ADDIS ABABA UNIVERSITY COLLEGE OF

NATURAL AND COMPUTITIONAL SCIENCE

SCHOOL OF INFORMATION SCIENCE

Automatic Amharic Factual Question Generation From Historic Text Using Rule Based Approach

By

Getaneh Damtie

June 2021

ADDIS ABABA UNIVERSITY COLLEGE OF

NATURAL AND COMPUTITIONAL SCIENCE

SCHOOL OF INFORMATION SCIENCE

Automatic Amharic Factual Question Generation From Historic Text Using Rule Based Approach

Getaneh Damtie

A Thesis Submitted to the School of Information Science of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Masters of Science in Information Science and System (Information Science Specialization)

June 2021

ADDIS ABABA UNIVERSITY COLLEGE OF

NATURAL AND COMPUTITIONAL SCIENCE

SCHOOL OF INFORMATION SCIENCE

Automatic Amharic Factual Question Generation From Historic Text Using Rule Based Approach

Getaneh Damtie

Name and signature of members of the examining Board

Name	Title	Signature	Date
Dereje Teferi (PhD)	Advisor	____________	____________
Martha Yifru (PhD)	Examiner	____________	____________
Solomon Teferra (PhD)	Examiner	____________	____________

Declaration

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

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

Signature: ________________________

Getaneh Damtie

This thesis has been submitted

Advisor's Signature: ________________________

Dr. Dereje Teferie (PhD)

Acknowledgment

First and foremost, I would like to thank **God**. St. **Mary**, St. **Michael**, St. **Gebrael** and St. **Ourael** thank you for your blessing and giving me the courage and wisdom to accomplish this thesis.

I would also like to thank my advisor, **Dr. Dereje Teferi,** for his continual support, encouragement, and patience. I greatly appreciate your support from title selection to the end. Thank you all the time.

There are no words to adequately express my gratitude to my wife **Simegn Alemneh**, for her unlimited love, support, sacrifice, and inspiration throughout my entire life.

Finally, my deepest gratitude goes to my **colleagues**, **classmates**, and **friends** who helped me to finish my research work

Table of Contents

Title **Page**

Acknowledgment.. i

List of Figure vi

List of Tables vii

List of Acronyms viii

Abstract.. ix

Chapter One 1

Introduction.. 1

1.1. Background 1

1.2. Statement of the problem 3

1.3. Research Questions.. 4

1.4. Objectives.. 4

1.4.1. General objective.. 4

1.4.2. Specific objectives 4

1.5. Significance of the study 5

1.6. Scope and Limitation of the study 5

1.7. Layout of the Thesis.. 5

Chapter Two.. 7

Literature Review 7

2.1. Natural language processing.. 7

2.2. Questions.. 8

2.3. Types of Questions 9

2.4. Automatic question generation 11

2.4.1. Sources of automatic question generation.. 12

2.4.2. Architecture of automatic question generation.. 13

2.4.2.1. Marking unmovable phrases.. 14

2.4.2.2. Generating Possible Question Phrases 14

2.4.2.3. Removing Answers and Inserting Question Phrases.. 15

2.4.2.4. Post-processing 15

2.5. Approaches to automatic question generation 16

2.5.1. Rule based approach ... 16
2.5.2. Example Based Approach ... 17
2.5.3. Machine learning approach ... 17
2.6. Automatic question generation in education ... 19
2.7. Pre-Processing Tools ... 19
2.7.1. Part-of-Speech (PoS) Tagging ... 19
2.7.2. Named entity recognition (NER) system ... 20
2.8. Evaluation ... 20
2.9. Related works ... 21
2.9.1. Automatic question Generation in Amharic language ... 21
2.9.2. Automatic question Generation in English language ... 22
2.9.3. Automatic question Generation in Chinese language ... 23
2.9.4. Automatic question generation in Portuguese language ... 24
2.9.5. Automatic question generation in Punjabi language ... 24
2.9.6. Summary ... 29
Chapter Three ... 30
Amharic language ... 30
3.1. Background ... 30
3.2. Amharic word classes ... 30
3.3. Amharic punctuations and numerals ... 31
3.4. Amharic Sentence ... 32
3.4.1. Amharic sentence function ... 32
3.4.2. Amharic Sentence Structure ... 33
3.5. Amharic interrogative sentence ... 34
Chapter Four ... 37
Research Methodology ... 37
4.1. Overview ... 37
4.2. Research Methodology ... 37
4.2.1. Problem Identification ... 39
4.2.2. Defining of objectives for a solution ... 39
4.2.3. Design and Development ... 39

4.2.4. Demonstration ... 40
4.2.4.1. POS Development ... 41
4.2.4.2. NER Development ... 41
4.2.4.3. Rule Definition ... 42
4.2.4.4. Content Selection ... 43
4.2.4.5. Prototype Development ... 44
4.2.5. Evaluation ... 44
4.2.5.1. POS and NER tagger ... 44
4.2.5.2. Content selection ... 45
4.2.5.3. Question evaluation ... 45
4.2.6. Communication ... 46
Chapter Five ... 47
Design of automatic Amharic question generation system ... 47
5.1 System design and development ... 47
5.1.1 System architecture ... 47
5.1.2 Dataset preparation ... 48
5.1.3 Preprocessing ... 48
5.1.4 Informative sentence selection ... 51
5.1.5 Question Transducer ... 52
5.1.5.1 Mark Unmovable phrase ... 52
5.1.5.2 Generate possible question phrase ... 53
5.1.5.3 Transformation ... 53
5.1.5.4 Post-processing ... 55
Chapter Six ... 56
Experiment and Evaluation ... 56
6.1. Experimental Environment ... 56
6.2. Metrics of evaluation ... 56
6.3. POS tagger evaluation ... 56
6.4. NER evaluation ... 57
6.5. Content selection evaluation ... 58
6.6. Questions evaluation ... 59

6.7. Findings 61
Chapter Seven 64
Conclusion and Recommendation 64
7.1. Conclusion 64
7.2. Recommendation 65
References 67
Appendix 1: Amharic Alphabets 74
Appendix 2: Amharic Punctuation Marks 75
Appendix 3: Amharic Numerals 75
Appendix 4: Amharic interchangeable characters and their normal character 76
Appendix 5: Sample text and possible questions generated from the text (Text 1 and possible generated questions 1) 77
Appendix 6: sample preprocessing python code 79
Appendix 7: sample question generation python code 81
Appendix 8: Automatic Amharic Question Generation System Interface 83

List of Figure

Title **Page**

Figure 2.1: Types of question generation sources……………………………………………11

Figure 2.2: Architecture of Automatic Question generation system [1]………………………12

Figure 4.1 DSRM Process Mode adopted from [2]……………………………………………38

Figure 5.1 Architecture of Automatic question generation [1]…………………………………48

Figure 5.2 Question Generation flow chat from input sentence…………………………………55

List of Tables

Title **Page**

Table 2.1: WH phrases generated from a given answer phrase…………….………………..…..13

Table 2.2 below summarizes the survey presented above ………………………………………28

Table 4.1 rules used to generate questions ……………………………………..…………….43

Table 6.1 Sample sentence selection ……………………………..………………………… 61

Table 6.2 sentence selection evaluation……………………………………..…….….…….62

Table 6.3 Accuracy Evaluation Result……..………………………...……………………….63

List of Acronyms

AAQG	**Automatic Amharic Question Generation**
AAFQG	**Automatic Amharic Factual Question Generation**
HMM	**Hidden Markov Model**
QG	**Question Generation**
PoS	**Part Of Speech**
ANER	**Amharic Named Entity Recognition**
ML	**Machine Learning**
NLP	**Natural Language Processing**
DS	**Design Science**
DSR	**Design Science Research**
DSRM	**Design Science Research Methodology**
NER	**Named Entity Recognition**
WIC	**Walta Information Center**
NLG	**Natural Language Generation**
NLU	**Natural Language Understanding**

Abstract

Nowadays, due to the availability of digital devices, important educational materials in a variety of languages have become available. However, these texts do not have sufficient amount of practical questions and assessments. Manually preparing meaningful and relevant questions from such materials is a time-consuming and difficult endeavor that necessitates expertise, experience, and resources. This research addresses the problem by automatically generating questions from Amharic texts, with a particular focus on automating the construction of factual questions from text. The automatic Amharic factual question generation systems, which is developed in this research, takes a historical text as input and produces a set of possible questions as output. Historical texts contain various named entities such as names of persons, locations name, cities name, countries name, dates and other entities, which helps to generate many questions.

The methodology used in this study is design science. It has six main activities namely, problem identification and motivation, defining objectives, design and development, demonstration, evaluation and communication. The current research used Part of Speech (PoS) tagger and Named Entity Recognition (NER). The PoS aids in the development of NER. The NER was also utilized to identify answer keywords and generate probable question phrases. In addition, informative sentence selection is used to select informative sentences from the text based on NER and using a certain rules. Transformation rules are used to construct questions from sentences. A prototype is developed using python. Human-evaluator is used to evaluate the question generation system.

The experimental results showed 86.4% accuracy for PoS tagger, 82.0% accuracy for NER and 95.3% accuracy for relevant sentence selection. The experimental results of each question type got 94.1% accuracy for “ስንት” (how much/many), 91.6% accuracy for “ማን” (who), 83.3% accuracy for “መቼ” (When) and 73.0% accuracy for “የት” (where). The overall question generation system come up with 84.6% of accuracy. This shows that the system has high accuracy in question type “ስንት” (how much/many) and needs some improvement in question type “የት” (where).

The system gives a good results for some question types. Accordingly, it is concluded that the system gives a good accuracy for a good coverage of domain specific datasets and also defining more rules by adding more word classes.

For future works, forming new rules to improve the existing rules by adding more word classes, handling exceptions, preparing more domain specific training datasets, preparing common automatic question generation architecture and evaluation techniques are recommended.

Keywords: automatic question generation, factual questions, natural language generation.

Chapter One

Introduction

1.1. Background

Questions are used to express informational needs. When we do not know something, the natural thing which we do is to ask about it. Questions are a teaching tool for learners' and preceptors to assess students' knowledge, promote comprehension, and stimulate critical thinking. Well-formed questions lead to new insights, generate discussion, and promote the comprehensive exploration of a subject matter. Despite its usefulness, manually creating meaningful and relevant questions is a time-consuming and challenging task and also requires training, experience, and resources. As computer systems are becoming more advanced and self-directed, informational needs are growing day by day and such computer systems which have the ability to ask questions can have many obvious advantages.

Automatic question generation (AQG) is an important yet challenging problem in natural language processing (NLP). It is the task of generating syntactically sound, semantically correct and relevant questions from various input formats such as text, structured database or knowledge base [3]. Question Generation (QG) from text is a Natural Language Generation (NLG) task concerned with generating questions from unstructured text. NLG is the task of translating machine-readable, nonlinguistic information into an equivalent human language representation. NLP also includes Natural Language Understanding (NLU), which is the inverse of NLG. A natural language understanding (NLU) system attempts to transform human language into a machine-understandable representation. NLU and NLG are both subset of NLP which contains tools for QG [4].

Approaches of AQG tasks are different depending the type of question generation system. One group [5] uses natural language processing techniques such as syntax processing, part of speech (PoS), named entity recognition (NER), and sentence transformation using rules (rule based approach) while other groups may use patterns stored in database [6] or interactive (machine learning approach) [7]. For under resourced and morphologically rich languages, rule based approach is suitable [8]. Amharic is one of the morphologically rich and under-resourced languages [9]. The current thesis used a rule-based approach because there are no available training datasets for machine learning.

Questions can be classified into different categories according to the source of questions, length of expected answer and complexity [10]. For example, in terms of target complexity, QG can be classified into shallow and deep QG. Shallow QG focuses mainly on generating facts such as who, what, when, where, which and how many/much questions whereas Deep QG involves mainly generating logical thinking such as why, why not, what-if, what-if-not and how questions [1]. This research focuses on the generation of shallow questions.

The generated questions can be factual, gap-fill, multiple choice etc. based on their response formats. Factual questions are questions transformed from declarative sentences into questions that have short, specific and concrete answers. They are very important in reading comprehensions. Multiple choice questions (MCQ) is composed of multiple distractor and one correct answer with a wh-type question. While, gap-fill questions are questions that consist of one or more blanks in a sentence/paragraph.

Specifically, this thesis focuses on automatically generating Amharic factual WH questions from historical texts. Factual questions are an important part of assessments, because they assess learners' comprehensive storage of declarative or factual knowledge, which is essential for learning [11]. Factual questions are useful for elementary school students to learn [5]. Typical examples of factual questions are of type who?, what?, where?, and when?. In addition [12], the historical texts main properties is that they contain dates, names of persons, names of locations etc. Therefore, historical texts contain a lot of questions which are required to test the knowledge of a person.

Amharic is the most spoken Semitic language next to Arabic and spoken by more than 29.3% of the country's total population (21.6 million) as a mother tongue. In rural or urban distribution 25.7% of rural and 48% of urban population speak Amharic as mother tongue. Considerably 2.7 million of population from given total urban population can expect to speak Amharic as second language as 2007 [12]. It is the official language of Ethiopia. The morphological richness of Amharic is a difficulty to natural language processing in general [13].

Although Amharic belongs to under resourced languages in the linguistics field and NLP, researchers have done some baseline NLP research on Amharic language. Some of them include, interactive question answering [14] [15], PoS tagging [16], and named entity recognition [17]. The current NLP research aims to conduct further research on Amharic language. Hence, this research attempts to develop a system to automatically generate Amharic Factoid Questions from text using rule based approach.

The generated questions could then be presented to Amharic language teachers, who could select and revise the ones she judges to be useful.

1.2. Statement of the problem

Nowadays due to the availability of technology, it has become easier to access and share texts with important educational values. However, these texts introduce challenges, such as lack of fresh and sufficient amount of practical exercises [18]. It is also a fact that manually writing good questions is a challenging and time-consuming task. Also, recently Corona virus (COVID 19) has become a global challenge. This led to schools being closed and students are forced to learn at home without teacher's in-person assistance. This creates a challenge for students to get sufficient questions.

Amharic is the official working language in Ethiopia. It is Ethiopia's second most widely spoken language and the world's second-most widely spoken Semitic language [19]. However, it belongs to under resourced languages in the field of linguistics and natural language processing (NLP). Since Amharic is a morphologically rich language, it poses a challenge to natural language processing in general [13].

Different researches used different approaches such as rule based and machine learning. To develop question generation systems using machine learning approach, it requires more question-answer/ sentence-question paired datasets than what we have used (300 simple sentences). According to Kumar et al [20], machine learning approaches need large number of question-answer pair training dataset. In Amharic there are no such large available datasets. In addition, studies like Miroslav et al [8], a rule-based approach is appropriate for under resourced languages.

There are various researches to generate factual questions from different sources automatically, mainly for well-resourced languages such as English. For Amharic, some authors conducted researches on interactive factual question answering (QA). Nevertheless, no research has been conducted for automatic Amharic factual question generation (AQG) that generates factual questions from a given Amharic text, which is an important element of learning in and educational environments. Therefore, this research intends to investigate question generation approaches to design and develop an automatic Amharic factual question generation system using rule based approach.

1.3. Research Questions

This thesis address the following research questions:

1. What feature of Amharic sentences can be used to define rules to generate Amharic factual questions?
2. To what extent can the development of PoS and NER be used to gain better performance for automatic question generation systems?

1.4. Objectives

1.4.1. General objective

The general objective of the study is to investigate the development of an automatic Amharic factual question generation system.

1.4.2. Specific objectives

In order to accomplish the general objective, the following specific objectives are drawn to:

- Conduct critical literature review on the concept of automatic question generation from text.
- Segment words which contains prepositions and conjunction as prefixes and suffixes in the word.
- Conduct informative sentence selection from which question can be generated.
- Design and Implement a prototype for the newly developed automatic Amharic factual question generation system.

- Evaluate the question generation system
- Report the result and make recommendations for future research

1.5. Significance of the study

The current research has a great significance to be a baseline for other researchers working on automatic Amharic factual questions generation and for others who work on a complete automatic Amharic Exercise generation that integrates factoid and other non-factoid questions. Furthermore, if the results of this research are likely to be implemented in the future, it can be used in different areas to get precise information to formulate automatic Amharic question generation, in which it will alleviate the problem of getting precise information in different fields. It can be applied in academic areas such as primary and secondary schools.

1.6. Scope and Limitation of the study

This study focuses to investigate and design automatic Amharic question generation. It mainly focused on factual question from Amharic historical input text because historical texts contain many questions which are required to test the knowledge of the students. In addition, Factual questions are focused more on facts such as such as “ማን (who)?”, “ስንት (how many/much)”, “መቼ (when)?” and ”የት (where)?” questions which is important in elementary and high school students to know basic facts. In this study, it does not include generating more than one question from one sentence at a time.

This research is limited on transformation rules and simple sentences to construct questions using NER. Transforming named entities and question phrase may not be always generate correct questions. Since Amharic sentences formed in different ways. It does not cover ambiguity resolution, complex sentence simplification and anaphora resolution.

1.7. Layout of the Thesis

The rest of the thesis is organized as follows:

Chapter 2 is literature review. The chapter starts with an overview on AQG and Natural Language Processing. Also, AAQG is discussed at large, with specifics on approaches used in AAFQG and the architecture of AAQG systems.

Chapter 3 presents about Amharic language. The languages word class, types of sentences based on function and structure has been presented.

Chapter 4 illustrates the methodology that is chosen to conduct this research. The chapter discusses the way corpus is developed and the tools used to build and evaluate this research.

Chapter 5 presents the design of AAFQG system using DSRM approach.

Chapter 6 presents the evaluation results of the proposed system along with its discussion.

Chapter 7 concludes our study with the research findings and future works.

Chapter Two

Literature Review

2.1. Natural language processing

Natural Language Processing (NLP) is an important field of computer science and Linguistics, concerned with making computers understand statements or phrases written in human languages. Today, it is a widely discussed and researched area that concerns with how machines can understand and operate natural language text. The aims of NLP, is to make ease the user's work and to fulfill their desire to converse with computers in natural language. NLP helps to those users who do not have enough time to learn new languages or perfect them, as not all users are well-versed in machine specific language [21].

A language is a set of rules or symbols that are used to communicate. Symbols are combined and utilized to transmit or broadcast information. NLP is divided into two parts: Natural Language Understanding (NLU) and Natural Language Generation (NLG), which evolve the process of comprehending and producing text [22].

NLG is the process of generating meaningful phrases, sentences, and paragraphs from an internal representation. It is a part of NLP that consists of four stages: establishing goals, planning how goals might be attained by assessing the circumstances and accessible communicative sources, and implementing the plans as a text [22].

NLU attempts to grasp the meaning behind the written word. It is quite possible that the same text has numerous interpretations, or different words have the same meaning, or that the meaning changes with the situation. Natural language comprehension, on the other hand, is based on a variety of strategies, including syntax, semantics, and pragmatics [4].

NLP is increasingly becoming an important application area. Some of these applications are, machine translation, PoS tagging, automatic question generation, optical character recognition, automatic text summarization, NER etc. [21]

2.2. Questions

A question asks for information that seeks to clarify, understand, evaluate, or affirm something [23]. Questioning is a basic cognitive process that underpins higher-level cognitive process like comprehension and reasoning. The ability to ask questions is the key cognitive element that distinguishes human and animal cognitive abilities [24]. Asking questions is an important element of gathering information since it prompts an answer, allows for changes in response to findings, enhances comprehension, promotes self-regulation, and it invites conversation. Questioning begins as a conscious response to some outside stimulus. A question arises because some piece of information is lacking or because the stimulus conflicts with existing information; the mind is trying to re-establish equilibrium [23].

