Seid Muhie Yimam
Haramaya University
Haramaya, Ethiopia
seidyimam@haramaya.edu.et,
seidymam@gmail.com

Mulugeta Libsie
Department of computer Science
Addis Ababa University
Addis Ababa, Ethiopia
mlibsie@cs.aau.edu.et, mlibsie@yahoo.com

Amharic Question Answering (AQA)

ABSTRACT: The number of Amharic documents on the Web is increasing as many newspaper publishers started providing their services electronically. People were relying on IR systems to satisfy their information needs but it has been criticized for lack of delivering "readymade" information to the user so that Question Answering systems emerge as the best solution to get the required information to the user with the help of information extraction techniques. In this paper we have developed a Question Answering System for Amharic (AQA). The language specific issues in Amharic are extensively studied and hence, document normalization was found very crucial for the performance of our Question Answering system. The performance on normalized documents is found to be higher than the un-normalized ones. Novel technique were developed to determine the question types, possible question focuses, and expected answer types as well as to generate proper Information Retrieval query, based on our language specific issue investigations. An approach in document retrieval focuses on retrieving three types of documents (Sentence, paragraph, and file). An algorithm has been developed for sentence/paragraph re-ranking and answer selection. The named-entity-(gazetteer) and pattern-based answer pinpointing algorithms developed help locating possible answer particles in a document. The rule based question classification module classifies about 89% of the question correctly. The document retrieval component shows greater coverage of relevant document retrieval (97%) while the sentence based retrieval has the least (93%) which contributes to the better recall of our system. The gazetteer-based answer selection using a paragraph answer selection technique answers 72% of the questions correctly which can be considered as promising. The file based answer selection technique exhibits better recall (91%) which indicates that most relevant documents which are thought to have the correct answer are returned.

KEYWORDS: Amharic Question Answering, Answer Selection Techniques, Sentence/paragraph Re-ranking, Question Answering Evaluation

1. INTRODUCTION

The traditional information retrieval techniques were considered insufficient in retrieving precise information to the user. While information retrieval is effective by itself, users these days demand a better tool. First, they want to reduce the time and effort involved in formulating effective queries for search engines (users are required to formulate queries that should maximize document matching and the search engine processes the query as submitted), and secondly they want their results to be real answers - not the list of relevant links. Automatic question answering has become an interesting research area and has resulted in substantial improvement in its performance [1]. The aim of question answering (QA) is to retrieve exact information from a large collection of documents such as those on the Web. The main initiative behind QA systems development is that users in general prefer to have a single answer or a couple of answers for their questions rather than having a number of documents to be read as it happens with the output of search engines [2]. Having a huge number of documents such as the

World Wide Web or a local collection, a QA system should be able to retrieve answers to questions formulated in natural language. QA systems have already been developed in different languages such as Chinese [3, 4, 5], English [6, 7] and so on.

This research is about Amharic Question Answering (AQA) System (ተጠየቅ), which is the first of its kind. This is because the number of Amharic documents on the Web is increasing gradually as many newspaper agencies started to provide their services electronically. Our QA system has been given the name ***ተጠየቅ*** *(Be questioned),* a historical verbalism in Ethiopia where two people appear before a judge used to ask a question for the defendant which are of kind ironic**.** Amharic is written with a version of the Ge'ez script known as ፊደል (Fidel). The Amharic language has its specific way of grammatical construction, character (fidel) representation, and statement formation [8, 9, 10] where a question answering system depends on all for question processing and answer selection techniques.

The question construction and answering techniques in Amharic language are different from English and other languages. In English, questions will be developed, for example, using "***wh***" words such as "Who is the Prime Minister of Ethiopia?". But this same question will have a different structure in Amharic such as a difference in character and word formation as well as grammatical arrangement and type of question particles (terms used to ask questions) used. For example, the above question will be translated to (¾›=ƒÄåÁ ÖpLÃ T>'>eƒ` **T"** ÃvLሉ; - *Yeethiopia Teqlay minister man yibalalu*). This question needs a special consideration to exactly return the correct answer, which is very different from English and other languages question answering techniques. In this paper, we investigated the problems and limitations of an Amharic search engine, the effect of developing a QA system, analyze the strengths and weaknesses of QA with respect to search engines and developed an Amharic question answering system.

