ሜትር ትርቃለች?	Numerical
65	የሾላ ምግብ ዝግጅት ባልትና አባላት ቁጥር ስንት ነው?	Numerical
66	የሾላ ምግብ ዝግጅት ባልትና የደረቅ እንጀራ አቅርቦት በቀን ስንት ነው?	Numerical
67	የሾላ ምግብ ዝግጅት ባልትና የአባላት የወር ገቢ ስንት ነው?	Numerical
68	አንበሳ የከተማ አውቶቡስ ምን ያህል አውቶቡሶችን ጠገነ?	Numerical
69	የአንበሳ አውቶቡስ አድሱን አሰራር የጀመረው በስንት መስመር ነው?	Numerical
70	ከቄራ የሚነሳው አውቶቡስ ስንት ቁጥር ነው?	Numerical
71	ከቄራ አውቶቡስ ተራ ድረስ ስንት ሳንቲም ያስከፍላል?	Numerical
72	አንበሳ አውቶቡስ በአሁኑ ሰአት በምን ያህል አውቶቡሶች እየሰራ ነው?	Numerical
73	የዘንድሮ ብሄራዊ ፈተናዎች በአጠቃላይ ስንት የመፈተኛ ጣቢያዎች አሉት?	Numerical
74	በአዲስ አበባ ውስጥ በአመት ከስንት በላይ ቤቶች ለመገንባት ታቅዷል?	Numerical
75	በአሁኑ ወቅት ያለው የአንድ ኪሎ ስኳር ዋጋ ስንት ነው?	Numerical
76	የኢትዮጵያ ስኳር ኤጀንሲ ያስቀመጠው የመነሻ እና የመድረሻ ዋጋ በየስንት ሳምንቱ ይቀያየራል?	Numerical
77	በበጀት አመቱ የዘጠኝ ወራት ውስጥ ስንት ፕሮጀክቶች የኢንቨስትመንት ፈቃድ ወስደዋል?	Numerical
78	ፍሬሰላም የወተትና ወተት ተዋጽኦ ማህበር ስንት ላሞች ገዛ?	Numerical
79	ፍሬሰላም የወተትና ወተት ተዋጽኦ ማህበር አባላት ገቢ ስንት ነው?	Numerical
80	የአለም ባንክ ስራ አስፈጻሚዎች ለኢትዮጵያ ምን ያህል ገንዘብ ለመስጠ አስበዋል?	Numerical
81	ህብረት ባንክ በተያዘው የበጀት አመት ስንት ዶላር አገኘ?	Numerical
82	የብሮድ ባንድ ኢንተርኔት አገልግሎት የመመዝገቢያ ክፍያን በስንት አሻሻለ?	Numerical
84	ሺሻ ሲያጨስ የተገኘ ግለሰብ ምን ያህል ብር ይቀጣል?	Numerical
85	የደሴ ከተማ ፖሊስ ስንት ቤት ፈትሿል?	Numerical
86	አትሌት መሰረት ደፋር የሴቶች የአምስት ሺህ ሜትር የቤት ውስጥ ውድድር በምን ያክል ሰዓት አሸነፈች?	Numerical
87	ቴዲ አፍሮ ለምን ያክል ጊዜ በስር ቤት ይቆያል?	Numerical