A good question is relatively short, clear, and unambiguous. A few qualities of a good question are as follows [24]:

- Evokes the truth
- Asks for an answer on only one dimension. A question that asks for a response on more than one dimension will not provide the information you are seeking.
- Has mutually exclusive options. A good question leaves no ambiguity in the mind of the respondent. There should be only one correct or appropriate choice for the respondent to make.
- Does not presuppose a certain state of affairs.

One of the most important uses of questions is reflection, improving our understanding of things we have found out. Questions are used to check the information from the existing contents or to extract information from the existing contents. It used from the most elementary stage of learning to original research. Questions are the basic requirement in learning. Students of all ages use questions in their learning of topics, and the skill of having learners creating “investigatable” questions is a central part of inquiry education. The method of questioning student responses may be used by a teacher to lead the student towards the truth without direct instruction, and also helps students to form logical conclusions. Questions have also been used to develop students’ interest in a topic. Another use of questions is to give students a map for self-recognition of reaching milestones of understanding as they study a unit, by asking, when a unit is begun, some difficult things that require understanding and insight into the material to be able to answer [24].

Questions can be organized by the following characteristics: their purpose, the type of information they seek, their sources of information and the length of the expected answer [10].

2.3. Types of Questions

AQG is a method for generating a suitable set of questions from a content, which can be text. It is the task of producing meaningful questions from various input sources such as text, a structured database, or a knowledge base that are syntactically sound and semantically valid. The generated questions can be different types, such as Factual, gap-fill, multiple choice etc.

Factual questions: are questions transformed from declarative sentences into questions that have short, specific and concrete answers. Factual questions are very important in reading comprehensions. Typical examples to generate factual items are who?, what?, when?, how many? and where?. For preparing factual questions, expository text is simpler than narrative text while expository texts include a lot of factual statements like place, time and person [5]. Factual questions are simple questions that are answered with a maximum of one sentence [25].

Factual questions are very important for elementary school students. Teachers prepare questions to ask simple factual questions since students are still learning to read and as a way to stimulate their processing of story reading that can help for a pedagogical assessment of reading skill. One important pedagogical strategy is asking questions which is used by teachers in order to teach their students how to ask questions. The strategy is highly supported for teachers as many studies showed that students have challenges to know their knowledge gap and not ask much questions during instructions [26]. Answering factual questions helps students to know the subject in depth. The creation of factual questions mainly needs syntactic information [27].

In addition, questions are used for many other application areas such as advanced educations (intelligent tutoring systems), question answering, for security context and also in health sector [28]. Human teachers can create either deep questions involving complex inference or shallow factual questions. Automated question generation system are much more likely to be error prone when complex inference is involved than when it is not. In addition, AQG systems can generate a large set of shallow questions very quickly that could help teachers to focus on preparing good deep questions [10].

Heilman [10] developed a system of automatic question generation that can take as input text that is helping materials for text books and produce output of a ranked list of factual questions. Questions are generated from informational texts and also the developed model is more generic in syntactic and lexical related questions within factual information. Learning materials used for education contain a small number of factual question that helps students' to practice exercise and assessments.

Multiple choice questions (MCQ): is composed of multiple distractor and one correct answer with a wh-type question. The preparation of multiple choice questions is a 3-step process: term extraction, selection of distractors and questions building [29].

MCQ complexity vary because of the similarity between the key and distractors. The more similar they are, the more knowledge you need to differentiate between them and choose the right one. Therefore, if we have an adequate measure of similarity for concepts, with predictable difficulty, we attempt to formulate MCQs [30].

One of the promising examples of educational applications of NLP techniques is an automated generation of MCQs. Multiple-choice question exams are commonly used, considered highly useful and are successful in testing the knowledge of students, but manually creating those questions is expensive [31].

MCQ can be generated using rule based approach and also evaluated with different criteria such as grammatical correctness, meaningfulness, appropriate WH word usage etc. [32].

Gap-fill questions: are fill-in-the-blank questions that consist of one or more blanks in a sentence/paragraph. Gap-fill questions play essential role in developing test materials and tutorial dialogues. There can be two types of Gap-fill questions: with alternate choice (key and distractors) and without choices. GFQG consists of 3 modules: Sentence Selection, Gap Selection and Distractor Selection. In sentence selection phase, relevant sentences should be selected from the document. Then from the selected sentences, make blank for the answer keys. Finally, determine the distractors for those answer keys [33].

The appeal of gap-fill questions is that they are perfect for automatic marking since the correct answer is simply the original word that corresponds to the gap in the original sentence. In GFQG, there are no issues about the linguistic aspects (e.g. grammaticality) since the stem is built by only removing a word or a phrase from a text section [34]. In addition, gap-fill questions are successful at diagnosing and evaluating the knowledge of students. The evaluation of GFQG systems can be performed in different ways. One of commonly used evaluating techniques is manual. The three modules (sentence selection, key selection and distractor selection) were manually evaluated [35].

2.4. Automatic question generation

Questions are very important features of learning that are used to extract useful information from the text [36]. Available research has shown that some people are not very competent to ask good questions. Consequently, people can benefit from automated QG system to assist them in fulfilling their need for investigation [37]. Questions are very important component of countless learning experiences, from one-to-one tutoring sessions to detailed reviews in addition to real life discussions, to be straight forward [10]. It is considered to be helpful to use questions in teaching. Preparing learners to create questions helps students to recall knowledge and deepen the material they have learned. Additionally, it may also improve thinking skill. Model questions can be used to help students develop the ability to construct questions and thereby the meta-cognitive abilities of students by deploying automated question generation in educational systems [38].

AQG is the means for constructing a reasonably correct (syntactically, semantically) set of possible and relevant question form different contents, such as text, structured database or a knowledgebase with the help of different NLP techniques. Automatic question generation (QG) is an interesting task but challenging in NLP field [39]. One of the most significant application in NLP is, AQG. It takes text sentence as an input to generate questions. Question generator learners are actively self-regulate their learning [37]. Asking questions is a strong reader's character to grasp basic knowledge or to prepare for competitive exams [40]. Students learn to formulate and respond to question about situations facts and ideas while engaged in understanding a text [21]. AQG can be applied to many different fields such as education, health, development of conversational agents, question answering, machine reading compression system [34]. Some examples include: factual questions, multiple choice questions, true and false questions and gap-fill questions.

AQG can be applied in different areas, such as education, health, security, etc. For example, in education sector, nationwide exam questions can be generated automatically [41]. Since writing questions is challenging and time consuming for question creators while preparing questions from different sources like books or there might not question banks that can be relevant to content studied by students [34]. The potential advantage of generating questions automatically minimize human dependency to prepare questions and other requirements related with systems interacting with natural languages [42].

The number of researches in automatic question generation came to be on the rise in recent years. The main factor is the availability of electronic learning platforms where people can take different courses from home [43]. From "A Systematic Review of Question Generation" reported, 93 articles published on AQG in 4 years from 2015 and 2019 [34].

AQG used tools from both NLU and NLG. In order to generate question, the primary task is understand the language even if the understanding is somewhat shallow [4].

2.4.1. Sources of automatic question generation

Question generation systems takes inputs and produce questions as output. Based on input sources, question generation can be classified in to two types [1] as shown in figure 2.1

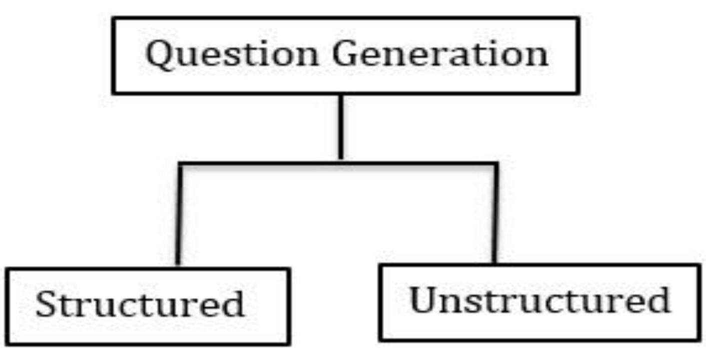

Figure 2.1: Types of question generation sources

Unstructured question generation uses rule based approach for text and documents. In task of question generation from text, a QG application takes an input text like word, set of words, a single sentence, a text and so on. Generating question from text can be decomposed into three subtasks. First, given the source text, a content selection. Second, given a target answer, select an appropriate question type. Third, from given content and question type, the actual question is constructed [25].

Structured databases can also be used in addition to texts as input for question formulation. Sneiders generated question templates, whose answers can be queried from a structured database. For instance, “When does <performer> perform in <place>?” has two entity slots, which represent the relationship (perform perform-place) in conceptual model of the database. Thus, this question template used for this specific relationship. Template based questions are suitable for special purpose question generation systems. However, a lot of human involvement is required in order to develop high-quality templates [44].

In addition to its input source, Question generation can be classified into deep QG and shallow QG based on its depth. Deep QG generates detailed logical questions such as ‘how’ and ‘why’ whereas shallow questions generate more on facts like ‘who’, ‘where’, ‘what’ and ‘when’ questions [1].

2.4.2. Architecture of automatic question generation

The architecture of AQG can be classified into four phases: detecting unmovable phrases, selecting answer keys and generating question phrases, transformation (removing answer keys and inserting possible question phrases) and performing post-processing as shown in figure 1 [1]. After taking preprocessed text (text cleaning, character normalization), the first phase of AQG is marking phrases that are not answer keywords phrases. In the second stage, select answer keywords (named entities such as person, location, organization, dates and numbers) from the sentence and generate possible question phrases (who, where, when, etc.) for each answer keywords. Then questions are constructed by removing answer keywords and insert possible question phrases. The final phase, post-processing phase is responsible to ensure the proper formatting of the final output questions.

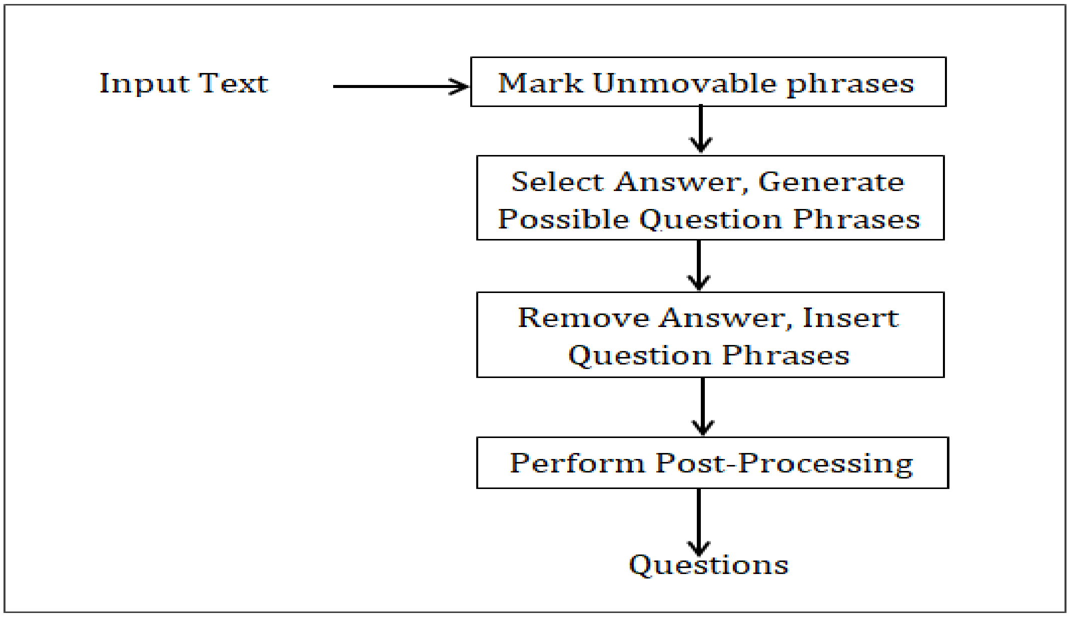

Figure 2.2: Architecture of Automatic Question generation system *[1]*

2.4.2.1. Marking unmovable phrases

In factual AQG task, all phrases in a sentence may not be answer key phrases. In this step, unmovable phrases are detected, which doesn't help to construct questions. NER tool is used to detect this phrases by comparing the values of several lists of named entities with the phrases of input text. If they don't match any of the values in the lists, they're considered Unmovable. This means that the Unmovable phrases are not named entities and might be any other words in the sentence. Since this phrases are skipped over the next steps of answer phrase selection, the system does not generate questions for them.

Marking unmovable phrases usually solved using different methods from the field of NLP such as PoS tagging or named entity recognizing [45].

2.4.2.2. Generating Possible Question Phrases

In QG task, generation of appropriate question type starts by identifying answer key from the input sentence. After selecting answer key phrases the system iterates across the available answer phrases and generate possible questions for each. Identifying domain specific words in the sentence is performed to select answer keys. This stage is about deciding which type of question is created (who, where, when, what etc.) depending on the keys [46].

This research used NER tools, to generate the question phrases based on possible answer keys. The system generates appropriate question phrases to construct a set of one or more possible question phrases for a given answer phrase, each of which is then utilized to construct a final question sentence.

In table 2.2 shows the possible question phrases generated from a given answer phrases.

Table 2.2: WH phrases generated from a given answer phrase.

Question phrase	Answer phrase	Examples
ማን (who)	The answer phrase's word is tagged as person (per)	አፄ ዮሃንስ፣አልማዝ …
የት (where)	The answer phrase's word is tagged as location (loc)	ጅማ፣ባህርዳር …
መቼ (when)	The answer phrase's word is tagged as date (date)	12/12/2019፣ yesterday
ስንት(how much/many)	The answer phrase's word is tagged as number (NUM)	25 አመት፣100 ብር …

2.4.2.3. Removing Answers and Inserting Question Phrases

Actual question should be constructed in grammatically correct natural language expression. The system removes the selected key, and insert possible question phrases generated from the answer phrases using a set of transformation rules. Applications for question generation mainly use transformation-based methods to generate well-formulated questions [47]. While the question generation application uses a set transformation rules for question construction.

The evaluation question generation system has been done manually. Question generated by the system presented to the human expert and he makes judgment the accuracy of the questions [1].

2.4.2.4. Post-processing

To ensure proper formatting and punctuation after creating questions, several additional post-processing mechanisms are required. The question marks have replaced the periods at the end of sentences. The generated questions is also de-tokenized to remove any unnecessary whitespace symbols [10].

2.5. Approaches to automatic question generation

Approaches of AQG tasks are different depending the type of question generation system. Each system uses many approaches and while some sort of evaluation is conducted, it is not possible to say which methodology is better because the system operates in various environments, use different source texts form which the questions are created, generate different types of questions, have different target groups and use different metrics of evaluation. Some QG systems, for instance, attempt to test vocabulary [48], or comprehension of reading [47] while others, concentrate on factual texts [10] or Math [49].

Based on the methods used, approaches of QG are categorized into different groups. One group use natural language processing techniques such as syntax processing, PoS, NER and sentence transformation using rules [5] while other groups may use patterns stored in database [6] or interactive (machine learning approach) [7].

2.5.1. Rule based approach

In rule based approach, rules should be developed based on language features to generate questions automatically from the input text [50]. Questions are generated by developing handcrafted rules using the key entities from a given text such as PERSON-NAME, DATE-FORMAT, LOCATION-NAME, NUMBER etc. Kaur et al developed ten different rules to generate questions: Location rules, Name rules, City rules, Year rules, Numeral rules, Measurement rules and Direction rules [51]. However rules might be different depending on the language features [52]

The rule based approach is necessary if there is lack of training data and resources. Many NLP researches have been successfully developed using this approach. For less-resourced and morphologically rich languages, the rule based approach is suitable [53]. Amharic is one of the morphologically rich and under-resourced languages [9]. For this research we used rule based approach.

2.5.2. Example Based Approach

This method is often referred to as the pattern matching approach, with a series of patterns stored in the database along with possible questions [6]. The system compares the pattern of the input to the patterns stored in the database when the input is supplied to the system. If the pattern matches the pattern stored in the database, it will produce possible set of questions using the available stored questions [1]. For example if a pattern stored < who is the president of Ethiopia > and the input text is <Joe Biden is the president of America>. The system generate questions < who is the president of America > based on stored pattern by replacing Ethiopia by America.

Pattern matching question generation approach mostly uses template based matching to generate questions [54].

Template Based Pattern:- Template based approaches typically operate with a collection of sentence patterns to which the sentences from the input text are allocated. Questions are created by patterns that deals with a predefined collection of possible questions in each pattern. Differences in different methods lie in pattern forms. Some AQG system used template based pattern for factual question generation but have some limitations. For example, if the patterns are more specific (e.g. pattern represented by sequence of PoS tags), fewer sentences are covered by one pattern and the amount of patterns required to cover a similar set of sentences as before is significantly larger. It also need additional knowledge about words in the area of factual questions (e.g. if the word is entity or location). The total number of them increases rapidly as more particular patterns are used. This generates some problems: which pattern to select if more accessible patterns are found or what to do if no pattern is found. Moreover preparing large set of patterns manually is challenging [8]. This method is recommended for special-purpose applications, as it is difficult to create templates for generic topics [55].

2.5.3. Machine learning approach

In machine learning approach, questions are generated by learning and generating model from training data. Miroslav et al [7] developed a framework that transform input text into a structured set of features and uses them to generate factual questions. The learning process is based on a combination of reinforcement learning and supervised learning methods. It learning start with a set of pairs created by declarative sentences and given questions and proceeds to learn how to turn sentences into questions.

There are different machine learning techniques that are for question generation task. The techniques can help for target selection [56], key word selection or in question formation [57]. Some of machine learning techniques used for AQG are Naive Bayes, Hidden Markov Models (HMM), Recurrent Neural Network (RNN), Support Vector Machines (SVM) etc.

Naive Bayes is the simplest and effective algorithm that is often used in text classification applications and experiments [58]. It is recognized as one of the best performing methods for the classification of documents. The basic idea is to estimate, for instance, the parameters of a multinomial generative model, then find the most probable class for a given instance using Bayes rule [59].

Support vector machine is a binary linear classifier, which takes in a set of training data and classifies each member of the set into one of the two groups. The training algorithm constructs a model that can be used to classify items in one category or another from a test data set. An SVM model is a representation of the examples as points in space, mapped such that a simple distance that is as large as possible separates the examples of the different categories [60].

SVM based text classification is the method of classifying information in text format by its content, i.e. by the content that can be transmitted by the words found within it. To be able to identify a large volume of text-based information in a time-critical manner, automating this process is essential. Automated text classification finds widespread use in a number of domains, such as text retrieval, summary, information extraction and query answering, among others, due to the large amount of textual information that needs to be processed [61].

Hidden Markov models (HMM) from the point of view of the mathematical framework, it is very powerful and has formed many theoretical bases of different applications, even if the model is well implemented, it could be used for many applications [62].

HMM has 3 parameters: Start Probability, Transition Probability, and Probability of Emission [63]. Start probability means the probability that in a sentence a tag appears first. Transition probability means the transition probability ratio from the given tag to the next tag, and the probability of the tag happening. Emission Likelihood is the probability ratio of the occurrence of a specific word with a tag and the probability of a tag occurring.