2. Question Answering Systems

Information Retrieval (IR) has been researched extensively mainly to help users in getting relevant documents from large collection of free-text documents. The way IR tackles the problem of document retrieval is based on the closeness of the document and the query submitted to the IR system. IR will not try to present answers to users explicitly. This was the critics of IR so that the need of Information Extraction (IE) came about. The IE technique involves NLP tools for precisely indicating a correct text. There should be deep analysis of queries (i.e., user questions) to understand the user's intention as well as deep analysis of the document to extract correct answers (sentences or passages). In the case of IR, a simple technique is sufficient to extract content-rich words from the query and applying stemming to make more uniformity of document retrieval that will be applied during indexing too [15].

A typical pipeline question answering architecture has four components; question analysis, document retrieval, passage (sentence) retrieval and answer extraction [13,14]. In this architecture, the Question Analyzer is responsible to analyze the question that is determining the proper expected answer type and formulating proper queries for the Document Retriever. The Document Retriever will retrieve the top n related documents that will be subjected to the Passage Retriever later. The Passage Retriever will extract passages that pinpoint possible answer strings. The final component, Answer Extractor, will extract the correct answer from the ranked extracted passages.

3. Related Work

There are many related works in the literature. Here, we will present some of them that are relevant to our work. The work in [16] investigated a number of techniques for open-domain question answering. Investigated techniques include: manually and automatically constructed question analyzers, document retrieval specifically for question answering, semantic type answer extraction, and answer extraction via automatically acquired surface matching text patterns. Besides Factoid QA techniques, the paper briefly investigated approaches in definitional questions. The novel techniques in the paper are combined to create two end-to-end question answering systems which allow answers to be found quickly. The first is AnswerFinder[1] which answers factoid questions such as *"When was Mozart born?"*, whilst the second, Varro, builds definitions for terms such as "what is *aspirin?"*, "what does *Aaron Copland* mean?", and "define *golden parachute"*. Both systems allow users to find answers to their questions using web documents retrieved by Google. Together, these two systems demonstrated that the techniques developed can be successfully used to provide quick and effective open-domain question answering.

Marsha Chinese question answering system [18] focuses on evaluating techniques employed for the Chinese language. Marsha consists of three components, query processing, information retrieval (Hanquery search engine), and answer extraction just like many other QA systems. The query processing component analyzes the Chinese questions submitted and formulates a formal query that will be submitted to the IR component. The search engine component retrieves related candidate documents from the database. The answer extraction module extracts correct answers or candidate answers that are presented to the user.

The paper in [19] focuses on developing Hindi QA system which has different language constructs, query structure, common words, and so on as compared to English. The main aim of the paper was to help elementary and high school students in getting correct answers for their subject related questions whereby facilitating e-learning in the Hindi language. For query construction, the researchers used self constructed lexical database of synonyms since there was no Hindi WordNet available. A case-based rule has been developed to classify questions. After the user question is changed to a proper query (by applying stop-word deletion, domain knowledge entity inclusion, and so on), the query will be submitted to the retrieval engine. Finally answer selection is done by extensive analysis of passages and the correct answer will be presented to the user.

4. THE AMHARIC LANGUAGE

Amharic is a Semitic language spoken in many parts of Ethiopia. It is the official working language of the Federal Democratic Republic of Ethiopia and thus has official status nationwide. It is also the official or working language of several of the states/regions within the federal system. Outside Ethiopia, Amharic is the language of millions of emigrants (notably in Egypt, Israel and Sweden), and is spoken in Eritrea [11]. It is written using a writing system called *fidel* or *abugida*, adapted from the one used for the now-extinct Ge'ez language. Ethiopic characters (fidels) have more than 380 Unicode representations (U+1200-U+137F) [12].

In every language, questions are constructed with the help of question particles (interrogative words) and question marks (?) which is placed at the end of the question. Amharic also has its own question particles some of which are shown in Table 1 [20].

[1] *AnswerFinder* is a project that is developed by the Centre for Language Technology at Macquarie University

Question word	Transliteration	Description
ማን	*man*	Who …
ለማን	*leman*	to whom
ማነ ው	*manew*	Who is …
የት	*yet*	Where …
ስንት	*sint*	How many
ለምን	*lemin*	Why …

Table 1: Some Amharic Question Particles

There are different challenges in Amharic question answering. One of the main problems is that question particles by themselves can't help in determining the question type. Extra analysis is required to determine the question type so as to know the expected answer types. Secondly, some proper names belong to more than one word categories, such as verb and noun so that determining whether that word is the expected proper name or not is very difficult. This problem is aggravated as there is no proper name capitalization in Amharic. Lastly, statement demarcation is a problem as there is no standardized writing by different writers.