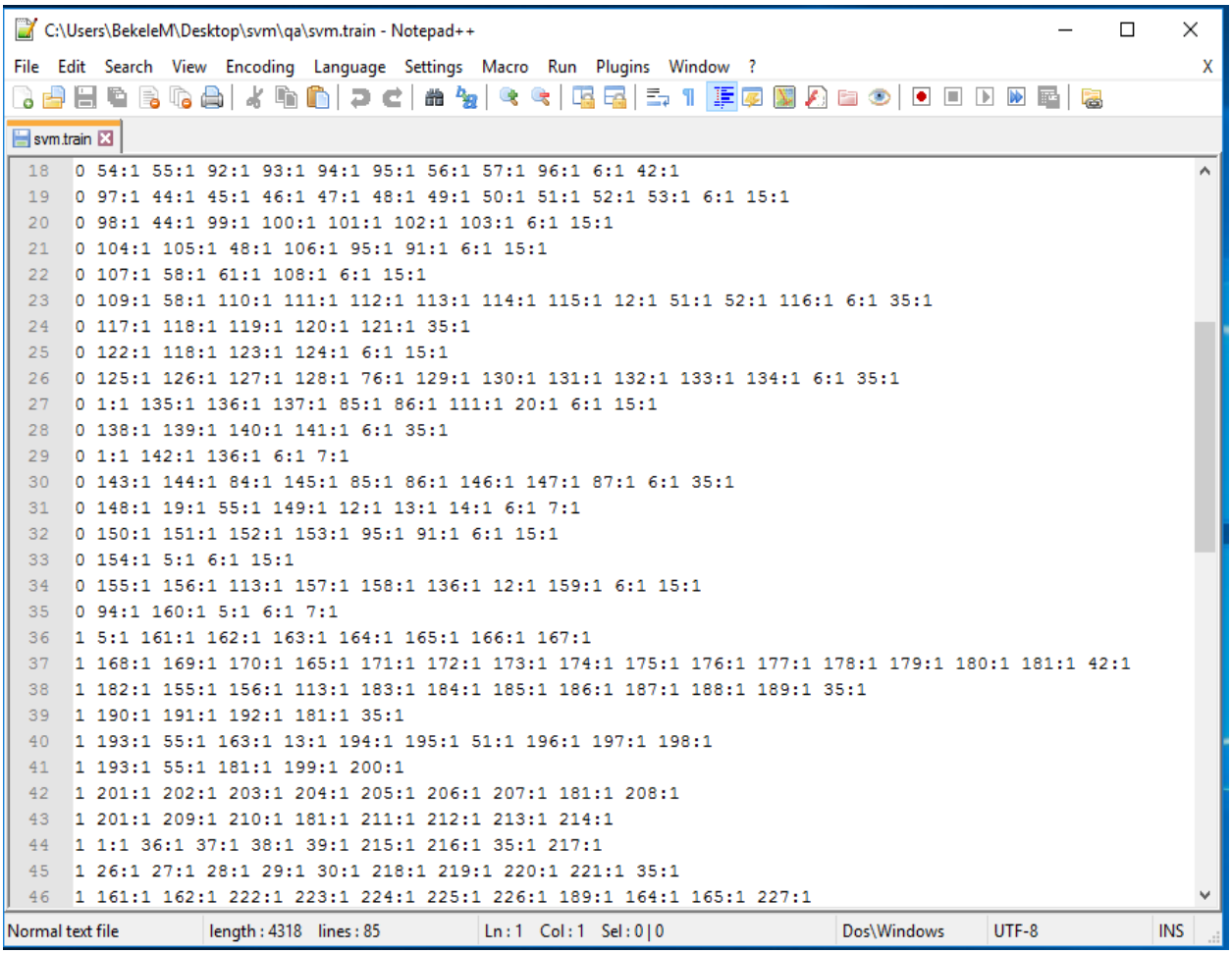
Appendix IV

Vocabulary list used in Automatic question classification: A vocabulary for SVM question classifier in training phase for the purpose to give the Wight of each term and index number for a specific question.