2.6. Automatic question generation in education

Although exam-style questions are a fundamental educational method that serves a number of purposes, constructing questions manually is a complicated process that requires preparation, expertise, and money. In fact, this hinders and slows down the use of educational activities and new developments that involve a wide pool of questions. Automatic question generation (AQG) techniques become an essential element of learning environment and can be implemented to reduce the expenses associated with manual construction of questions and to meet the need for a constant supply of new questions. Due to this fact, there are many researches being conducted for different language and different types of question such as factual, multiple choose, gap-fill, list etc [34].

2.7. Pre-Processing Tools

2.7.1. Part-of-Speech (PoS) Tagging

Part-of-speech (PoS) tagging is a process of assigning tags to the words corresponding to a specific portion of speech in a text. The recognition of terms as nouns, verbs, adjectives etc. is a fundamental version of PoS tagging. Even though there are works for PoS tagging, they are not accessible to the general public. This is the reason to develop PoS tagging for this research.

PoS tagging can be broken down into three categories: rule-based tagging, statistical tagging, and hybrid tagging. Statistical tagging approach is used for this research.

PoS tagger is important for Amharic named entity recognition, due to the absence of capitalization in proper nouns. One of the ambiguities of a word is finding the correct tag for each segmented word [19]. PoS tagger helps the NE recognizer to identify words that are nouns and assign the appropriate tag to each nouns that are named-entities [15]. For this study, PoS tagger helps to build Amharic NER by taking POS tagged corpus and prepare for training data by replacing the nouns, which are named-entities by the appropriate tag. Since NER used mainly to construct questions in this study.

2.7.2. Named entity recognition (NER) system

Named entity recognition (NER) is often applied to identifying the entity and extracting the entity. In text, it locates and classifies named entities into predefined categories such as location, individual names, organization, time expression, quantity, monetary values, percentages, etc. Its purpose is to classify certain named entities. The NER method depends on rules written by hand [64].

For this study, Amharic NER is developed using PoS tagged corpus and manually collected names of person, location and organization. In order to build ANER, replace nouns from PoS tagger which are named-entities (person, location and organization) by per, loc and org and also replace cardinal (CN) and ordinal (ON) by NUM.

NER is a very powerful tool for generating questions from text by defining and classifying Words from text into predetermined classes, such as location names, person names, and other entities [50]. Factual questions need entity names that can be as answer and extracted using NER. This tool makes the process of extracting answer keywords and selecting informative sentences much faster and easier to construct questions. For instance, the sentence <<አበበ በ1955 ዓ.ም ጎንደር ተወለደ ፡፡>> named entities recognized as <<አበበ/PER በ1955/DATE ዓ.ም/NN ጎንደር/LOC ተወለደ/VB ፡፡ /PUNC>>. From this sentence አበበ, 1955 and ጎንደር are named entities that serves as answer keywords and helps to construct questions types ማን,መቼ and የት respectivly.

2.8. Evaluation

The development of question generation system and evaluation methodology should be parallel. A systematic analysis of evaluation methodologies could play a central role in the effort to construct machines capable of achieving linguistic communication standards that are human-like [65].

The evaluation of AQG are based on different criteria, such as user satisfiabilty, linguistic well-foundness, maintainability, cost, effectiveness output quality and flexibility [66]. AQG evaluation uses features like length of answer phrase, grammatical correctness, transformable, semantic, negation and vagueness [55]. There are different AQG evaluation techniques, namely, intrinsic and extrinsic evaluation [67], Black-box, Glass-box evaluation, Manual evaluation uses human judges to evaluate the performance of the generation system performance. It is easy and reasonably easily accessible [55]. Evaluation can be performed by experts, students or comparison between systems generated questions and human generated questions. Expert-based question generation evaluation is the most known evaluation approach, in which a sample of generated items are presented to experts to review. Given that expert review is a standard method for choosing questions for actual test, it is assumed that expert rating is a high quality alternative. The second most frequently used evaluation technique is comparing machine generated questions with human generated questions which is done automatically or as part of expert analysis. The comparison presented to approve different aspect of question quality [34]. Zhang et al [68] assessed their approach by counting the number of popular questions between human-and machine generated questions. Similarly [34], students evaluate questions by review questions or take exams. Experiments should assure that assessments do not affect the grades or motivations of learners.

Montse et al [69] developed automatic question generation systems for Basque language at sentence level. The author evaluate generated questions based on grammatical correctness and appropriateness of the questions created from the sentence as well as their interrogative pronouns. [25] After generating all the questions from the sentence recall-based evaluation has been done. Recall, precision and F-Measure [50] are used as evaluation parameters for automatic question generation.

2.9. Related works

Automatic Amharic factual question generation system has been designed for different languages in the world such English, Chinese, Portuguese and Punjabi. Even though, it is not designed for factual questions, there is a work for Amharic language. In these section, we present some of selected related works based on methods, domain, purpose, input, question format and evaluation techniques.

2.9.1. Automatic question Generation in Amharic language

Although there are no research in automatic Amharic factual question generation, the research in [18] explored automatic Amharic non-factual question generation system using template based approach. The study was conducted to generate Amharic Math Word problem and equation automatically, to help for understanding and learning problems in mathematics. The study used the Walta Information Center (WIC) tagged corpus, which is manually tagged by the Ethiopian Languages Research Center staff member and 154 collected Amharic math word problems from elementary school mathematics textbooks and other school worksheets. The collected corpus from elementary school math textbooks and other school worksheets passed through some preprocessed phases such as segmentation, tokenization, and PoS tagging based on WIC tagged corpus.

After preprocessing sample AMW problem, the task of AMW problem and equation generation involves two steps to generate AMW problem and equation: template formation and generation of AMW problem and equation. In template formation process, segmentation, tokenization, PoS tagging, equation formation and template formulation have been performed. A template includes a placeholder AMW problem, problem type and equation template. In the second phase to generate AMW problem and equation the system used formed template. To generate semantically equivalent new problems, Amharic WordNet used to generate similar patterns by generated templates. In effect, in the engine, the Equation Solver subcomponent solves the equation and puts the solution back to a database along with the necessary measures.

Evaluation of the system were conducted on three experiments. The first evaluation has been performed on generating AMW problem and equation template which has 93.84% overall performance, the second evaluation was compared system generated AMW problems to human generated AMW problems to measure the problem relatedness has also shown 90.24% relatedness. Finally, the generated problems solvability by comparing given correct answer to answer by human solver which has shown 90% accuracy.

2.9.2. Automatic question Generation in English language

The work of [10] was designed to resolve the difficulties of texts that lack practice exercises and evaluations. The study was carried out on the automated generation of factual questions that can take texts as input and produce questions as output to assess the awareness of the readers, how much information from the text should be grasped. The research used text corpora of English Wikipedia and Encyclopedia Britannica articles and also basic NLP techniques to analyze the linguistic properties of source texts. The main NLP tool used for preprocessing was Stanford Parser to automatically sentence-split, tokenize, and parse input texts, Tregex tree to determine syntactic elements that are important, Supersense Tagger to map appropriate answer phrases to WH question words.

The research used rule based declarative sentence transformation approach. After preprocessing step, Tregex mark phrases which cannot be answer phrases on WH-movement from declarative sentence. Then the next step select answers by ignoring those unanswered phrases which did not generate questions by the system. After identifying unmovable phrases, the system generates possible questions based on possible answer phrases. To generate questions, subject-auxiliary inversion was performed by decomposing main verb on the selected answer phrases. In the next step, the system removes answer keys and substitute WH-question words to generate possible questions and also post-processing and question rank has been performed.

The system evaluated individual components as well as the whole system. The manual evaluation was conducted to evaluate the quality of the generated questions by taking 150 input sentences from Wikipedia articles. The evaluation used two people who didn't participate in the research and rate 1 to 5 scale for the output of 150 output sentence paraphrase and sentence compression based on fluency and correctness. The system covers 87.9 % of input words appeared in at least one output from the full extraction system.

2.9.3. Automatic question Generation in Chinese language

The study of [5] was aimed to explore tools to construct question that could be essential in educational technology. The research was conducted to generate factual questions that have realistic responses, primarily in a declarative sentence using syntactic and semantic information, which is then converted into questions. The study used Chinese Reader textbook corpus and performed preprocess such as word segmentation, part of speech tagging (PoS), named entity recognition (NER), dependency parsing and semantic role labeling.

The research used rule based approach to construct questions. The system followed three steps to generate questions: sentence simplification, question generation and ranking. Sentence simplification is the first stage in question construction and essential to reduce complexity of question generation since Chinese sentences could be very long and contains two or more independent sentences connect by conjunctions. Sentence simplification performed by a set of predefined rules. After declarative sentence simplification stage, sentences transformed into a set of questions based on rules for question generation. Answer phrase identification from the input declarative sentence performed as a subtask in question constructions. Finally, questions could be ranked based on their quality with respect to syntax, semantics or importance.

The system evaluated the quality of automatically created questions before and after ranking. Precision and recall was used to measure the performance of question generation systems. The average recall and precision was 64% and 68.9% respectively.

2.9.4. Automatic question generation in Portuguese language

The work of [70] was attempted to solve challenges in education to construct questions manually and used for assessing learners' achievement. The study was carried out to automate factual question construction based on rule based approach. The research performed preprocessing, key selection, question construction and post processing phases.

Questions were constructed where one or more selected keys are removed from sentences and replaced by appropriate question phrases. The researcher used PoS, NER dependency parsing, SRL parsing techniques to generate questions. Finally, generated questions are processed for complete and full question construction.

The evaluation of the system was done by evaluation metrics of question generation based on the syntactic, semantic and dependency based results. The overall generated question acceptability was between 70 to 80 percent.

2.9.5. Automatic question generation in Punjabi language

The research in [1] on factual question generation systems was conducted for Punjabi language. The work was conducted to generate questions from Punjabi text based on rule based approach. The corpus used for the study was collected from Punjabi websites and Punjabi textbooks. Before constructing questions, the researcher performed main preprocessing activities on the dataset such as tokenization with sentence and word level and rule based named entity preparations.

Question construction mainly passed content selection, question type selection and question construction process. The researcher selected target phrases by unmarking phrases which could not answer phrases by comparing with the values in the list of named entities. If the phrase are not found in the list of named entities, it could have been considered unmovable. After marking unmovable phrases, the system generates questions on the remaining phrases. To generate question phrases, the system compare answer phrases with values of named entity list such as Names, Locations, dates etc. The researcher used various named entity rules to resolve unknown names. After generating possible question phrases, remove the answer phrases and insert possible question phrases with a separate copy to generate new questions. To generate question, the researcher used language dependent rules. For example: if the answer phrase could be date, month, week then the answer phrase used the question type when. After generating questions, post-processing has been ensured for proper formatting of the final output such as de-tokenized to remove additional spaces, symbols and also add question marks at the end of sentences.

The evaluation question generation system has been done by manually. The researcher used recall, precision and f-measure standard evaluation metrics to evaluate the performance of the system. Questions generated from the system and questions prepared manually from Punjabi websites and textbooks was compared to evaluate the system. The sentence that did not include any named entity affects the overall performance. Over all recall of the system was 35.72 % because sentences without named entity recall has been 0.00% and also 63% of overall precision.

Table 2.2 below summarizes the survey presented above

Author	Title	Purpose	Input	Method/ tool	Domain	Question format	Language	Evaluation
Andinet A [18]	Automatic Generation Of Amharic Math Word Problem And Equation	support learning	questions	template based, wordnet	Math	Math Word Problem and Equation	Amharic	Human-expert review and Compared with pre-questions, 93.84% accuracy
Michael H [10]	Automatic Factual Question Generation From Text	Assessment, providing questions for academic courses and school textbooks	text	rule based, machine learning (for ranking), stanford parser, supersense tagger	Generic	Wh	English	Human-expert review, average of 61.7% precision.
Ming et al [5]	Automatic Chinese Factual	Education	text	template based, rule based	Generic	Wh	Chinese	automatic evaluation metrics, 64% recall and

	Question Generation							68.9% precision
Bidyut et al [46]	Factual Open Cloze Question Generation For Assessment Of Learner's Knowledge	assessment	text	rule based, Stanford Parser, Stanford CoreNLP	Generic	fill-in-the-blank	English	Human-expert review, 92.37% accuracy.
Vishal et al [1]	System for Generating Questions Automatically From Given Punjabi Text	assessment, support learning	text	rule based, NER, manually collected person Name and location name	Generic	Wh	Punjabi	compared with human-authored questions, 63.0 % precision
Bernardo et al [70]	Factual Question Generation For The Portuguese Language	Education	texts	rule based, Semantic Role Labeling, dependency parser	Generic	Wh	Portuguese	Human-expert review, 69.85% average accuracy.
Yvonne S [71]	Automatic Generation Of Factual Questions	assessment, support learning	video	rule based, Stanford	History	Wh	English	manually compared with pre-questions,

	From Video Documentaries			parser, NER				average 56.0% accuracy
Husam et al [42]	Automatic Question Generation From Sentences	support learning	Sentence	rule based, Oak system for PoS and NER, syntactic parse	Generic	Wh	English	manually compared with pre-questions, 58.7% precision
Iulian et al [20]	Generating Factoid Questions With Recurrent Neural Networks: The 30M Factoid Question-Answer Corpus	assessment	text	template based, machine learning	Generic	Wh	English	human-expert review and automatic evaluation metric, 93.0% relevance
Onur K [72]	Automatic Question Generation Using Natural Language Processing Techniques	assessment, tutoring, support learning	text	rule based approach, NER parser, dependency parser, SRL	generic	Wh	English	human-expert review and automatic evaluation metric

2.9.6. Summary

Automatic AMW Problem and Equation generation systems generate non-factoid questions from pre-prepared math question dataset. It generates questions by changing numbers and entity names from the dataset using template based pattern matching. Amharic factual question can't be automatically generated by the AMW problem and equation because questions are not simply from stored questions but from any text. In Amharic there are no publications or works available for automatic factual question generation to refer as previous work on Amharic AQG using rule-based or automated approaches are not found. On the other hand, the reviewed automatic factual question generation systems are language dependent, i.e., only generate question with their respective language. Since issues related to Amharic language are different from these languages, in addition to the question generation patterns of factual Amharic questions is different from those languages, factual AQG system can't generate Amharic factual questions. Therefore this research tried to work on Amharic factual AQG using rule-based approach from the scratch.

Chapter Three

Amharic language

3.1. Background

Amharic is the most spoken semitic language next to Arabic and also spoken by more than hundred million Ethiopians and a minimum of 500 years written language. It is the Federal Democratic Republic of Ethiopian official working language and thus has official status nationwide. It is also the official language of many of the states/regions within the federal system, including Amhara and the multi-ethnic Southern Nations, Nationalities and Peoples region. It is the language of millions of emigrants in the world outside Ethiopia and is also spoken in Eritrea [73]. It used for instructions in elementary school as a major language in Ethiopia.

The Amharic alphabet used a writing system called Fidel /ፊደል or abugida, adapted from that used for the now-extinct language of Ge'ez. The alphabet is written from left to right and consists 7 vowels and 33 consonant and 231 syllable patterns, or fidels [74]. There are other non-standard alphabets, in addition to the 231 characters or in total 275 Fidels. Amharic alphabets are shown at Appendix 1 [75].

3.2. Amharic word classes

Amharic language has different categories of word classes. Some researchers have categorized into eight parts of speech but others grouped into five. The eight word classes are: ስም (noun), ግስ (verb), ተውሳከ ግስ (Adverb), መስተዋድድ (preposition), ተውላጠ ስም (pronoun), ቅፅል (adjective), መሥተፃምር (conjunction) and interjection (ቃለ-አጋኖ) [76].

Nouns/ስም in Amharic represents gender (masculine or feminine), object, case, number and definiteness. The inflection is accomplished by either changing vowels or repeating consonants; the requisite affixes are then added. For example, vowel ending nouns take the form -ዎች for instance ተማሪ + ዎች = ተማሪዎች while consonant ending nouns add -ኦች for instance ወንበር + ኦች = ወንበሮች at the end of their root word. In the case uncountable nouns, repeat themselves to make it plural noun. For instance ጌጥ can be pluralized by ጌጥ-ኣ-ጌጥ = ጌጣጌጥ [76].

Verbs/ግስ are words that inflect person, number, gender, mood, voice, and polarity, formed from roots and affixes [77]. It can be placed at the end of a sentence. Some examples of Amharic verbs are በላ, ጠጣ, ተኛ, ተናገረ [78].

Adjective/ቅፅል are any word that qualifies a noun or an adverb, which actually comes before a noun (ጎበዝ ተማሪ) and after an adverb (በጣም ጎበዝ) [15].

Adverbs/ተውሳከ ግስ are words that will be used to qualify a verb by adding extra idea on the sentence. Some examples of Amharic adverbs are ትናንት, ገና, ዛሬ,ቶሎ,እንደገና [15].

Preposition/መስተዋድድ is a word which can be placed before a noun and perform adverbial operations related to place, time, cause and so on; which can't accept any suffix or prefix; and which is never used to create a new word. Some examples of Amharic prepositions are ከ፣ ለ፣ ወደ፣ ስለ፣ እንደ [78].

Pronouns/ተውላጠ ስም are words that used in place of noun or noun phrase. Pronouns refer either to a noun already described or to a noun that does not clearly need to be named. It can be further divided as deictic specifier (e.g. እሱ,ያ, አንተ, አንቺ), quantitative specifier (አንድ, አንዳንድ, ጥቂት, በጣም) and possession specifier (የኔ, የአንተ, የሱ) [78].

Conjunction/መሥተፃምር are words that can help to connect words, phrases and also clauses. Some examples of Amharic conjunctions are እና, ወይም, ነገር ግን [15].

Interjection/ቃለ-አጋኖ are words or phrases used to express such emotions as sudden surprise, pleasure, annoyance and so on. Such Amharic words are called interjections. For Some examples of Amharic interjections are የምሥራች!, ስማ! ጎሽ! [78].

The main objective of this research is to generate questions from set of sentences by using PoS tag and NER tag. However, to prepare PoS tag and NER, parts of word are very essential.

3.3. Amharic punctuations and numerals

Amharic documents contain different punctuations for different purpose. Before generating questions, documents should be preprocessed. For instance, an Amharic sentence mostly ends with double colon (።), question mark (?) or exclamation mark (!). Colons (:) has been used to separate words in a sentence in older writing system. These days it is replaced with whitespace. Like comma in English, ነጠላ ሰረዝ (፣ or ፤) is also used to separate list of ideas. Thus, sentences that end with Amharic full stop (።) should be selected to generate questions. Amharic punctuation marks are shown at Appendix 2 [75].

Numerals in the generation of Amharic questions have a very significant effect. Numbers are written in different forms in Amharic documents, such as Ethiopic and Arabic number representation. In question generation, numbers in a document should be normalized to the same standard in order to make a document suitable for answer key selection component. Amharic has Ethiopic number representation. Numerals can be also be represented using Arabic symbols and in alphanumerical words [79]. Amharic numerals are shown at Appendix 3 [75].

3.4. Amharic Sentence

According to Andersen, a sentence is a collection of words that can give a complete meaning. Therefore, one can decide, a sentence is a group of words that consists subject and verb [80]. Ayinadis [81] states that a sentence is a word or group of words expressing a judgment of the mind. A sentence should be complete to express something. According to Frank [82], sentence is generally classified into two categories. These are based on its function and structure.

3.4.1. Amharic sentence function

Sentences based on their function or purpose can be classified into four categories. These are declarative sentence, interrogative sentence, imperative sentence and exclamatory sentence.

Declarative sentence: a sentence that explains a fact and ends with a full stop (.). In Amharic this type of sentence ends with four point (::). አልማዝ ቤተ ክርስቲያን አጠገብ ትኖራለች፡፡ /'Almaz lives near the church.'