5. DESIGN OF AQA

Every question answering system will have basic components of Question Analysis, Document retrieval and Answer Extraction [13, 14]. Our QA system has mainly five components, document pre-processing, question processing, document retrieval, sentence/paragraph re-ranking, and answer selection modules as shown in figure 1.

In the **document preprocessing** module, documents will be normalized to show similar standards for document retrieval and answer selection processing. Amharic is too specific in having different character representation with the same reading and writing style. For example, the character ሀ (*ha*) has five equivalent characters with the same reading style and are used interchangeably. These are ሃ, ሐ, ሓ, ኀ, and ኃ. All occurrences of these characters should be replaced with ሀ (*ha*). The research shows that document processing improves the performance of our system nearly by 12 percent.

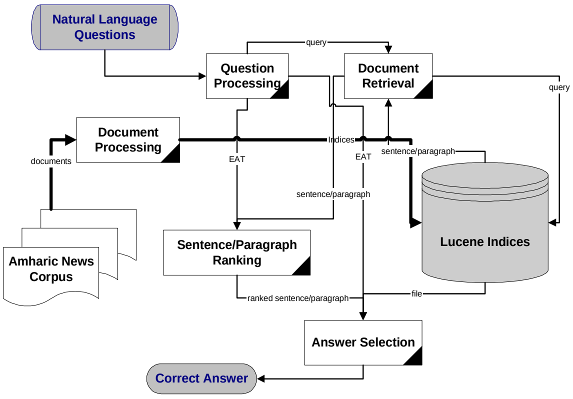


Figure 1: Design of AQA

Besides character normalization, we also did number normalization. Numbers in Amharic can be represented in Arabic numerals, Ethiopic numerals, or alphabetically using words. Number normalization helps to detect all possible numeric answer particles (expected answers) in the document which otherwise are left unmatched. Once character and number normalizations are done, stopwords are removed and proper stemming is performed. A total of 83 terms and all question particles are considered stop words. To delimit documents into sentence and paragraph, we have used different techniques. First sentences are detected with the Amharic full stop (።) if the document uses this punctuation mark. If the document uses group of Amharic word spaces (፡) or group of colons, we replace it with Amharic full stop. If the document is prepared with none of the punctuation marks mentioned, sentence finishing words such as ነው-*newu*, ታውቋል-*tawuquwal*, ተባለ-*tebale*, ገልጿል-*geltsuwal*, etc. are used. Similarly paragraphs are detected by the normal paragraph separator (new line followed by a blank line) or by an average number of sentences that can make up a paragraph. Once the document is normalized, sentences and paragraphs are delimited; then, the final task is to create sentence, paragraph and file indexes.

The **question processing** module accepts the user's question and performs tasks such as question type determination, question focus (important terms about the question) identification, and expected answer type determination. The question type will be determined based on the question particles and the question focuses. Since most of the question particles in Amharic are multipurpose, the question focus plays the greater role in determining the question type. We have developed a question typology that is be used to determine the expected answer type. The question processing module also generates the proper IR query that is submitted to the document retrieval component.

The question processing sub-component is shown in figure 2.

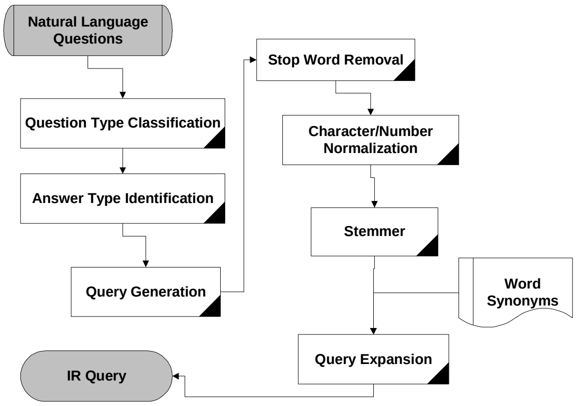

Natural Language Questions
Question Type Classification
Answer Type Identification
Query Generation
IR Query
Stop Word Removal
Character/Number Normalization
Stemmer
Word Synonyms
Query Expansion

Figure 2: question processing subcomponent

The **document retrieval** component retrieves relevant documents to be used by the sentence/paragraph re-ranking module. For the document retrieval module, different techniques were used from the Lucene contribution package. Lucene is a high performance, scalable IR library. It helps to add indexing and searching capabilities to applications. It is a mature, free, open-source project implemented in Java and a member of the popular Apache Jakarta family of projects, licensed under the liberal Apache Software License. It has facilities for text indexing and searching that can be integrated into applications [17].