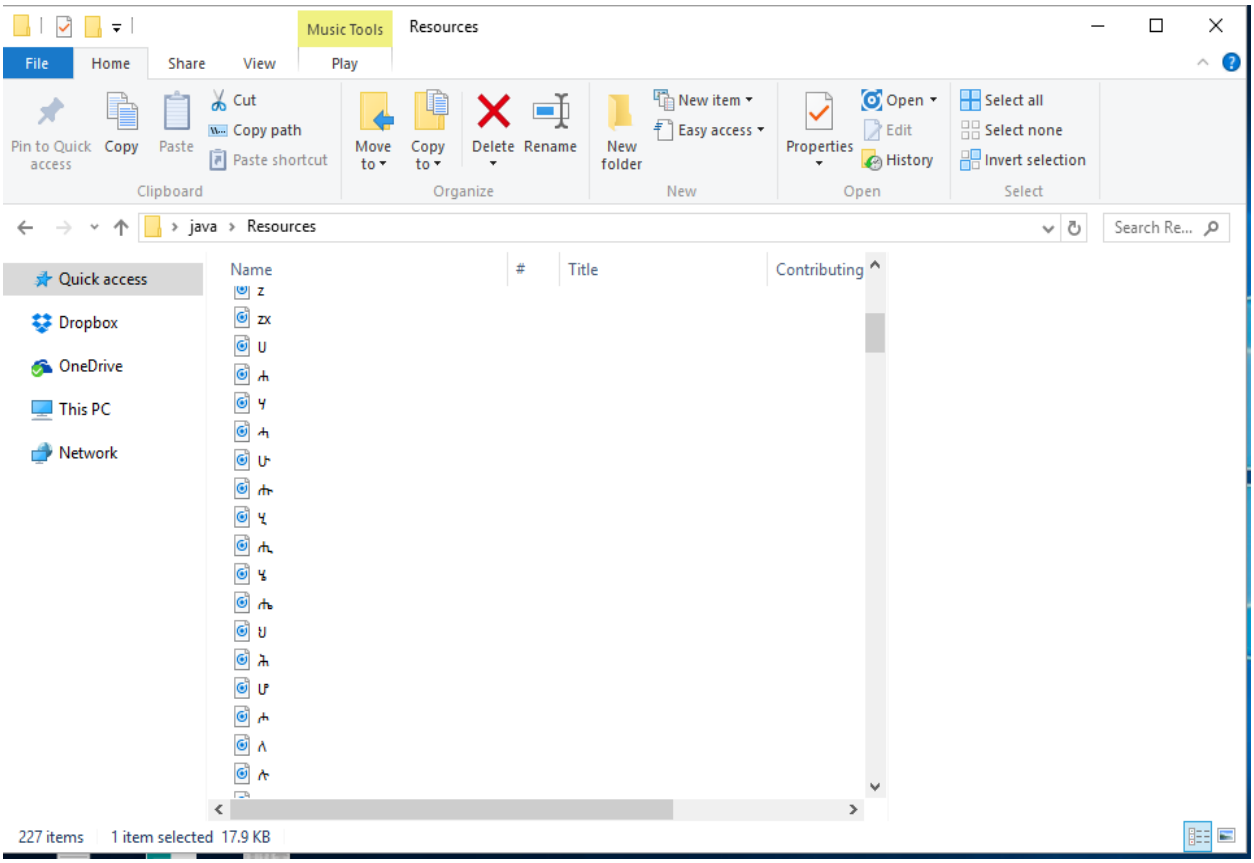
Appendix V

Automatic question classification in the training phase. There are two classes the first person and the second one is a number, the libSVM format is 0 and 1 for respectively.



Appendix VI

Amharic phoneme audio files.



APPENDIX VII

Questionnaire for user acceptance testing:

Questionnaire to test and validate the performance of the speech based system on factoid QA system to providing information in the form of audio for visually impaired users.

1. Using an automatic question classification for SBAQA system enable to better inform more quickly.

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
2. Do you find an automatic question classification for SBAQA system useful for receiving knowledge

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
3. Using an automatic question classification for SBAQA system would improve access to question answering service performance

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
4. It would be easy to get an automatic question classification for SBAQA system to do what I want to get answer

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
5. It is easy to use an automatic question classification for SBAQA system with training

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
6. Do you like the idea of using an automatic question classification for SBAQA system

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
7. will you use an automatic question classification for SBAQA system in the future

   1. Disagree 2.Nutrual 3.Agree 4.strong Agree
8. What issues are covered by an automatic question classification for SBAQA retrieval of the system?
9. Do you feel any uncovered areas by the prototype about an automatic question classification for SBAQA retrieval system? If you feel please state them.
10. Does the system has any significance in the information retrieval?
11. What is the strength of an automatic question classification for SBAQA system?
12. What are the limitations an automatic question classification for SBAQA system?