Interrogative sentence: it is a sentence that helps to get some information about something and ends with a question mark (?). Amharic interrogative sentences also ends with question mark (?). For example, አልማዝ የት ትኖራለች? /'where do Almaz live?'

Imperative sentence: it is a command or a request that is a friendly type sentence and also ends with a period (.) unless the writers want to convey strength and powerful emotion, then the sentence ends with exclamation mark (!). For example, ሁሉንም ኩኪስ አትብላ. / ‘Don't eat all the cookies.’ ከዚህ ሂድ! / ‘Get out of here!’

Exclamatory sentence: This sentence helps to exhibits excitement or demonstrates intense feeling and also ends with an exclamation mark (!). For example, የኮንሰረቱን ትኬት አገኘሁ!, “I got the concert tickets!”

3.4.2. Amharic Sentence Structure

In addition to functions, sentences based on its structure can be classified into four groups [82]. Namely, Simple sentence, Complex sentence, Compound sentence and Compound complex sentence.

Simple sentence is a sentence that consists of only subject and predicate. It contains a single independent clause, a clause includes of a sentence's main subject and verb. For example, ምድር ሰፊ ናት: “The Earth is spacious “.

Complex sentence is a sentence which contains more than one noun phrase or verb phrases. For example, እንደጠበቅኩት ሙዝየሙ በጣም አስደሳች ነበር ፡፡/’ The museum was very interesting, as I expected.’

Compound sentence is a sentence which contains more than one independent clause with clauses joined by coordinate conjunction words like and, but, or, nor, yet, etc. For example እሷ ብልህ ግን እብሪተኛ ናት:: /’she is smart but arrogant.’

Compound complex sentence is a sentence which has two or more independent clause and one or more dependent clause. For example, ኤሪን ወንድሟን ትወዳለች ፣ እሱ ደግሞ የበለጠ ይወዳታል ምክንያቱም ሂሳቦቹን ትከፍልለታለች ። / "Erin loves her brother, he loves her even more because she pays his bills."

According to Ayinadis, sentences in Amharic can be classified into to two: simple sentence and complex sentence. A simple sentence is constructed from noun and verb phrases. It has only a single verb phrase. Whereas a complex sentence is a sentence formed by complex noun phrase or complex verb phrase or combination of two phrases. In Amharic, the word order inside a sentence is different from that in English (SVO). A general sentence structure in Amharic is subject /object / verb (SOV) or a simple Subject +verb arrangement [81]. But there may be conditions that sentences formed as OSV. For instance, the sentence ‘አበበ ከበደን ገደለው’ / ‘abebe kebeden gedelew’ which is in SOV form, can be rearrange as ‘ከበደን አበበ ገደለው’ /’kebeden’abebe gedelew’, in OSV arrangement. Taking these sentences into account, one can only be sure regarding the verb occur at the end of simple sentences and the subject and verb may share place. Although, the OSV word arrangement cannot be used in a common Amharic text. The word order in a sentence is not limited by only subject object verb. Sentences may contain other word classes like Adjective (ቅፅል), Adverbs (ተውሳከ-ግስ) etc in their own grammatical order.

3.5. Amharic interrogative sentence

Interrogative sentence is a form of sentence that asks a question as compared to sentences that make an argument, deliver an order, or express an exclamation. This type of sentence has a significant role in our everyday life, since most of the discussions occur while asking and answering questions. An interrogative sentence ends with question mark (?). Interrogative questions can be wh-question (content question) or yes/no question [76].

Questions might be posed regarding any action or situation, the performer of an action, the person performing the action or time and place, the cause of the action or intent of the action, how the action is performed or the techniques used to execute the action etc [83].

There are a number of interrogatory words to construct questions in different languages. English has question words like who, where, what, when, why… are used to construct a question. In Amharic, there are a variety of interrogative words that will help to create a question [76]. Amharic pronoun question words are ማን, ምን, የት, ስንት and መቼ, but by joining these question words with prepositions/መስተዋድድ, we can also create prepositional question words like ለምን, በምን, ከምን, ከየት, ከማን, ከስንት, በስንት, በመቼ, በየት etc. to construct interrogative sentences.

The formation of interrogative sentence structure varies from one language to another language. It is most likely that only one language family has the same structure when asking questions. Although, questions are built in each language using interrogative words and a question mark (?), which is placed at the end of the question. For example, in English, interrogative sentences used interrogative words and question mark at the end of the sentence, but the internal structure of the sentence is formed by the inversion of the subject and predicate; that is, the first verb in a verb phrase comes before the subject. But Interrogative sentence in Amharic are varied. Using different grammatical sentence structures, Amharic interrogative sentences can be constructed by using wh-questions. Yes /no interrogative sentences doesn't used interrogative words but add question mark at the end of the sentence simply and the sound of interrogator. From Amharic sentence, we can ask the subject of the sentence, complement/መሙያ of the sentence, specifiers/አጎላማሽ, Adverbs/ተውሳከ ግስ, verbs/ግስ etc to construct interrogative sentences. For example, to ask questions that we don't fully know about, we use adverbs, to be certain of what we know, we use other words or tones [76].

For example: ካሳ ትላንት ሁለት በጎች ገዛ፡፡

From this sentence we can construct several questions using different pronoun question words. Like:

- ማን ትላንት ሁለት በጎች ገዛ?
- ካሳ መቸ ሁለት በጎች ገዛ?
- ካሳ ትላንት ምን ገዛ?
- ካሳ ትላንት ስንት በጎች ገዛ?

The first interrogative sentence is constructed by asking the subject/person (ካሳ) by replacing interrogative pronoun words ማን. Such kind of questions need answers that specifies person like ካሳ or አበበ for this specific question ካሳ. In the second example, the interrogator needs to know the time when ካሳ bought two sheep and used an adverb (መቸ). The answer for this kind of questions can be yesterday or today or last week. From the above declarative sentence (ካሳ ትላንት ሁለት በጎች ገዛ፡፡) all words are replaced by interrogative words respectively in its structure to construct interrogative sentences except action word (verb). To construct interrogative sentence using action word (verb), we can join ምን (what) pronoun interrogative word with አደረገ. For example: ካሳ ትላንት ሁለት በጎች ምን አደረገ? From all the above examples, there is no interrogative sentence constructed by specifiers and complement. The example bellow shows how to construct interrogative sentence using specifiers and complements.

For example: ካሳ በመኪና ወደ ጎጃም ሔደ፡፡

This sentence contains two prepositional/መስተዋድድ phrases.

- በመኪና is specifier/አጎላማሽ prepositional phrase.
- ወደ ጎጃም is complement /መሙያ prepositional phrase.

By using these prepositional phrases, the following interrogative sentences can be constructed.

- ካሳ በምን ወደ ጎጃም ሔደ?
- ካሳ በመኪና ወደ የት ሔደ?

From all the above examples, we have seen a single interrogative words in each interrogative sentences. But, in a single interrogative sentence, we can get more than one interrogative word. In other words, using one interrogative sentence we can ask many questions. For example: ካሳ በጦር አንበሳ ገደለ፡፡ From this declarative sentence, we can formulate interrogative sentences like, ማን በምን አንበሳ ገደለ?. This sentence used Subject/ባለቤት and specifiers/መሙያ prepositional interrogative words to ask questions in one sentence (ማን በጦር አንበሳ ገደለ?, ካሳ በምን አንበሳ ገደለ?).

Generating question in Amharic is challenging. One of the challenges is that, Amharic does not use rule or concept of capitalization to identify entity classes of a word. For example, the word ታደሰ can be person name or verb (something is renewed). Therefore, it is very difficult to decide the word is a proper name or not.

Chapter Four

Research Methodology

4.1. Overview

This chapter discusses the research methodology in detail that is used to provide the mechanism to solve the research problem. Research methodology is the set of procedures, methods, tools and techniques implemented and used to conduct a research and reach to the final output of the study [84]. Selecting appropriate research methodology is an essential step in conducting a research. The research paradigm can produce the desired solution to the identified problem. A methodology that incorporates procedures and approaches that are the best fit for the research is chosen in order to achieve the research's aims and objectives, as well as offer valid and reliable results. The research includes designing a new artifact to tackle observed problems, evaluating the artifact, and presenting the findings. As a result, the design science research approach is a best fit for this study. Hence, an artifact, a prototype that used to construct Amharic questions automatically from text, is rigorously designed and evaluated. In the next section, design science research are presented in details.

4.2. Research Methodology

According to Hevner et al [85], there are two main research paradigms in Information Systems. These are behavioral science and design science. Behavioral science is concerned with the "*development and justification of theories that explain or predict phenomena related to the identified business need*" whereas design science is concerned in the "*building and evaluation of artifacts designed to meet the identified business need*". Since this research intended to design and develop a prototype based on the language syntax, the researcher utilized the design science research methodology (DSRM) as a research approach. Design science research, according to Baskerville et al [86], is a research methodology that will tackle practical and theoretical limits in IS.

Design artifacts and design theories are two perspectives provided by DSR methodology. The approach includes: constructs, frameworks, models, architectures, design principles, methods, instantiations, and design theories as a research output. Design science will play a key role in bridging the gap by delivering the aforementioned outputs through design, analysis, reflection, and abstraction [86].

According to Marcus et al, design science process includes six steps: problem identification, definition of the objectives, design and development, demonstration, evaluation, and communication [2]. Since this research aimed to design and develop a prototype for automatic factual question generation for Amharic language, the researcher focused on the Design and Development centered approach. Therefore, the DSRM process model shown in Fig 4.1 is adopted and modified for this research.

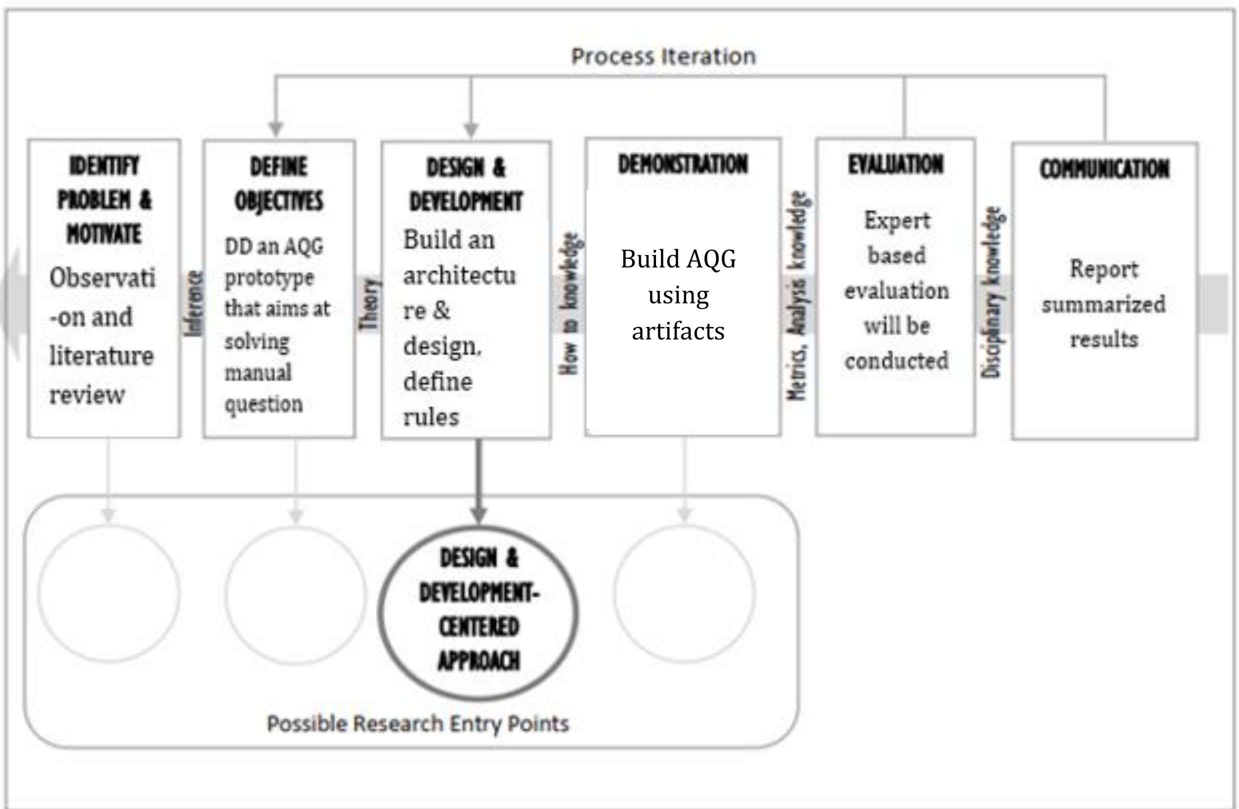

Figure 4.1 DSRM Process Mode adopted from *[2]*

4.2.1. Problem Identification

Problem identification is the first stage in design science research approach. Identifying key areas and needs are considered, while conducting research on some topic. This research started by identifying and understanding the problem to automate question generation for Amharic language. We did an observation and a literature review on the topic. Journal articles, books, lecture notes, resources on the web are referenced and reviewed. By observing the problems, asking people some questions about the study, and conducting a literature review, we attempted to design a solution based on the problems identified.

4.2.2. Defining of objectives for a solution

The research objectives should be well defined after identifying the problems in the previous step that need to be solved. As a result, after identifying the problems and define the objectives of the study, we proceed with the design and development of the study while keeping these objectives into consideration.

4.2.3. Design and Development

The design and development process involves developing a design and architecture to automate Amharic factual questions generation. The architecture includes tagging sentences in a text with their respective named entities, identifying informative sentences, selecting answer keys, transform sentences to questions and post-processed for appropriate format of the question. Based on the architecture, the prototype of AAFQG was designed and developed.

According to Peffers et al [2], designing process starts with data collection and analyzing the data. For this research three types of data are collected. The first data is collected about Ethiopia historic data from different available sources. The second data is collected about Amharic news corpus from Walta Information Center (WIC) which is manually tagged by the staff member of the Ethiopian Languages Research Center (ELRC) [87] and Amharic web corpus tagged by TreeTagger [88]. The third data is collected about Amharic person, location and organization names that helps to develop NER using PoS tagging by replacing nouns with its named entities.

After data collection, it should be pre-processed, in order to make the data that we have on hand in the appropriate format and to make it easier for the machine to read. The pre-processed steps will perform for this research include character normalization, tokenization, and text cleaning.

In order to identify named entities (person, location, organization, date and numbers) in a sentence, this research used named entities recognition tools. The researcher developed PoS tagger using WIC news training data which is manually tagged by the staff member of the ELRC [87] and also combine with Amharic web corpus tagged by TreeTagger [88] (simply taking some top sentences). Since PoS tagger is important for the development of NER as it helps the named entity recognizer to identify words that are nouns and assign the appropriate tag to each nouns that are named-entities. Many studies [89] [90], recommend HMM based tagger for statistical-based approach. The current study used HMM tagger since we are using manually annotated statistical training data.

In this phase, AAFQG design prototype was proposed. The prototype developed using transformation rules that we have prepared based on tagged entity names.

4.2.4. Demonstration

According to Peffers et al [2], Design science prove the application of the artifact on the problem. In the demonstration step, by taking one or more instances which involves prototype, simulation, case study, proof or other suitable activities by using resources such as actual guideline to use the artifact.

For this research, to demonstrate the artifact, prototype is used because it helps to understand how usable and valuable the artifact, to the intended users as well as how to improve it. According to Peffers et al [2], prototype instantiation to demonstrate the efficacy of a design can provide significant proof when used to show a design works as intended.

In order to implement a design and build a prototype, the researcher used different techniques and tools such as Python programing language. Since PoS and NER are used for question generation, we used python to developed PoS and NER because python has an excellent functionality for processing linguistic data [91] and the researcher are familiar to this tool. The approach we have used for building the PoS tagger and NER model is a statistical approach.

The approach selected to this particular research is a rule based approach. Rules can be used to solve tasks like identifying relevant sentences, answer keywords, and question formation in QG. From the possible machine learning algorithms, we reviewed and used HMM for our research for PoS and NER tagger. [63] HMM has 3 parameters: Start Probability, Transition Probability and Emission Probability. Start Probability refers to the likelihood that the form labeled as a tag appears first in a sentence. Transition Probability refers the ratio of the probability of transitioning from one tag to the next and the probability of a given tag occurring. Emission probability is the ratio of the probability of a given word occurring with tag t to the probability of a tag t occurring.

4.2.4.1. POS Development

The prototype development used ANER to recognize entity names in a sentence. Even though there are works for ANER, they are not accessible to the general public. The researcher developed Amharic POS tagger, to be used by the ANER. POS is important to develop Named Entity Recognition since it helps the NER in identifying words that are nouns and assigning the appropriate tag to each noun that are named entities [15]. This research used Hidden Markov Model for part of speech tagging, using Amharic POS tagged WIC news manually tagged and Amharic web corpus tagged by TreeTagger training data.

This research used 15,934 sentences with tagged tokens for training data which is tokenized at sentence level. Training data contains a format word/TAG with words separated by spaces and each sentence on a new line. While, testing data will be untagged development data which tokenized in word levels with separated by white space and also each sentence on a new line.

In order to build the model, the machine learns from the tagged training data using HMM. After the model constructed, the raw untagged data which is tokenized in word level with separated in white space and each sentence in a new line will give as an input and it generated tagged data based on the parameters of HMM from the model and write the result as output file with the same format as the training data.

4.2.4.2. NER Development

For Amharic NLP, especially for the Named Entity Recognizer (NER) there are no such available corpus. So we manually collected and tagged about (3,816 words) for person name and about (500 words) for organization name and about (4012 words) for local place names from the Webs. And also, we used 368 NER tagged words from [15].

In order to build this tools for this research, the researcher used a PoS tagged corpus and the above manually collected (person, location and organization) names. According to Agizew [19], PoS tagged corpus can be manually edited by removing and adding some tag sets that helps to develop statistical NER. The PoS tagged corpus in which the word and tag are separated by a space (word TAG format) and tokenized at word level. Then the data used for training will be prepared by replacing the nouns, which are named-entities by the appropriate tag using rules.

To build the model the machine learns from named entity tagged training data using HMM. Based on the generated model, the untagged raw data which is tokenized in word levels with separated by white space and each sentence on a new line will give as an input. Then it generate named entity tagged data based on the parameters of HMM from the model and write the result as output file with the same format as the training data.

4.2.4.3. Rule Definition

In rule based approach to generate questions automatically from the input text, rules should be developed based on language features [50]. Rules used to generate questions. Approaches that use text as input are often accompanied with rules. Typically, rule-based systems used annotated sentences by named entity information. Then match the input to a pattern provided in the rules using these annotations. These rules used to identify named entities and generate appropriate question type to construct questions from sentences [34].

Since this research used rule based approach, questions are generated by developing handcrafted rules based on entity names such as PERSON-NAME, DATE-FORMAT, LOCATION-NAME, NUMBER etc. According to Kaur et al [51], different rules can be developed to generate questions such as Location rules, Name rules, City rules, Year rules, Numeral rules, Measurement rules and Direction rules. However [52], rules might be different depending on the language features.

Table 4.1 below shows rules, which used to generate Amharic factual questions

	Sentence	**Appropriate wh-words**
Rule 1	If the sentence contains answer key word and tagged as person (per). For example, አፄ ዮሃንስ	ማን (who)
Rule 2	If the sentence contains answer key word and tagged as location (loc). For example, location, city, country name…	የት (where)
Rule 3	If the sentence contains any date format, year, time and list of weeks or months or ትላንት (yesterday) or ነገ (tomorrow) …	መቼ (when)
Rule 4	If the sentence contains any cardinal or ordinal numbers.	ስንት(How much /many)
Rule 5	Replace Amharic full stop (::)	Question mark (?)