SpanNearQuery and RegexQuery of Lucene have been used to maximize retrieval of documents with possible answer particles present. The RegexQuery was specially used to retrieve documents specifically for date and numeric related documents. The SpanNearQuery helps to filter out relevant documents by considering how far the query terms are present in the document. In addition to these techniques, we have also regulated the number of query terms present in the document to be considered relevant to maximize relevant document retrieval. If the number of query terms is less than 3, the document is required to contain all of the query terms to be considered relevant. If the query terms vary from 4 to 6, at least 3/4th of the query terms should be present in the document and if the number of query terms is greater than 7, the document is required to consist at least half of the query terms to be considered relevant.

The rules that we have designed indicated that as more number of query terms is present in a document, it is a better answer bearing document. Hence, the document retrieval component retrieves the sentence/paragraph and presents these documents to the sentence/paragraph re-ranking module and it also retrieves the total files and present them to the answer selection module.

The **sentence/paragraph re-ranking** module first detects a possible answer particle in the retuned document. We have used two techniques to pinpoint a candidate answer in a document. The first one is Named Entity based (using a gazetteer for place names and person names, and regular expressions for numeric and date question types). The second one is pattern based answer pinpointing where a generic pattern is determined especially for person names. Once answer particles are identified, the best answer is determined based on query term-answer particle distance computation, if multiple answers are detected.

The candidate answer in a document which seems very near to the query terms will be considered possible best answer. Once all possible candidate answers are identified from all documents, then another computation is done based on the number of query terms present in the document. When re-ranking, the document which shows more number of the query terms will be ranked atop, while the one with least number of query terms receives the least rank.

The algorithm we have developed for the re-ranking sentences based on their weight is given in figure 3.

```
For each query term and expected answer type (EAT)
accepted from the query generation subcomponent of
question processing
    For each sentence
        i.    Find answer particle(s) based on EAT and
              count the number of answer particles (count)
        ii.   If count > 0 go to iii. Else go to vii.
        iii.  For each answer particle present
              For each query term
                  a. Count the number of terms between the
                     answer particle and itself
                  b. Add the value of each query term
                     distance (distancecount)
        iv.   If still there are more answer particles, go to iii.
        v.    If count=1//only one expected answer
              Return the answer particle
        vi.   Else if count >1, return the answer particle
              with the least distance (distancecount )
        vii.  Else if count = 0, discard the sentence //no
              answer particle presents
        viii. If the selected answer particle
              Count the number query terms in a sentence
              (countqueryterm)
              Assign countqueryterm as a weight to the
              answer particle
    End for
For all sentences with the identified answer particle
    Sort the sentences based on their weight//based on
    countqueryterm
End for
```

Figure 3: Algorithm for determining sentence weight

The **answer selection** module selects the best top 5 answers from the previously ranked documents. Besides the already determined rank, the answer selection module also checks for possible repetition of answer particles from the candidate answer pool. If a given answer particle is repeated, the rank of the two will be summed to give a newer rank. Answer particles with the maximum rank value will be selected as an exact answer. The answer selection module considers two answers as equivalent if one is a short form of the other.

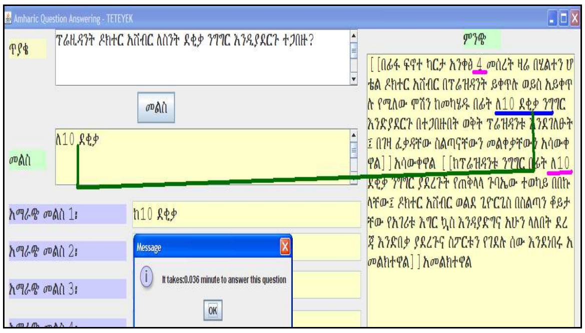
For example ጠቅላይ ሚኒስትር አቶ መለስ ዜናዊ (Prime Minister Ato Meles Zenawi), አቶ መለስ ዜናዊ (Ato Meles Zenawi), ጠቅላይ ሚኒስትር መለስ (Prime Minister Meles), and አቶ መለስ (Ato Meles) are all considered equivalent. The algorithm for selecting best answer is shown in figure 4.

```
While there are extra sentences containing a candidate
answer
    For a candidate answer in a sentence
        Count the number of query terms in the sentence
        //count
    For a candidate answer in a sentence
        Compare the candidate answer with candidate
        answers in other sentences
            If it matches with other candidate answers (or
            contains function)
                Add count of the two answer particles
                Concatenate the two sentences for summary
                Remove the sentence from the list
    End for
End while
For each weighted sentence //selection
    For each sentence (concatenated sentences) compare
    its weight with succeeding sentence(s)
        If the weight of the current sentence(s) is less than
        the succeeding sentence
            Swap their position
        Return the highest count sentence
    End for
End for
```

Figure 4: Selecting a sentence based on the higher occurrence of a candidate

6. EXPERIMENT

Java Programming language, the Lucene API, and a number of other third-party Java libraries such as *Fileutils* are used in developing our prototype. Figure 5 shows the user interface of our prototype.