Table 4.1 rules used to generate questions

4.2.4.4. Content Selection

In factual QG task all sentences in a textual document may not be useful. In order to formulate factual questions the sentence that has adequate and quality information will serve as an insightful sentence. Therefore, sentence selection has played a pioneering role in the automated task of producing factual questions. Using a few POS tagging or NER based rules, some of those sentences are selected as insightful ones [46] [45].

The analysis of Part-of-Speech (POS) tags and named entity recognition (NER) is the backbone in selecting informative sentences [45]. The analysis performed by collecting simple sentences from text corpus, then tagged it. For this study, we used NER tagged which helps to select relevant sentence such as PER, LOC, ORG, Date, NUM and the sentence ends with Amharic full stop (::). If there are determiners (e.g., those) or pronouns in the sentence, the system will remove them. It also filters away inquiries that are longer than 30 tokens since, while they are uncommon, they are virtually always awkward and unpleasant [10].

4.2.4.5. Prototype Development

A prototype that automate Amharic factual question generation from text is developed based on the syntax respect to their named entities using transformation rules. It is implemented using Python programming language.

4.2.5. Evaluation

According to Peffers et al [92], the research output in DSR approach will be evaluated through different evaluation measures. A logical argument, human-expert evaluation, technical experiment, prototype, subject-based experiment, action research, case study, and illustrated scenario are DSR evaluation approaches. After prototype development, the results are tested, analyzed, and evaluated. This research, evaluated using models automatically for annotation of POS and NER and human-expert evaluator for content selection. Since, this helps to improve the development of prototype. Then the overall developed prototype evaluated using expert evaluator, mainly for the accuracy or correctness of generated questions.

4.2.5.1. POS and NER tagger

An HMM PoS tagger which was discussed above, is constructed for use by the NER. Since performance of PoS tagger affects the NER, the tagger should be properly evaluated. The PoS tagger trained with tokenized at sentence level and manually tagged training data and tested with unseen test data. The PoS tagger trained with 15,934 sentences with tagged tokens and tested with 3,984 sentences with untagged tokens of the prepared test corpus. The performance evaluated automatically using accuracy.

After evaluating PoS tagger, NER was evaluated which is mainly used for this study for content selection, answer key word identification and rule construction. The HMM based NER tagger trained using around 16,130 sentences with tagged tokens and tested by 4,033 sentences with untagged tokens. Accuracy is used to measure the performance of the tagger.

4.2.5.2. Content selection

Question generation targets are selected using simple selection rule based on NER. According to Sedigheh et al [93] , sentence selection methods evaluated using precision, recall, accuracy, and F1scores. This research used accuracy to evaluate sentence selection method. For the purpose of this experiment, we used 300 simple sentences from the collected Ethiopian historic corpus as input to the system and generate 212 relevant sentences. To evaluate relevant sentence selection method, we present the generated relevant sentences for Human evaluator. Then an evaluator classified those sentences as '1' (relevant) and 0 (not relevant) at the end of each sentence. With consideration human evaluator select sentences as relevant, if any of the named entities such as location, organization, date, count and person are present in the sentence, sentence length less than 30 words, sentences ends with Amharic full stop (::) and sentences ignored which contains pronouns.

The systems accuracy measured by comparing correctly selected relevant sentences to total number of sentences selected by the system using human evaluator.

4.2.5.3. Question evaluation

Evaluation is the crucial stage for the growth of natural language generation (NLG) systems. It helps to enhance efficiency and define new tasks to which generation system can be applied. The development of question generation system and evaluation methodology should be parallel, a systematic analysis of evaluation methodologies could play a central role in the effort to construct machines capable of achieving linguistic communication standards that are human-like [65]. Recall, precision, accuracy and F-Measure can be used as evaluation parameters for automatic question generation [50].

Manual evaluation are human judges to evaluate generation system performance. It is easy and reasonably easily accessible [55]. Evaluation can be performed by experts, students or comparison between systems generated questions and human generated question [34]. The generated questions evaluate based on grammatical correctness and appropriateness of the questions created from the sentence as well as their interrogative pronouns [69].

For this study, the system evaluated using accuracy or correctness of generated questions. Questions, generated by the system should be evaluated. In order to evaluate the questions, the researcher should generate questions using the system against the selected 100 relevant sentences from collected Ethiopian historic corpus. Then the generated questions are presented to human evaluators to classify the questions either correct or incorrect with grammatical and semantic correctness.

Since this research used accuracy, as evaluation metrics. We define Accuracy [64] as follows.

$$Accuracy = \left(\frac{\boldsymbol{QGcs}}{\boldsymbol{QGts}}\right) \boldsymbol{*100}$$

$\boldsymbol{QGcs}$ $= Questions\ generated\ by\ the\ system\ correctly$

$\boldsymbol{QGts}$ $= Questions\ generated\ by\ the\ system\ totally$

Where, QGcs is the number of correctly generated questions by the system as human judgment and QGts is the number of totally generated question by the system.

4.2.6. Communication

Communication is the final stage of design science research process. In this step the final results from the experiment should be done. In addition, the complete research process was thoroughly documented and communicated as a thesis work.

Chapter Five

Design of automatic Amharic question generation system

5.1 System design and development

5.1.1 System architecture

The architecture is adopted from [1], with some components included for this research, such as content selection, PoS tagger and NER which is important to develop our prototype. Figure 5.1 shows the adopted architecture of the system.

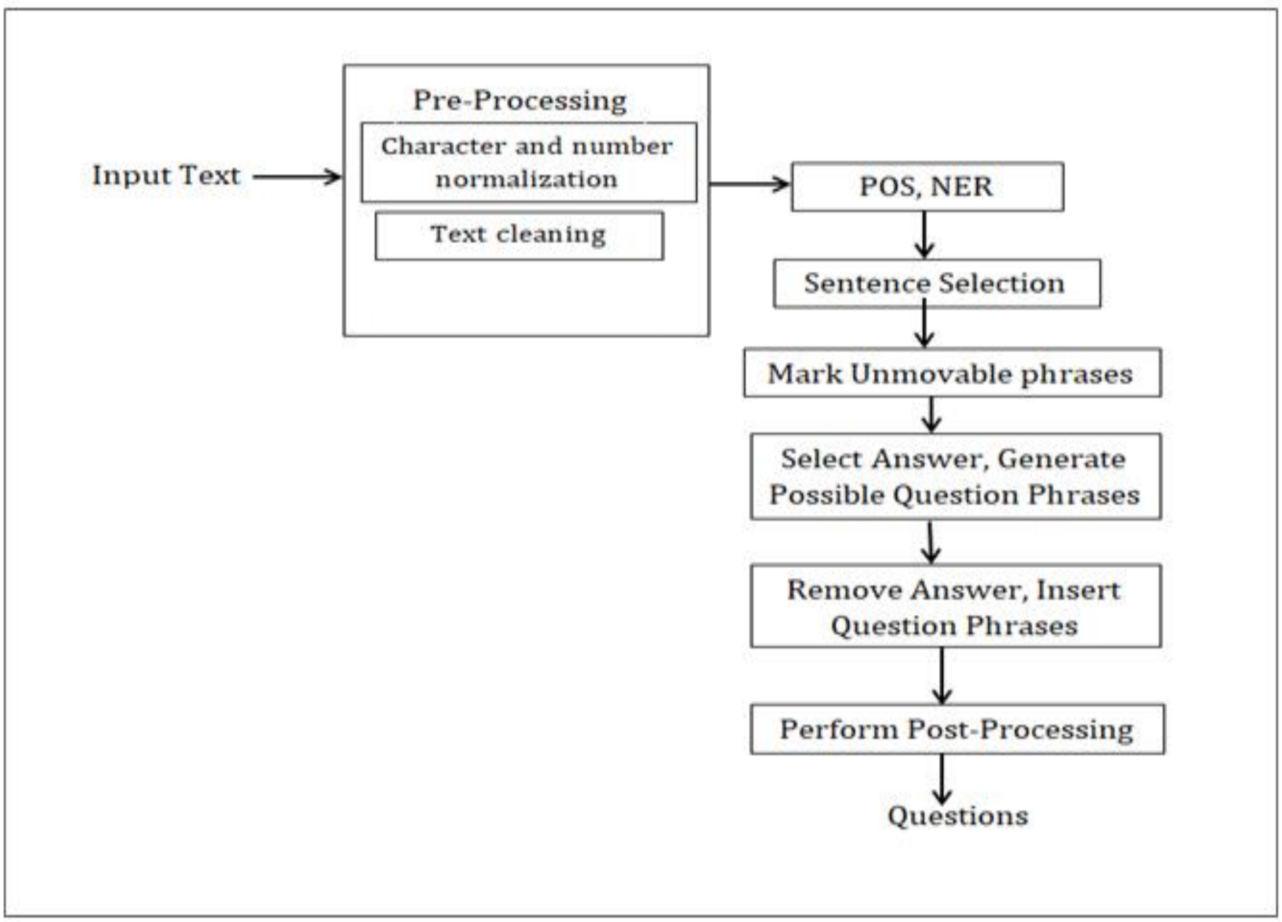

Figure 5.1 Architecture of Automatic question generation *[1]*

5.1.2 Dataset preparation

Dataset preparation is the first phase in the system's architecture. A corpus needs to be prepared before proceeding to the design and development step. Since the research focuses on a specific domain or closed domain automated question generation, the corpus used in this study is only concerned with Ethiopian history. A corpus of 300 simple sentences (section 3.4.2) are collected from various Amharic history books, newspapers, online history books, and the internet.

The other 19,918 sentences with tagged tokens which is important to develop Amharic PoS tagger is collected from Walta Information Center (WIC) which is manually tagged Amharic news corpus by the staff member of the Ethiopian Languages Research Center (ELRC) [87] and Amharic web corpus tagged by TreeTagger [88].

5.1.3 Preprocessing

Following data collection, pre-processing is the next step to ensure that the corpus at hand is in the appropriate format and readable by the machine. Cleaning and processing of raw data is the important preliminary step of any NLP task. Character normalization, tokenization and text cleaning are the pre-processing steps performed for this research. In addition, other tools such as PoS tagger and NER are used before getting to the question generation components.

Character and number Normalization: In Amharic one word can be written in different characters or fidels but the sound and the meaning are the same. For example ሀ, ሃ, ሐ, ሓ, ኀ, ኃ and ኻ sound the same and አ, ዐ, ኣ and ዓ are Amharic alphabets that are often used interchangeably in a single word. For instance ሀይማኖት, ሃይማኖት, ሐይማኖት, ሓይማኖት and ኀይማኖት and አብይ, ዐብይ ኣብይ and ዓብይ have the same sound and the same meaning Religion and person name respectively but the characters are used according to an individuals' interest. Therefore, one needs to normalize these characters to avoid any differences on similar words that are written using different characters with the same pronunciation but different structural appearance in order to make the machine understand each meaning.

The purpose of character normalization in this study is to eliminate ambiguity while applying machine learning (HMM) to tag our historical data with PoS and NER. For example, the machine may tag አብይ and ዐብይ differently if the model trains with አብይ. The machine may accurately tag

አብይ but not for ዐብይ. Character normalization can also help for PoS and NER tagging to eliminate redundant tags of single words.

To normalize characters, python is used to simply find and replace any characters at once that have the same meaning and pronunciation with one representative characters. Amharic interchangeable characters and their normalize character for each case are shown at appendix 4.

Tokenization: is another pre-processing techniques which is a key aspect of working with text data. Sentence and word level tokenization is performed in this study. Sentence tokenization is used since we need a single sentence for a single line for the analysis purpose, which is done by using different sentence demarcations such as question mark (?), exclamation mark (!) and full stop (።). In this study, we used the Amharic full stop (።) to signify the end of a sentence. Additionally, word tokenization used white spaces to tokenized words. This process will help to select more useful sentences and to get answer keys easily.

Word tokenization is also used to do some tasks such as PoS and NER because a word-tokenized data is provided to both PoS and NER as an input.

Segmentation: Most higher-level natural language processing tasks, such as part-of-speech tagging (PoS), parsing, and machine translation, begin with word segmentation. It can be regarded as the problem of correctly identifying word forms from a character string [94].

We identified some Amharic prefixes and suffixes latters and segmented them, which are mostly found in large words, particularly on named entity words. Those are prefixes like **ለ, ከ, የ, በ, እነ** and suffixes like **ንና, ና, ን, ም.**

After segmentation has been done, we used Named entity tagger to tag our corpus and prepare our corpus for question generation. For example, the sentence <<የአበበ ቢቂላ እናት ወይዘሮ ውድነሽ መንበሩ ይባላሉ፡፡>> segmented as <<**የ** አበበ ቢቂላ እናት ወይዘሮ ውድነሽ መንበሩ ይባላሉ፡፡>>. This sentence can be tagged "**የ**" as preposition and **"አበበ"** as person which helps to generate question like **"የማን"** after De-segmentation "**የ"** in **"የ ማን"** in post-processing stage.

In this study, word segmentation helps to generate more questions by joining Amharic pronoun question words **(ማን, የት, ስንት, መቼ etc**.) with prepositions/መስተዋድድ, to construct prepositional question words like **ከየት, ከማን, ከስንት, በማን, በስንት, በየት etc**. to construct interrogative sentences. Sample preprocessing python code are shown at Appendix 6.

PoS tagging: is an NLP technique which performs the process of marking up a word in a text as appropriate to a particular part of speech of words. It helps to identify answer key words from a sentence using the part of speech of each word and select based on their relevance. Furthermore, PoS tagger is important to develop Named Entity Recognition since it helps the named entity recognizer in identifying words that are nouns and assigning the appropriate tag to each noun that are named entities. The PoS tagging developed to identify part of speech of words in a text also helps to develop Amharic NER. Hidden Markov Model is used for part of speech tagging using tag sets noun, pronoun, verb, adverb, adjective, preposition, number, punctuation and conjunction [15]. 15,934 sentences with PoS tagged tokens are used for training which is tokenized at sentence level. Training data contains a format word/TAG with words separated by spaces and each sentence on a new line. While, testing data will be untagged development data which is tokenized in word levels and separated by white space and each sentence on a new line.

To build a model the machine learns from tagged training data using Hidden Markov Model. After the model is generated, the raw untagged development data which is tokenized in word levels separated by white space and each sentence on a new line will be given as an input. The model then generates tagged data based on the parameters of HMM and write the result as output file with the same format as the training data.

Question generation is also highly dependent on PoS tagging. NER needs PoS tagged corpus in order to find the nouns used for question formation to select answer key words and question phrases which needs a PoS tagged data.

Named Entity Recognition (NER): is used to recognize the names from a given input text. This helps to generate question based to the recognized entities. Many of the factual-based questions are generated by substituting entity names that can be detected and extracted using NER with question words. The NER used in question generation is developed to extract the entities and helps in the reduction of unwanted strings to generate the question automatically [51]. The development of NER mostly used two approaches, Statistical and Rule based approaches. For this study, we used statistical based NER tool for relevant sentence selection and answer keyword identification.

For Amharic NLP, especially for the Named Entity Recognizer (NER) there are no such available corpus. So 3,816 words for person names and 500 words for organization names, 4012 words for local place names from the Web are collected and manually tagged for this research. And also, we used 368 NER tagged words from [15].

The data used in order to build this tools for this study, are a PoS tagged corpus and manually collected (person, location and organization) names. PoS tagged corpus manually edited by removing and adding some tag sets with manually collected person, location and organization names that helps to develop statistical NER. For instance, tags cardinal number (CN) and ordinal number (ON) as NUM and nouns (NN) which are named entities (person, location and organization) are replaced by PER, LOC and ORG respectively. The PoS tagged corpus in which the word and tag are separated by a space (word TAG format) and tokenized at word level.

After the training data is prepared by the machine learns to build the model from tagged training data (NERLearn.py) using HMM. Based on the generated model, the untagged raw data which is tokenized in word levels with separated by white space and each sentence on a new line is given as an input (NERDecode.py) and the generates tagged data based on the parameters of HMM from the model and write the result as output file (output.txt) with the same format as the training data.

In the training corpus, begging and inner entities were used to tag named entities that were composed of more than one word. For example, the entity: <<ዳግማዊ ዓፄ ምኒሊክ >> is tagged as <<ዲግማዊ/B-PER አፄ /I-PER ምኒሊክ/I-PER>> to signify Emperior Minilik and <<ዘውዲቱ ሆስፒታል>> is tagged as <<ዘውዲቱ/B-ORG ሆስፒታል/I-ORG>> to represent a hospital.

For this research, we used named entities; person (PER), location (LOC), organization (ORG), date (DATE) and Number that mainly helps to construct questions. To build a model the machine learns from tagged training data using HMM. After the model is generated, the raw untagged development data which is tokenized in word levels separated by white will be given as an input. The model then generates tagged data as output file with the same format as the training data.

5.1.4 Informative sentence selection

In factual QG, all sentences in a textual document may not be useful. In order to formulate factual questions the sentence that has adequate and quality information will serve as an insightful sentence. Therefore, sentence selection has played a pioneering role in the automated task of producing factual questions. Using a few PoS tagging based rules, some of those sentences are selected as insightful ones [46].

Informative sentence selection is usually done using different methods from the field of NLP such as sentence simplification, parsing, semantic role labeling, PoS tagging or named entity recognizing [45]. The analysis of PoS tags and named NER is the backbone in identifying informative sentences. The analysis is performed by collecting simple sentences from text corpus. For this study, NER tagged sentences are selected which have PER, LOC, ORG, Date, NUM and ends with Amharic full stop (::). Sentences exceeding 30 number of words are ignored.

For example:

1. በካፋ/B-PER ህጻንነቱን/NN ያሳለፈው/V ወህኒ/B-LOC አምባ/I-LOC ላይ/PREP ነበር/V::/PUNC
2. ቀጥሎም/ADV ለዚህ/PRO ስርኣት/NN ክብር/NN ሲባሌ/NN በአዲራሽ/NN ውስጥ/PREP ግብዣ/NN አዴርጎ/VP ሲያበቃ/VP ሚስቲቱ/NN ከቀረበው/VP ምግብ/NN እንደበላች/VP ታመመችና/VP ማታውኑ/ADV አረፈች/VB::/PUNC
3. የልጅ/NN ምኒልክ/I-PER ሠራዊት/NN እና/CONJ የዓፄ/B-PER ቴዎድሮስ/I-PER ሠራዊት/NN 03/30/1848/Date ዓ.ም/NN ጦርነት/NN ገጠሙ/VB።/PUNC

From the above example, sentence 1 and sentence 2 should be selected to generate questions.

5.1.5 Question Transducer

In order to generate a list of possible questions from an input selected sentences, tasks such as unmovable phrases identification, answer phrase identification, transformation and post-processing should be done.

5.1.5.1 Mark Unmovable phrase

All phrases in a sentence may not be answer key phrases in a factual AQG task. Unmovable phrases are detected in this step, which makes it difficult to generate questions. Marking unmovable phrases usually solved using different NLP techniques such as PoS tagging or NER [45]. For this research, NER is used to detect this phrases by comparing the values of several lists of named entities with the phrases of input text. If phrase don't exist in the lists, they're considered

Unmovable. This means that the Unmovable phrases are not named entities and might be any other words in the sentence. Since this phrases are skipped over the next steps of answer phrase selection, the system does not generate questions for them.

5.1.5.2 Generate possible question phrase

In QG task, selection of appropriate question type starts by identifying answer key words from the selected sentence. Identifying domain specific words in the sentence is performed to select answer keys. In this stage the possible question phrases (who, where, when etc.) can be generated depending on the keys [46]. NER tool is used to select answer key. Named entity (person, number, location, organization or date) can be an answer keys which helps to generate possible type of questions.

For example:

From <<በካፋ/B-PER ህጻንነቱን/NN ያሳለፈው/V ወህኒ/B-LOC አምባ/I-LOC ላይ/PREP ነበር/V::/PUNC>>. From this sentence <<በካፋ>> and <<ወህኒ አምባ>> should be selected as answer key words based on its tagged named entities person and location respectivly. Then it generates possible question phrases “ማን” and “የት” depending on the key.

5.1.5.3 Transformation

Actual question should be constructed in grammatically correct natural language expression using a set of transformation rules. Applications for question generation mainly use transformation based methods by inserting question phrases for the corresponding answer keys to generate well-formulated questions [47]. After answer key selection using NER, the system removes the selected answer keys, and insert possible question phrases (“ማን”(who), “የት”(where), “መቼ”(when), “ስንት”(how much/many)) generated from the answer key phrases using a set of transformation rules. Questions phrases insert into a separate copy to generate new questions.

Rules: For Amharic language, there is no existing rules to construct questions from texts. This research defines rule-based technique to generate the reasonable questions from an input text. The rules are applied to generate questions such as -“የት” (where), “ማን” (who), “ስንት” (how much /how many). The transformations rules relies on Named Entity Recognition (NER) tool. The rule helps to remove answer keys and insert possible question phrases. The rules prepared for this research is as follow:

Rule 1:-if we found name of any location, city, country name or organization in given input sentence such as "ጅማ" (Jimma) replace it with **"የት"** (where).

Rule 2:- If any date format, year, time and the list of weeks or months or "ትላንት" (yesterday) or ነገ (tomorrow) or any integer type and the word is preceded by the word "ዓ.ም" (E.C), "ዓ.ዓ" (B.C) is found in given input sentence the it replace with **"መቼ"** (when).

Rule 3:-if we found any person name such as "ዘሚካኤል" in the given input sentence, we replace it with **"ማን"** (who).

Rule 4:-if we found any cardinal or ordinal or integer or the numbers in word such as "100","12", then we replace it with **"ስንት"** (How much /How many).

Rule 5:-If question has been generated, then the punctuation marks replace with "?" (Question mark)

Figure 5.1 shows how questions are constructed using rules respect with named entities.

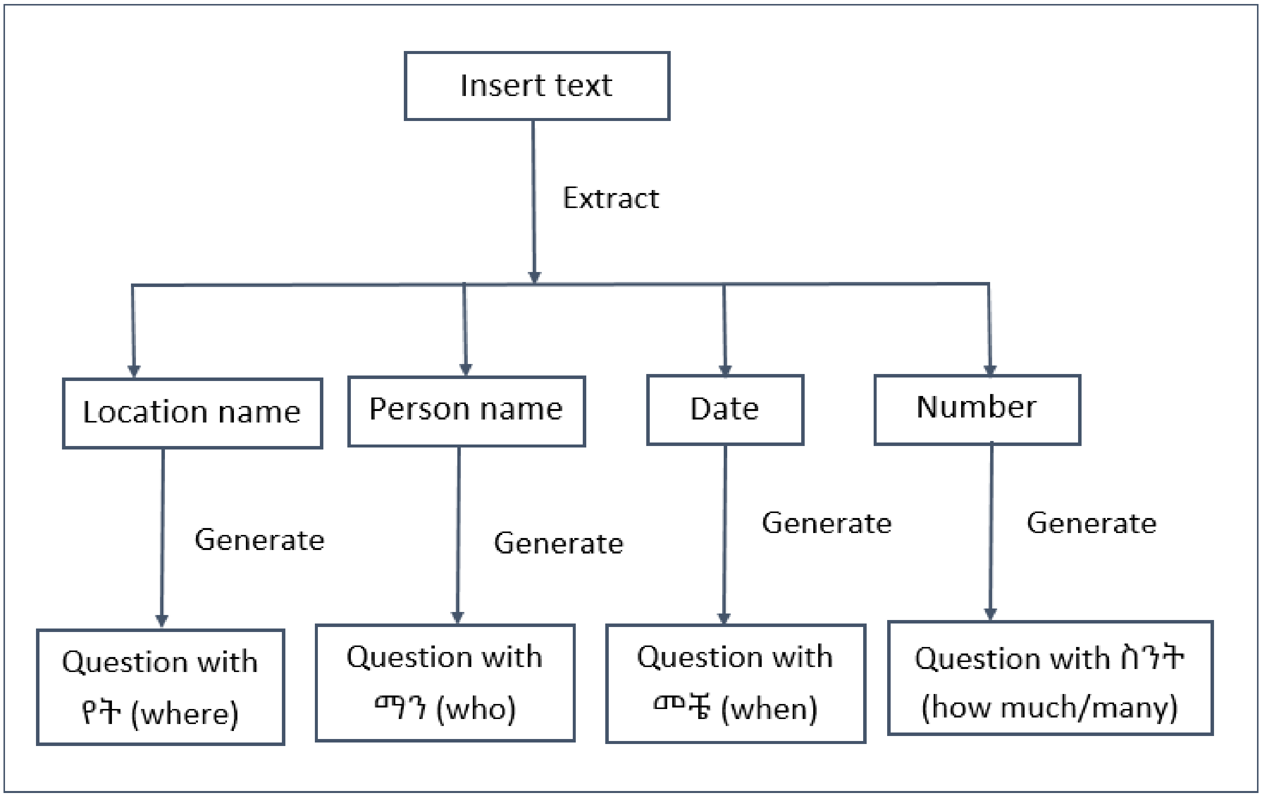

Figure 5.1 Question Generation flow chat from input sentence

The answer keys are extracted based on Named entities from input sentence and generate possible question phrase. Each answer keys remove and insert appropriate questions, for each question in a new line. Sample text and possible questions generated from the text at Appendix 5.

5.1.5.4 Post-processing

After generating questions some additional post-processing techniques are necessary to ensure proper formatting and punctuation. Such as de-segmentation, de-tokenized and remove white space /symbols, change four points (::) to question marks ate the end of each question. The sample question construction python code and AAQG system interface are shown at Appendix 7 and 8 respectively.

Chapter Six

Experiment and Evaluation

6.1. Experimental Environment

To conduct the experiment, we collected around 300 simple sentences from various sources that covers Ethiopian history. After the corpus collected, the first task was relevant sentence selection using Named entity recognition features based on rules. From the selected sentence we extract answer keywords using NER. After answer key selection, using transformation rules we replaced answer keywords by corresponding question phrases to construct questions. Then the questions evaluated using human expert with recall, precision and f-measure metrics.

The system is developed and tested using windows 10 Enterprise as an operating system. The hardware component comprises of Intel core i7 CPU of 2.7 GHz, 8 GB memory, and 1TB hard disk. The Python programming language (python 3.8) is used for developing the tools used in this research such as the POS tagger and the NER. In addition, the relevant sentence selection, and question creation are implemented using python.

6.2. Metrics of evaluation

To evaluate our simple sentence selection rule, we used accuracy. Accuracy [64] used to evaluate natural language generation. For question generation, this research used accuracy scores based on human evaluator. To evaluate POS and NER, we used automatic evaluation techniques by comparing annotated data with human annotator.

6.3. POS tagger evaluation

The purpose of evaluation is to determine the value of a solution to a particular problem. Since, we used HMM based statistical methods PoS tagger to be used by the NER, the developed tagger should be evaluated. For this study, we have done two experiments.

The first experiment used 8,075 WIC news sentences of manually tagged by ELRC staff member. For this experiment, 80/20 train/test dataset splitting is done. 6,460 sentences with tagged tokens used for training data which is tokenized at the sentence level, and for testing data, we utilized 1,615 sentences with untagged tokens tokenized at the word level, with each sentence on a new line. The tagger performance is evaluated automatically using accuracy measure by comparing the new tagged tokens with manually tagged tokens. The tagger achieved 78.25% of accuracy.

The second experiment performed by combining 8,075 WIC news sentences of manually tagged corpus and 11,843 (simply taking top sentences) Amharic web corpus tagged by TreeTagger. The dataset has been split with 80/20 train/test. 15,934 sentences with tagged tokens used for training data which is tokenized at the sentence level and for testing data, we used 3,984 sentences with untagged tokens tokenized at word levels, with each sentence on a newline. The tagger performance come up with 84.6%. This result is great for our next NE Classifier work since we did not use different features to enhance the performance of PoS tagger because our aim is not improving the performance of the tagger but develop question generation system. The researcher believes that, the performance of the tagger can improve by increasing training data. Furthermore, the type of data has an impact on the performance of the tagger since we used historic corpus. As we tried to find some words like "ደጃዝማች","ንጉሰ ነገስት" etc not found in the first experiment training dataset but found in the second training dataset. Since the development of NER used PoS tagged data, the performance of PoS tagged data is directly related to NER.

6.4. NER evaluation

NER is an important tool for content selection and possible question phrase generation. The tool is evaluated by comparing the output of system annotated data to the human annotated test data. For testing the performance of the NER system, from 20,163 sentences with tagged tokens, 16,130 used for training data which is tokenized at sentence level and 4,032 sentences with untagged tokens which is tokenized in word levels with separated by white space and also each sentence on a new line are used. Then the performance of the system done by comparing the new tagged data with manually tagged data development. Then the system is evaluated by the correctness of the classification tag using accuracy by comparing the new tagged data with manually tagged data and got an accuracy of 82.0%. The size of training data we used has an effect on the accuracy of the system since the training data does not cover large amount of tagged data specifically historic data, the system did not tag appropriately. For instance, the training data does not contain

“እቴጌ”,”ነገስታት” etc. therefore the system does not correctly tagged these tokens. In addition, the types of data has an impact on the accuracy of the NER system since we used a historical text that contains different names of people, date, places and organizations that are not encountered in the training data and the NE tagger might just tag incorrectly.

6.5. Content selection evaluation

Question generation targets are selected using simple selection rule based on NER. According to Sedigheh et al [93] , sentence selection methods evaluated using precision, recall, accuracy, and F1scores. This research used accuracy to evaluate our sentence selection method. For the purpose of this experiment, we used 100 simple sentences from the collected Ethiopian historic corpus as input to the system and generate 86 relevant sentences. To evaluate relevant sentence selection method, we present those selected relevant sentences for Human evaluator. Then, the evaluator classified those sentences as ‘1’ (relevant) and 0 (not relevant) at the end of each sentence. With consideration human evaluator select sentences as relevant, if any of the named entities such as location, organization, date, count and person are present in the sentence. Evaluators manually label sentences either relevant or not (see table 6.1).

Table 6.1 Sample sentence selection

No	**Sentences**	Relevance based on NER
1	ዓጼ ቴወድሮስ የቋራ አስተዳዳሪ የነበሩት ደጃዝማች ኃይሉ ወልደ ጊዮርጊስ ልጅ ናቸው ።	1
2	ብዙ ሳይቆይም በመማረኩ ወደፊት እንዲይነስ አፍንጫው ላይ ጠባሳ ተደረገበት ።	0
3	በካፋ ህጻንነቱን ያሳለፈው ወህኒ አምባ ላይ ነበር ።	1
4	በ1966 ዓ.ም የሶማሊያ መሪ የነበረው ሜጀር ጄኔራል ሲያድ ባሬ ኢትዮጵያን ወረረ ።	1
…	…	…..

After getting the evaluator's result, we used an accuracy measure to calculate the correctness of informative sentence selection systems. It defined as the number of correctly selected relevant sentences dividing by total number of relevant sentence generated by the system.

We measured the evaluation using accuracy as follows:

$$Accuracy = \frac{\mathbf{Scsr}}{\mathbf{Strs}} \mathbf{*100}$$

Scsr = *Number of correctly selected relevant sentences*

Strs =*Total number of relevant sentence generated by system*

Where, Scsr is the number of correctly selected relevant sentences by the system which is cheeked by human evaluator and Strs are number of selected sentences as relevant by the system. In the table 6.2 below shows the number selected relevant and irrelevant sentences by the human.

Table 6.2 sentence selection evaluation

Total number of input sentences	**Number of relevant sentences by the system**	**Correctly selected relevant sentences**	**Accuracy**
100	86	82	95.3%

We found that the accuracy of sentence selection system for Amharic text is 95.3 %. The selected sentences are used for Amharic question generation. 4.7 % of the sentence selected as relevant but based on human judgment are not relevant sentence.

6.6. Questions evaluation

For the evaluation of the questions, we generate questions using the system from the selected relevant sentences from collected Ethiopian historic corpus. We used the 86 selected relevant sentences from the corpus as input for the system and it generates 155 question, with pronoun questions related to 'ማን?',' የት?', 'መቼ?' And 'ስንት?'. The system can generate more than one question based on the existence of named entities in the sentence. For instance, if a sentence contain+ person name, location name and date, the system generates question types 'ማን?,' የት?' and 'መቼ?' respectively. In addition, to generate more questions and also to improve grammar and semantics errors, word segmentation in noun conjunction and noun preposition words has been performed. This helps to generate questions such as 'ማንን?,'በማን?, ከየት?, 'በስንት? etc after De-segmentation in post-processing stage.

Then by taking the generated questions and presented to a human evaluator to classify the questions either correct or incorrect. Prepositional questions (e.g. **የማን**) are considered their respective pronoun questions (e.g. **ማን**). Then we used widely used evaluation measure: Accuracy**.** We define Accuracy [64] as follows:

$$Accuracy = (\frac{\mathbf{Qc}}{\mathbf{Qt}})\ \mathbf{*100}$$

Qc = $Correct\ number\ of\ Question$

Qt = $Total\ number\ of\ Questions$

Where, Qc is the number of correctly generated questions by our system as human evaluator and Qt is the number of totally generated question by the system. In table 6.3 can be seen the result of the experiment with the Accuracy measure for different question types generated by the system and also over all accuracy of the system.

Table 6.3 Accuracy Evaluation Result

Question Type	Total number of Questions	Correct number of Questions	Incorrect number of Questions	Accuracy
ማን	60	55	5	91.6%
መቼ	24	20	4	83.3%
የት	37	27	10	73.0%
ስንት	34	32	2	94.1%
Overall	155	134	21	86.4%

From table 6.3 the generated questions "ስንት" (how much/many) got high score and "የት" (where) scores lowest accuracy. In addition, the overall accuracy of our question generation system got 86.4 %. The generated questions with question phrase "የት" got lowest score. Since Amharic sentences can be formed in a free order but gives the same meaning, which affects the possible question phrase generation based on their named entities. Question phrases generated based on the corresponding answer keywords using named entities, which might not always create correct questions, especially for question phrase “የት” (where). For instance: from the sentence <<አበበ ቢቂላ 1925 ዓ.ም በደብርሃን ከምትባል ከተማ ተወለደ::>> the system generates question << አበበ ቢቂላ በ1925 ዓ.ም የት ከምትባል ከተማ ተወለደ?>> the next phrase “ከምትባል” affects the question phrase “የት” (where) which depends on the tagged as location “በደብርሃን”. This question has grammatical error. To generate grammatically correct question from the sentence, the question phrase should change from “የት” (where) to “ማን” (who). For example: << አበበ ቢቂላ በ1925 ዓ.ም ማን ከምትባል ከተማ ተወለደ?>>. In addition, the question can be corrected by removing phrases “ከምትባል” and used question phrase “የት” (where) generated based on its answer keyword phrase(location). For instance: << አበበ ቢቂላ በ1925 ዓ.ም የት ከተማ ተወለደ?>>.

In addition, some generated questions have grammar or semantics errors due to in appropriate NER tagging. For instance, from the sentence <<በጦርነቱም ብዙ ሰዎችን አጥቷል፡፡>> ብዙ tagged as a person (per). As a result, the generated questions from the sentence becomes error. For example, <<በጦርነቱም ማን ሰዎችን አጥቷል?>>. This question is grammatically incorrect. The performance of the system affected by the performance of NER.

For this study, word segmentation on “noun with preposition” and “noun with conjunction” word classes has been performed to generate more useful and meaningful questions. We identified some Amharic prefixes and suffixes latters and segmented them, which are mostly found in large words, particularly on named entity words. For example, the sentence <<የኢትዮጵያ ጦር የዚያድ ባሬን ጦር አሳዶ ከመሬቱ አስወጣ::>> segmented at word level as <<**የ** ኢትዮጵያ ጦር **የ** ዚያድ ባሬ **ን** ጦር አሳዶ ከመሬቱ አስወጣ::>> with latters **“የ”** and **“ን”** from words at prefix and suffix. This helps to generate questions from the sentence like <<የኢትዮጵያ ጦር **የማንን** ጦር አሳዶ ከመሬቱ አስወጣ ?>>.

6.7. Findings

The performance of the system evaluated at different stages, since the performance of one stage affects the performance of the next stage. After evaluated different components of the system, the overall performance the system have been evaluated. The accuracy of PoS tagger, NER and informative sentence selection have been computed. The accuracy of PoS tagger come up with 84.6%. The size of training dataset and type of data has an effect on the result since the training data does not cover large amount of historic data. As we have seen in the second experiment, the training data contains more historical data which helps to improve the performance of the tagger. The PoS tagger performance has an effect on NER system, since the NER developed by adding, replacing, or changing part of speech. The experimental results of NER got the accuracy of 82.0%.

The performance of NER system has a great impact to determine informative sentences and generate possible question phrases. To identify informative sentences from the collected simple sentence, NER and certain rules have been used. The system correctly selected 82 simple sentences from 86 selected sentences by the system based on human evaluator. NER has an effect on informative sentence selection. For example: from the sentence <<በጨቅላ ዕድሜው ጥሩ የገና ተጫዋች ነበር>> “ጥሩ” is wrongly tagged as person (per) name but is an adjective (good). The informative sentence selection methods select the sentence as informative sentence wrongly. As a result, the questions generated from the sentence becomes wrong.

As we have explained previously, the performance of question generation system highly depends on the NER tools. In Amharic some person names are adjectives or vice versa. For example, from the sentence <<ቴዎድሮስ መልካም ተዋጊ ነበር፡፡ >> “መልካም” tagged as person (per) incorrectly As a result, the generated questions becomes incorrect. For example, instead of the question **<<ማን** መልካም ተዋጊ ነበር?>> it becomes **<<ማን** ተዋጊ ነበር?>> incorrectly.

From the experiment, some question types showed good results but other question types showed low results. The accuracy of questions generated by question phrase “ስንት” has got high score and questions generated by question phrase “የት” has got lower score. Sentence with named entity location, which might not always take question phrase “የት” to construct grammatically and semantically correct questions. For example: << አበበ ቢቂላ 1925 ዓ.ም **ደነባ** ከምትባል ስፍራ ተወለደ::>> from the sentence, question can be constructed based on their named entity location **(“ደነባ”)** such as <<አበበ ቢቂላ 1925 ዓ.ም **የት** ከምትባል ስፍራ ተወለደ ?>> this sentence has grammatical error. Instead of “የት”, using “ማን” question phrase can create grammatically correct sentence.

In this study, word segmentation helps to add noun with preposition (NP) and noun with conjunction (NC) word classes to generate questions such as prepositional and conjunctional questions. It helps to generate correct questions.

Finally, based on human judgment the overall accuracy of the system computed by dividing correctly created question by the system by total number of questions generated by the system and got 86.4%.

Chapter Seven

Conclusion and Recommendation

7.1. Conclusion

Automatic question generation (AQG) is an important yet challenging problem in natural language processing (NLP) [3]. In this research, an attempt was made to apply rule-based approach for the implementation of automatic Amharic factual question generation from historical texts. The system uses manually defined rules and collected simple historic texts from the web, books and newspapers.

The Automatic Amharic factual question generation was implemented using python programming language. The system uses manually defined rules along with NER tagged simple sentences. To generate questions, informative sentences are selected using NER and a certain rules. The NER system also used to identify answer keywords from the sentence. After determining answer keywords, using transformation rules, we removed answer key words and insert possible question phrases in the sentences.

The system evaluated at different stages such as PoS, NER and informative sentence selection methods. The accuracy of each type of questions is computed. The system shows good results for some question types but for other question types the system shows low results. The accuracy of questions created by question phrase **"ስንት"** got high score but questions created by question phrase **"የት"** scores low accuracy. The number of questions created by question phrase **"ማን"** is higher from others. In addition, overall performance of the system for each type of question is evaluated using human evaluator and achieved 86.4% accuracy.. The accuracy of the system affected by different stages of the system. Word segmentation also helps to add more word classes to generate correct questions. Increasing the size of domain specific training data improves the performance of generating possible question phrases that helps to increase the performance of our system.

Based on the experiment results of the study, the following conclusions are made:

- From the evaluation results found, it was shown that large coverage of domain specific training dataset helps to improve the performance of PoS tagger which helps to develop

NER. The tagger come up with 84.6% accuracy which is a good result to construct questions since our main aim is not to improve the performance of the tagger.

- Defining rules with different word classes helps to construct more correct questions. From the experiments, it was shown that word segmentation helps to add noun with preposition (NP) and noun with conjunction (NC) word classes which helps to generate prepositional and conjunctional questions.

In conclusion, the study shows that a good coverage of domain specific training dataset which helps to identify named entities in the sentence and defining more rules which includes more word classes has a great impact on automatic Amharic question generation system. More improvements to the implementation of AAQG using transformation rules are needed, such as handling exceptions and developing new rules to improve existing rules by adding more word classes.

7.2. Recommendation

In this thesis we designed an Amharic AQG that tries to generate factual question. However, in this attempt we have seen that generating such questions is not simple. After the experiment results shown, the researcher recommends that helps to enhance the performance of the system in the future works.

- When developing NLP systems, having a high-quality corpus is critical. As the corpus grows larger, the POS and NER tagger's performance will improve as well. As a result, researchers should prepare a corpus that is larger in size and of high quality in the topic of their research.
- Identifying and handling exceptions that stray from the defined rules can be also be a future work. For example, removing answer keywords and inserting question phrases has some exceptions. Incorporating such exception would greatly enhance the performance of the AAQG systems.
- Questions are generated from simple sentences. Due to time constraint, anaphora resolution and sentence simplification techniques are not included. Therefore, researchers include those techniques to generate Amharic questions automatically from the text needs to be considered in future studies.

- Automatic question generation systems has no common architecture and evaluation metrics. Therefore, researchers could prepare common architecture and evaluation metrics.
- Due to time constraint, Researches done on closed domain AAQG. As a result, researchers can work on open domain AQG. Furthermore, researches should be conducted by combining factual and non-factual AQG to create a full-fledged AQG.

References

[1] Vishal G, "System for Generating Questions Automatically from Given Punjabi Text," Punjabi University, Patiala, 2016.

[2] Ken P, Tuure T, Marcus A and Samir C, "Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems,* vol. 24, no. 3, pp. 45-78, 2008.

[3] Zhuang L, Kaiyu H and Degen H , "Semantics-Reinforced Networks for Question Generation," in *24th European Conference on Artificial Intelligence - ECAI*, Santiago de Compostela, Spain, 2020.

[4] David L, "AUTOMATIC QUESTION GENERATION FROM TEXT FOR SELF-DIRECTED LEARNING," Simon Fraser University, 2010.

[5] Ming L, Vasile R and Li L, "Automatic Chinese Factual Question Generation," Chongqing University, Chongqing, 2017.

[6] Poonam Gupta and Vishal Gupta, "Algorithm for Punjabi Question Answering System," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 3, no. 7, pp. 902-909, 2013.

[7] Miroslav Bl and Viera Rozinajov, "Machine Learning Approach to the Process of Question Generation," Information Systems and Software Engineering, Slovak University of Technology, 2017.

[8] Miroslav B and Viera R, "Automatic Question Generation Based on Analysis of Sentence Structure," Springer International Publishing , Switzerland, 2016.

[9] Martha Yifiru and Solomon Teferra , "Effect of Language Resources on Automatic Speech Recognition for Amharic," Addis Ababa University, Addis Ababa, 2015.

[10] Michael H, "Automatic Factual Question Generation from Text,Ph.D. Dissertation," Carnegie Mellon University, 2011.

[11] Haskell R, "Transfer of Learning: Cognition,Instruction, and Reasoning," Academic Press, 2011.

[12] BEZZA TESFAW, "THE SUBMORPHEMIC STRUCTURE OF AMHARIC: TOWARD A PHONOSEMANTIC ANALYSIS," University of Illinois at Urbana-Champaign, DISSERTATION, Urbana, Illinois, 2013 .

[13] Dehadari J, Tounsi L and Genabith J, "Morphological Features for Parsing Morphologically-Rich Languages: A Case of Arabic," *Proceedings of the 2nd Workshop on Statistical Parsing of Morphologically Rich Language,* pp. 12-21, 11 2011.

[14] A. Tilahun , "Amharic Question Answering For Definitional, Biographical and Description Questions," Addis Aababa University, Addis Aababa, 2013.

[15] Medihant G, "Amharic Question Answering for Factoid and List Questions using Machine-learning Approaches," Addis Ababa University, Addis ababa, 2019.

[16] Gasser M, "HornMorpho: A System for Morphological Processing of Amharic, Oromo, and Tigrinya.," in *Conference on Human Language Technology for Development*, Alexandria, 2011.

[17] Alemu B, "A Named Entity Recognition for Amharic," Addis Ababa university, Addis ababa, 2013.

[18] Andinet A, "Automatic Generation of Amharic Math Word Problem and Equation," *Journal of Computer and Communications,* vol. 8, pp. 59-77, 2020.

[19] SOLOMON A, "AMHARIC NAMED ENTITY RECOGNITION [ANER]: USING SUPERVISED APPROACH," Gonder University, Gonder, 2015.

[20] Iulian V, Alberto G, Caglar G, Sungjin A, Sarath C, Aaron C and Yoshua B, "Generating Factoid Questions With Recurrent Neural Networks: The 30M," in *University of Montreal*, Canada, 2016.

[21] Kamaldeep Kaur and v, "Name Entity Recognition for Punjabi Language," *International Journal of Computer Science and Information Technology & Security*, vol. 2, no. 3, VJune 2012.

[22] Khurana D, Koli A, Khatter and Singh S, "Natural Language Processing: State of The Art, Current Trends and Challenges," Manav Rachna International University,Accendere Knowledge Management Services Pvt. Ltd, Faridabad India.

[23] L. S., "What is the Question?," in *WORLD LIBRARY AND INFORMATION CONGRESS: 72ND IFLA GENERAL CONFERENCE AND COUNCIL*, California, 2006.

[24] R. S, "Automatic Question Generation using Discourse Cues and Distractor Selection for Cloze Questions,Thesis," Language Technology and Research Center (LTRC) International Institute of Information Technology, INDIA, 2012.

[25] HUSAM D and ALI A, "Automatic question generation : a syntactical approach to the sentence-to-question generation case," Lethbridge University, canada, 2012.

[26] Graesser C and Person K, "Question asking during tutoring," *Amer Edu. Res. J,* vol. 31, p. 104–137, 1994.

[27] Chali Y and Hasan A, "Towards Topic-to-Question Generation," University of Lethbridge, North America, 2015.

[28] Rus V, Cai Q and Graesser C, "Experiments on generating questions about facts," in *in Proc. 8th Int. Conf. Comput. Linguistics Intell. Text Process.*, 2007.

[29] Narendra A, Agarwal M and Shah R, "Automatic cloze questions generation. In Proceedings of Recent Advances in Natural Language Processing," Hissar, Bulgaria, 2013.

[30] Tahani A, Bijan P and Ulrike S, "Ontology-Based Multiple Choice Question Generation," University of Manchester, UK, 2015.

[31] Hoshino A and Nakagawa H, "A real-time multiple-choice question generation for language testing a preliminary study," Proceedings of the 2nd Workshop on Building Educational Applications Using NLP.University of Tokyo, June 2005.

[32] Stanescu L, Spahiu C and Spahiu A, "Question Generation for Learning Evaluation," in *International Multi-Conference on Computer Science and Information Technology, 509-513*, 2008.

[33] Kumar G, Rafael E and Luis F, "RevUP: Automatic Gap-Fill Question Generation from Educational Texts," in *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, Colorado, 2015.

[34] Ghader K, Jared L, Bijan P, Uli S and Salam A, "A Systematic Review of Automatic Question Generation for Educational Purposes," *International Journal of Artificial Intelligence in Education,* vol. 30, p. 121–204, 2020.

[35] Manish A and Prashanth M, "Automatic Gap-fill Question Generation from Text Books," in *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications, pages 56–64,*, Portland, 2011.

[36] Parshan S and Rajbhupinder K, "Rule Based Question Generation System from Punjabi Text Contain Historical Information," *International Journal of Computer Science and Mobile Computing,* vol. 3, no. 10, p. 86 – 91, 2014.

[37] Rus V and Graesser A, "THE QUESTION GENERATION SHARED TASK AND EVALUATION CHALLENGE," The University of Memphis, 2009.

[38] Nguyen-Thinh L, Tomoko K and Niels P, "Automatic Question Generation for Educational Applications – The State of Art," Universität zu Berlin, Germany, 2014.

[39] Sonam S, Kumar P and Saha A, "Automatic Question Generation: A Systematic Review," Information Systems &eBusiness Network, India, 2019.

[40] JANSSEN T, "Instruction in Self-Questioning as a Literary Reading Strategy: An exploration of empirical research," Kluwer Academic Publishers., Netherlands, 2014.

[41] Victoria H and August J, "Automatically Generate Questions for National Examinations in English Reading Comprehension," KTH ROYAL INSTITUTE OF TECHNOLOGY, SWEDEN, 2020.

[42] Husam A, Yllias C, Sadid A and Hasan, "Automation of Question Generation From Sentences," University of Lethbridge Proceedings of the Third Workshop on Question Generation, Canada.

[43] Blšták, Rozinajová and Viera, "Automatic question generation based onanalysisofsentencestructure," in *InternationalConferenceonText,Speech,and Dialogue*, 2016.

[44] Eriks Sneiders, "Automated Question Answering Using Question Templates that Cover the Conceptual Model of the database," Stockholm University and the Royal Institute of Technology, 2002.

[45] Ratinov and Dan , "Design Challenges and Misconceptions in Named Entity Recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL), pages 147–155*, USA, 2009.

[46] Bidyut D and Mukta M, "Factual open cloze question generation for assessment of learner’s knowledge," *nternational Journal of Educational Technology in higher education*, vol. 3, pp. 17-60, 2017.

[47] Heilman M and Smith N, "Question Generation via Overgenerating Transformations and Ranking," Carnegie Mellon University, 2009.

[48] Brown G, Frishkoff and Eskenazi M, "Automatic question generation for vocabulary assessment," in *in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing,Association for Computational Linguistics*, 2005.

[49] Sandra Williams, "Generating Mathematical Word Problems," Association for the Advancement of Artificial, 2011.

[50] Parshan S and Rajbhupinder K, "Rule Based Question Generation System from Punjabi Text Contain Historical Information," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 10, p. 86 – 91, October 2014.

[51] Kaur J and Kumar A, "AUTOMATIC QUESTION GENERATION FROM HINDI TEXT USING HYBRID APPROACH," in *second international confernce on science,technology and management*, India, 2015.

[52] Jaspreet Kaur and Kumar Bathla, "A REVIEW ON AUTOMATIC QUESTION GENERATION SYSTEM FROM A GIVEN HINDI TEXT," *INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS,* vol. 3, no. 6, pp. 87-92, 2015.

[53] "Designing a Rule Based Disambiguator for Afan Oromo Words," *American Journal of Computer Science and Information Technolog* , vol. 5 , no. 2, 2017.

[54] Goshtasby A, "Template matching in rotated images,," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 7, no. 3, p. 338–344, 1985.

[55] Manisha D and Ambuja S, "Automatic question generation approaches and evaluation techniques," University of Mumbai, india, NOVEMBER 2017.

[56] Colin Kelly, Ann Copestake and Nikiforos Karamanis, "Investigating Content Selection for Language Generation using Machine learning," Proceedings of the 12th European Workshop on Natural Language Generation, Athens,Greece, 2009.

[57] Miroslav Blšták, "Automatic Question Generation Based on Sentence Structure Analysis," Information Sciences and Technologies Bulletin of the ACM Slovakia, 2018.

[58] Rim H, Yook D, Lim H and Kim S, "Effective Methods for Improving Naive Bayes Text Classifiers," 2002.

[59] Zhang D and Lee W, "Question classification using support vector machines," in *SIGIR 03 Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, 2003.

[60] Nikolay Stanevski and Dimiter Tsvetkov, "Using Support Vector Machine as a Binary Classifier," in *International Conference on Computer Systems and Technologies - CompSysTech' 2005*, 2005.

[61] Anurag Sarkar, Saptarshi Chatterjee, Writayan Das and Debabrata Datta, "Text Classification using Support Vector Machine," *International Journal of Engineering Science Invention,* vol. 4, no. 11, pp. 33-37, 2015.

[62] D. Sonntag, "Distributed NLP and Machine Learning for Question Answering Grid.".

[63] Chopra D and Morwal S, "NAMED E NTITY R ECOGNITION I N E NGLISH USING HIDDEN MARKOV MODEL," 2013.

[64] Deepali K, Gaikwad and Namrata M, "Question Generation System for Marathi Text," *International Journal of Scientific Research in Computer Science,* vol. 3, no. 3, 2018.

[65] Jacopo A, Paul P and Alistair W, "Evaluation methodologies in Automatic Question Generation 2013-2018," in *Proceedings of The 11th International Natural Language Generation Conference, pages 307–317,*, UK, 2018.

[66] Rus V, Cai Z and Graesser A, "Evaluation in Natural Language Generation: The Question Generation Task.," Proceedings of the Workshop on Shared Tasks and Comparative Evaluation in Natural Language Generation, Hilton Arlington, 2007.

[67] Lin C, "Automatic question generation from quesries," in Workshop on Question Generation Shared Task, Pittsburg, 2008.

[68] "How do machine-generated questions compare to human-generated questions?," Research and Practice in Technology Enhanced Learning, USA, December 2016.

[69] Aldabe I, Dios I and IMaritxalar M, "Two Approach to Generate Question in Basque," 2013.

[70] Bernardo L, Henrique L, Lu´ıs P and Carlos S, "Factual Question Generation for the Portuguese language," in *University of Porto,IEEE*, portugal, 2020.

[71] YVONNE S, "AUTOMATIC GENERATION OF FACTUAL QUESTIONS FROM VIDEO DOCUMENTARIES,Ph.D. Dissertation," University of Wolverhampton, 2013.

[72] Onur K, "AUTOMATIC QUESTION GENERATION USINGNATURAL LANGUAGE PROCESSINGTECHNIQUES,masters thesis," Izmir Institute of Technology, 2018.

[73] " WIKIPEDIA," FROM WIKIPEDIA , [Online]. Available: http://www.lonweb.org/link-amharic.htm. [Accessed 06 01 2021].

[74] S. Eyassu, "Classifying Amharic News Text Using Self Organizing Maps," Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, 2005.

[75] Teshome K, WORD SENSE DISAMBIGUATION FOR AMHARIC TEXT RETRIEVAL: A CASE STUDY FOR LEGAL DOCUMENTS,Masters thesis, Addis Ababa: Addis Ababa University, 2009.

[76] ጌታሁን አማረ, ዘመናዊ የአማርኛ ሰዋስው በቀላል አቀራረብ, አዲስ አበባ: ንግድ ማተሚያ ቤት, 1990 ዓ.ም.

[77] ባየ ይማም, የአማርኛ ሰዋስው, አዲስ አበባ: ት.መ.ማ.ድ, 1987 (ዓ.ም).

[78] DAWED O, AMHARIC NAMED ENTITY RECOGNITION SYSTEM USING STATISTICAL METHOD, Bahirdar: Bahirdar University, 2020.

[79] Teshome, W, "Designing Amharic Definitive Question Answering," MA Thesis, Addis Ababa University, Addis ababa, 2013.

[80] Andersen S, "Sentence Types and Functions," San José State University Writing Center, 2014.

[81] Ayinadis temesigen, "Design and Development of Amharic Grammar Checker," Addis Ababa University, Addis Ababa, 2013.

[82] Frank M, Modern English : a practical reference guide, Englewood Cliffs : Prentice Hall, 1972.

[83] Russell S, "Amharic," In the Feinstein School of Education and Human Development, 2009.

[84] M. Asema, DESIGNING A KNOWLEDGE REUSE FRAMWORK FOR PROJECT BASED ORGANIZATIONS:THE CASE OF INFORMATION NETWORK SECURITY AGENCY, addid ababa: addis ababa university, 2020.

[85] Alan R, Salvatore T, Jinsoo P and Sudha R, "Design science in information systems research. MIS Quarterly: Management Information Systems," https://www.researchgate.net/publication/201168946, 2004 .

[86] Baskerville R, Baiyere A, Gregor S, Hevner A and Rossi M, "Design science research contributions: Finding a balance between artifact and theory," *Jornal of Association of Information Systems*, vol. 19, no. 5, p. 358–376, 2018.

[87] Demeke G and Getachew M, "Manual Annotation of Amharic News Items with Part-of-Speech Tags and Its Challenges. Ethiopian Languages Research Center Working Papers, 1-16.," Ethiopian Languages Research Center Working Papers, Addis ababa, 2006.

[88] E. L. R. Center, "Ethiopian Languages Research Center," [Online]. Available: http://habit-project.eu/wiki/SetOfEthiopianWebCorpora. [Accessed 20 6 /2021].

[89] U. A. a. C. S. Muljono, "Morphology analysis for Hidden Markov Model based Indonesian part-of-speech tagger," in *2017 1st International Conference on Informatics and Computational Sciences (ICICoS), pp. 237-240*, 2017.

[90] S. M. T. a. M. P. Harper, "A Second-Order Hidden Markov Model for Part-of-Speech Tagging," School of Electrical and Computer Engineering, Purdue University , 2013.

[91] Steven B, Ewan K and Edward L, Natural Language Processing with Python, United States of America: O'Reilly Media, Inc, 2014.

[92] Peffers K, Rothenberge M and Tuunanen T& Vae, "Design science research evaluation,," in *Proceedings of the 7th International Conference on Design:Advances in Theory and Practice*, 2012.

[93] Sedigheh M, Aijun A, Heidar D, Marjan D and Emad G, "Question-Worthy Sentence Selection for Question Generation," York University, Toronto Canada, 2020.

[94] Yan S, Christian H and Joakim N, "Universal Word Segmentation: Implementation and Interpretation," Uppsala University, 2018.

Appendix

Appendix 1: Amharic Alphabets

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

Appendix 2: Amharic Punctuation Marks

፡	-hulet netib	-Amharic word space
::	-mulu arat netib	-Amharic full stop
፣	-netela serez	-Amharic comma
፤	-dirib serez	-Amharic semi-colon
“ “	-temiherte tikes	-Amharic quotation mark
!	-timiherte ankro	-Amharic exclamation mark
()	-qinif	-Amharic bracket
_	-cheret	-Amharic underscore
-	-neus cheret	-Amharic Hyphen
…	-netebtab	-Amharic etc.
?	-temhrte tiyakie	-Amharic question mark
.	-yizet (netib)	-Amharic dot

Appendix 3: Amharic Numerals

፩	፪	፫	፬	፭	፮	፯	፰	፱		
1	2	3	4	5	6	7	8	9		
፲	፳	፴	፵	፶	፷	፸	፹	፺	፻	፼
10	20	30	40	50	60	70	80	90	100	1000

Appendix 4: Amharic interchangeable characters and their normal character

Interchangeable characters	Normalize to
ሀ፣ ሃ፣ ሐ፣ ሓ፣ ኅ፣ ኃ ፣ኻ	ሀ
ሁ፣ሑ፣ኁ፣ኹ	ሁ
ሂ፣ሒ፣ኂ፣ኺ	ሂ
ሄ፣ሔ፣ኄ፣ኼ	ሄ
ህ፣ሕ፣ኅ፣ኽ	ህ
ሆ፣ሖ፣ኆ፣ኾ	ሆ
ወ፣ዎ and ዉ፣ው	ወ and ው respectivly
ጨ፣ጬ and ጪ፣ጭ	ጨ and ጭ respectivly
ቸ፣ ቼ and ች፣ ቺ	ቸ and ች respectivly
ሠ፣ሰ፣ሡ፣ሱ፣ሢ፣ሲ፣ሣ፣ሳ፣ሤ፣ሴ፣ሥ፣ስ፣ሦ፣ሶ	ሰ፣ሱ፣ሱ፣ … ፣ሶ respectivly
ጸ፣ፀ፣ጹ፣ፁ፣ጺ፣ፂ፣ጻ፣ፃ፣ጼ፣ፄ፣ጽ፣ፅ፣ጾ፣ፆ	ፀ፣ፁ፣ጺ፣ …፣ፆ respectivly
አ፣ዐ፣ኣ፣ዓ	አ
ኡ፣ዑ	ኡ
ኢ፣ዒ	ኢ
ኤ፣ዔ	ኤ
እ፣ዕ	እ
ኦ፣ዖ	ኦ
ሸ፣ ሼ and ሽ፣ ሺ	ሸ and ሽ respectively
የ፣ ዬ and ዩ፣ ይ	የ and ይ respectively
ጎ፣ ጐ	ጐ
ኋ፣ ዃ፣ ሗ	ኋ
ኮ፣ ኰ	ኮ

Appendix 5: Sample text and possible questions generated from the text (Text 1 and possible generated questions 1)

Text 1

ሻምበል አበበ ቢቂላ

አበበ ቢቂላ 1925 ዓ.ም በደብርሃን ከተማ ተወለደ ። አበበ በ12 ዓመቱ የቄስ ትምሕርቱን አጠናቀቀ ። አበበ በጨቅላ ዕድሜው ስመ ጥሩ የገና ተጫዌች ነበር ። በዚያም በገና ጨዋታ እና በስፖርት እየዳበረ ቆየ ። አበበ ለኢትዮጵያና ጥቁር አፍሪቃ የመጀመሪያውን የኦሊምፒክ ማራቶን ወርቅን ከዓለም ክብረ ወሰን ጋር አስገኘ ። አበበ በሮማ የኦሊምፒክ ውድድር በባዶ እግሩ ሮጦ አሸነፈ ። አበበ 1973 ዓ.ም ከዚህ አለም በሞት ተለየ ።

የካራማራ ጦርነት

በ 1966 ዓ.ም ሶማሊያ የአረብ ሊግን ተቀላቀለች ። በ 1966 ዓ.ም የሶማሊያ መሪ የነበረው ሜጀር ጄኔራል ሲያድ ባሬ ኢትዮጵያን ወረረ ። ጦርነቱ የተጀመረ ሰሞን የ ሲያድ ባሬ ጦር በሶቪዬት ኅብረት ይደገፍ ነበር ። የኢትዮጵያ መንግሥት ደግሞ ከዩናይትስ ስቴትስ ጋር ወዳጅነት አለው ። የካራማራ ወሳኟ ቦታ ደግሞ አጋዴን ነበረች ። የጦርነቱ መባቻ ተደርጎ የሚቆጠረውም ይህ ጥቃት ነበር ። በመቀጠል የዚያድ ባሬ ጦር በሶቪዬት ኅብረት የጦር መሣሪያ ታግዞ ወደ ኢጋዴን ገሰገሰ ። በ 1966 ዓ.ም ኢትዮጵያ ከንጉሡ አገዛዝ ወደ ወታደራዊው ደርግ ተሻጋገረች ። በ ካራማራ ጦርነት አሜሪካ ድጋፏን ለ ሶማሊያ አደረገች ። ሁለቱ አገራት ወደ ቀለጠ ጦርነት ገቡ ። በወቅቱ የ ሲያድ ባሬ መንግሥት ኩባውያን የኢትዮጵያውን ጦር ደግፈው እየተዋጉ ነው ሲል ወቀሰ ። ይህንንም ለማሳመን ሲሉ የኩባ ወታደሮች መያዝ ጀመሩ ። በ ካራማራ ጦርነት ኩባዊው ወታደር ካርሎስ ኦርላንዶ ተይዞ ለጋዜጠኞች መግለጫ እንዲሰጥ ተገደደ። የ ዚያድ ባሬ መንግሥት ወራራውን አጠናክሮ በ ጅግጅጋ በኩል ደንበር አልፎ ገብቶ ብዙ ጥፋት አደረሰ ። የ ዚያድ ባሬ መንግሥት ሐረር ሲገባ የኢትዮጵያ ክተት የታወጀው ። ሐረር ላይ 40 ሺህ ገደማ የኢትዮጵያ ጦር ኃይል ጠበቀው ። ተጨማሪ 10 ሺህ የኩባ ጦረኞች ከኢትዮጵያ ጎን ነበሩ ። የኢትዮጵያ ጦር የዚያድ ባሬን ጦር አሳዶ ከመሬቱ አስወጣ ። የመጨረሻው ውጊያ የነበረው ካራማራ ላይ ነበር ።

ደግማዊ አፄ ምኒልክ

ምኒልክ በ1836 ዓ.ም ደብረብርሃን አካባቢ ተወለዱ ። የምኒልክ አባት የሸዋ ንጉሥ ነበር ። ምኒልክ እስከ ሰባት ዓመታቸው መንዝ እናታቸው ጋር አደጉ ። ምኒልክ በ1848 ዓ.ም መቅደላ ገቡ። ምኒልክ በ1856 ዓ.ም አገቡ ።ምኒልክ1859 ዓ.ም ከልቼ ዘመቱ ።

Possible generated questions 1

Q1: **ማን** 1925 ዓ.ም በደብርሃን ከምትባል ከተማ ተወለደ ?

Q2: አበበ ቢቂላ በ1925 ዓ.ም **የት** ከምትባል ከተማ ተወለደ ?

Q3: አበበ **በስንት** ዓ.ም በደብርሃን ከተማ ተወለደ ?

Q4: አበበ **በስንት** ዓመቱ የቄስ ትምሕርቱን አጠናቀቀ ?

Q5: **ማን** በ12 ዓመቱ የቄስ ትምሕርቱን አጠናቀቀ ?

Q6: **ማን** በጨቅላ ዕድሜው ስመ ጥሩ የገና ተጫዌች ነበር ?

Q7: **ማን** ለኢትዮጵያና ጥቁር አፍሪቃ የመጀመሪያውን የኦሊምፒክ ማራቶን ወርቅን ከዓለም ክብረ ወሰን ጋር አስገኘ ?
Q8: **ማን** በሮማ የኦሊምፒክ ውድድር በባዶ እግሩ ሮጦ አሸነፈ ?

Q9: **ማን** 1973 ዓ.ም ከዚህ አለም በሞት ተለየ ?

Q10: አበበ **በስንት** ዓ.ም ከዚህ አለም በሞት ተለየ ?

Q11: **በስንት** ዓ.ም ሶማሊያ የአረብ ሊግን ተቀላቀለች ?

Q12: **በስንት** ዓ.ም የሶማሊያ መሪ የነበረው ሜጀር ጄኔራል ሲያድ ባሬ ኢትዮጵያን ወረረ ?

Q13: ጦርነቱ የተጀመረ ሰሞን **የማን** ጦር በሶቪዬት ኅብረት ይደገፍ ነበር ?

Q14: **የማን** መንግሥት ደግሞ ከዩናይትስ ስቴትስ ጋር ወዳጅነት አለው ?

Q15: በመቀጠል የዚያድ ጦር በሶቪዬት ኅብረት የጦር መሣሪያ ታግዞ ወደ **የት** ገሰገሰ ?

Q16: በመቀጠል **የማን** ጦር በሶቪዬት ኅብረት የጦር መሣሪያ ታግዞ ወደ ኦጋዴን ገሰገሰ ?

Q17: በ1966 ዓ.ም **ማን** ከንጉሡ አገዛዝ ወደ ወታደራዊው ደርግ ተሻጋገረች ?

Q18: **በስንት** ዓ.ም ኢትዮጵያ ከንጉሡ አገዛዝ ወደ ወታደራዊው ደርግ ተሻጋገረች ?

Q19: **በየት** ጦርነት አሜሪካ ድጋፏን ለሶማሊያ አደረገች ?

Q20: በካራማራ ጦርነት **ማን** ድጋፏን ለሶማሊያ አደረገች ?

Q21: **በየት** ጦርነት ሁለቱ አገራት ወደ ቀለጠ ጦርነት ገቡ ?

Q22: በካራማራ ጦርነት **ስንት** አገራት ወደ ቀለጠ ጦርነት ገቡ ?

Q23: በወቅቱ **የማን** መንግሥት ኩባውያን የኢትዮጵያውን ጦር ደግፈው እየተዋጉ ነው ሲል ወቀሰ ?

Q24: ይህንንም ለማሳመን ሲሉ **የማን** ወታደሮች መያዝ ጀመሩ ?

Q25: **በየት** ጦርነት ኩባዊው ወታደር ካርሎስ ኦርላንዶ ተይዞ ለጋዜጠኞች መግለጫ እንዲሰጥ ተገደደ ?

Q26: የ ዚያድ ባሬ መንግሥት ወራራውን አጠናክሮ **በየት** በኩል ደንበር አልፎ ገብቶ ብዙ ጥፋት አደረሰ ?

Q27: **የማን** መንግሥት ወራራውን አጠናክሮ በ ጅግጅጋ በኩል ደንበር አልፎ ገብቶ ብዙ ጥፋት አደረሰ ?

Q28: የዚያድ ባሬ መንግሥት **የት** ሲገባ የኢትዮጵያ ክተት ታወጀ ?

Q29: **የማን** መንግሥት ሐረር ሲገባ የኢትዮጵያ ክተት ታወጀ ?

Q30: **የት** ላይ 40 ሺህ ገደማ የኢትዮጵያ ጦር ኃይል ጠበቀው ?

Q31: ሐረር ላይ **ስንት** ሺህ ገደማ የኢትዮጵያ ጦር ኃይል ጠበቀው ?

Q32: ተጨማሪ **ስንት** ሺህ የ ኩባ ጦረኞች ከኢትዮጵያ ጎን ነበሩ ?

Q33: ተጨማሪ 10 ሺህ **የማን** ጦረኞች ከኢትዮጵያ ጎን ነበሩ ?

Q34: የኢትዮጵያ ጦር **የማን** ጦር አሳዶ ከመሬቱ አስወጣ ?

Q35: የመጨረሻው ውጊያ የነበረው **የት** ላይ ነበር ?

Q36: ምኒልክ በ1836 ዓ.ም **የት** አካባቢ ተወለዱ ?

Q37: **ማን** በ1836 ዓ.ም ደብረብርሃን አካባቢ ተወለዱ ?

Q38: ምኒልክ **በስንት** ዓ.ም ደብረብርሃን አካባቢ ተወለዱ ?

Q39: የምኒልክ አባት **የየት** ንጉሥ ነበር ?

Q40: **የማን** አባት የ ሸዋ ንጉሥ ነበር ?

Q41: ምኒልክ እስከ ሰባት ዓመታቸው **የት** እናታቸው ጋር አደጉ ?

Q42: **ማን** እስከ ሰባት ዓመታቸው መንዝ እናታቸው ጋር አደጉ ?

Q43: ምኒልክ በ1848 ዓ.ም **የት** ገቡ ?

Q44: **ማን** በ1848 ዓ .ም መቅደላ ገቡ ?

Q45: ምኒልክ **በስንት** ዓ.ም መቅደላ ገቡ ?

Q46: **ማን** በ1856 ዓ.ም አገቡ ?

Q47: ምኒልክ **መቼ** አገቡ ?

Q48: ምኒልክ 1859 ዓ.ም **የት** ዘመቱ ?

Q49: **ማን** 1859 ዓ.ም ከልቼ ዘመቱ ?

Q50: ምኒልክ **መቼ** ከልቼ ዘመቱ ?

Appendix 6: sample preprocessing python code

```
userinput=input('ጥያቄ የምታወጠበትን ፋይል ይጨምሩ:')

myfile=open(userinput,"r",encoding='utf-8')

userInputFile= myfile.read()

text=userInputFile

print('\n')

###########  Normalization of characters   #################

simmilar_characters=
{"ሐ":"ሀ","ሑ":"ሁ","ሒ":"ሂ","ሓ":"ሃ","ሔ":"ሄ","ሕ":"ህ","ሖ":"ሆ",\

"ሃ":"ሀ","ሓ":"ሀ","ኀ":"ሀ","ኃ":"ሀ","ኻ":"ሀ",\


"ኀ":"ሀ","ኁ":"ሁ","ኂ":"ሂ","ኃ":"ሃ","ኄ":"ሄ","ኅ":"ህ","ኆ":"ሆ",\


"ሠ":"ሰ","ሡ":"ሱ","ሢ":"ሲ","ሣ":"ሳ","ሤ":"ሴ","ሥ":"ስ","ሦ":"ሶ","ሧ":"ሷ"
,\
```

```
"ዐ":"አ","ዑ":"ኡ","ዒ":"ኢ","ዓ":"ኣ","ዔ":"ኤ","ዕ":"እ","ዖ":"ኦ",\

"ጸ":"ፀ","ጹ":"ፁ","ጺ":"ፂ","ጻ":"ፃ","ጼ":"ፄ","ጽ":"ፅ","ጾ":"ፆ"}
def
preprocess(text,replace_characters=True,replace_punctuation=True
,remove_last=0):
  # Replace punctuation
    if replace_punctuation:
        text = text.replace('(', ' ')
        text = text.replace(')', ' ')
        text = text.replace('--', ' ')
        text = text.replace('፣', ' ')
        text = text.replace(':', '')
        text = text.replace('፤',' ')
        text = text.replace('…',' ')
        text = text.replace('[',' ')
        text = text.replace(']',' ')
    if replace_characters:
        for letter, alternative in simmilar_characters.items():
            text = text.replace(letter, alternative)
    words = text.split()
    word_counts = Counter(words)
    trimmed_words = [word for word in words if word_counts[word]]
    return trimmed_words
words = preprocess(text)
listToStr = ' '.join(map(str, words)) # list to strig Conversion
```

```
with   open("02_preproc_Am_raw_input.txt",mode='w',encoding='utf-8') as fh:

    fh.writelines(listToStr)

    fh.close()

with   open("02_preproc_Am_raw_input.txt",mode='r',encoding='utf-8') as prep_txt:

    read_prep_txt=prep_txt.read()

    text=read_prep_txt.replace('። ', '።\n')

    text=re.sub('  በ','  በ  ',text)  #  segment  prepositional  and conjunctional phrases

    text=re.sub(' የ',' የ ',text)

    text=re.sub('ን ',' ን ',text)

    text=re.sub(' ለ',' ለ ',text)

    text=re.sub(' ከ',' ከ ',text)

    text=re.sub('ና ',' ና ',text)

    text=re.sub(' እነ',' እነ ',text)

    #print(text)

with open("02_preproc_Am_raw_input.txt", "w",encoding='utf-8') as filehandle:

    filehandle.writelines(text)

    filehandle.close()
```

Appendix 7: sample question generation python code

```
keys_to_replace    =    {'የት':'loc','ስንት':'num','ማን':'per','መቼ':'date'}

# Iterate over all key-value pairs in dictionary

file= open("05_selected_sentences.txt", "r", encoding='utf-8')
```

```
count=0

line = file.readline()   # include newline

while line:

    count += 1

    line = line.rstrip()

    line = file.readline()

    for key, value in keys_to_replace.items():

        # Replace key character with value character in string

        ans_key_replace = line.replace(value,key)

        ans_key_replace=ans_key_replace.split()

        for pls in ans_key_replace:

            if any( [pls == 'ማን', pls == 'መቼ', pls ==
'የት',pls=='ስንት'] ):

                indx=ans_key_replace.index(pls)-1

                del ans_key_replace[indx]

                indx +=1

                postProcess_questions = [w for w in
ans_key_replace if not w in entities] #Remove unnecessary taggs
and spaces

                listToStr = ' '.join(map(str,
postProcess_questions)) # list to strig Conversion

                addquestionmark=listToStr.replace('።','?')

                addquestionmark=re.sub(' በ ',' በ',addquestionmark)
# De-segmentation

                addquestionmark=re.sub(' የ ',' የ',addquestionmark)

                addquestionmark=re.sub(' ን ','ን ',addquestionmark)

                addquestionmark=re.sub(' ለ ',' ለ',addquestionmark)
```

```
addquestionmark=re.sub(' ከ ',' ከ',addquestionmark)

addquestionmark=re.sub(' ና ','ና ',addquestionmark)

addquestionmark=re.sub(' እን ',' እን',addquestionmark)

with open("06_Generated_Questions.tx", "w",encoding='utf-8') as filehandle:

    filehandle.write(addquestionmark)

filehandle.close()

print("Q{}:{}".format(count, addquestionmark.strip()))
```

Appendix 8: Automatic Amharic Question Generation System Interface

```
#####################################################################################
#####################################################################################
##                                                                                 ##
##                 ADDIS ABABA UNIVERSITY COLLEGE OF NATURAL                       ##
##                       AND COMPUTITIONAL SCIENCE                                 ##
##                      SCHOOL OF INFORMATION SCIENCE                              ##
##                                                                                 ##
##                                ጥያቄ ያውጡ                                        ##
##                                                                                 ##
##                                                                                 ##
#####################################################################################
#####################################################################################


ጥያቄ የምታወጠበትን ፋይል ይጨምሩ:

ጥያቄ የምታወጠበትን ፋይል ይጨምሩ:E:\2013_Document\AAFQG\04_Demonstration\01_Am_User_input.txt


Q1: አበበ በስንት ዓመቱ የቄስ ትምህርቱን አጠናቀቀ ?
Q2: ማን በጨቅላ ዕድሜው ስመ ጥሩ የገና ተጫዋች ነበር ?
Q3: አበበ ስንት ዓ.ም ከዚህ አለም በሞት ተለየ?
```