Amharic Question Answering - TETEYEK

ጥያቄ: ፕሬዚዳንት ዶክተር አሸብር ለስንት ደቂቃ ንግግር እንዲያደርጉ ተጋበዙ?

መልስ

መልስ: ለ10 ደቂቃ

አማራጭ መልስ 1፡ ከ10 ደቂቃ

አማራጭ መልስ 2፡

አማራጭ መልስ 3፡

Message

It takes:0.036 minute to answer this question

OK

ምንጭ

[[በሪፋ ፍኖተ ካርታ አንቀፅ 4 መሰረት ዛሬ በሂልተን ሆቴል ዶክተር አሸብር በፕሬዝዳንት ይቀጥሉ ወይስ አይቀጥሉ የሚለው ምሽን ከመካሄዱ በፊት ለ10 ደቂቃ ንግግር እንድያደርጉ በተጋበዙበት ወቅት ፕሬዝዳንቱ እንደገለፁት፤ በገዛ ፈቃዳቸው ስልጣናቸውን መልቀቃቸውን አሳውቀዋል]] አሳውቀዋል [[ከፕሬዝዳንቱ ንግግር በፊት ለ10 ደቂቃ ንግግር ያደረጉት የጠቅላላ ጉባኤው ተወካይ በበኩላቸው፤ ዶክተር አሸብር ወልደ ጊዮርጊስ በስልጣን ቆይታቸው የአገሪቱ እግር ኳስ እንዲያድግና አሁን ላለበት ደረጃ እንድበቃ ያደረጉና ስፖርቱን የገደሉ ሰው እንደነበሩ አመልክተዋል]] አመልክተዋል

Figure 5: User Interface for AQA

Nearly 12000 question sets have been collected from the Web, Ethiopian Television games and from questionnaire respondents. A total of 15600 Amharic news articles (42 MB) corpus has been collected and normalized. Out of 12000 total questions, nearly 500 factoid questions are selected for the experiment. Hence, the experiment is conducted on the designed sample question and answer sets. The evaluation criteria we have used were correct answer accuracy. Hence the accuracy of our system is evaluated for recall, precision, percentage, and mean reciprocal rank (MRR). The evaluation methods for these criteria are as follows:

Recall: is calculated as the total number of correct answers over the total of correct and missed answers while present in the corpus.

$$\mathrm{Re}call = \frac{ca}{ca + ma} X100\%$$

where ca is correct answers and ma is missed answers

Precision: is calculated as the percentage of correct answers over the total of correct answers, wrong answers, and No answers.

$$\mathrm{Pr}\,ecision = \frac{ca}{ca + wa + na} X100\%$$

where ca is correct answers, wa is wrong answers and na is no answers.

Percentage: is calculated as the total number of correct answers over all responses, wrong answers over all responses, and No answers over all responses.

$$Percentages = \frac{ca}{ta}, \frac{wa}{ta}, \frac{na}{ta} X100\%$$

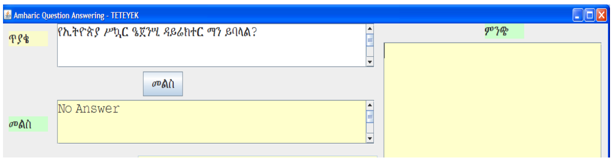
where ta is total answers.

Mean Reciprocal rank (MRR): is also computed to evaluated average rank of answers; where rank is from top one to top five.

$$\mathrm{MRR} = \frac{\sum_{i}^{n} \frac{1}{Ri}}{n}$$

where Ri is the rank of a given answer which ranges from 1 to 5, and n is the total number of answers (correct + wrong + No answer).

The performance of our system has been evaluated before and after document normalization. The evaluation showed the significant effect of document normalization for performance. Consider figures 6 and 7 to see the impact of document normalization.



Amharic Question Answering - TETEYEK

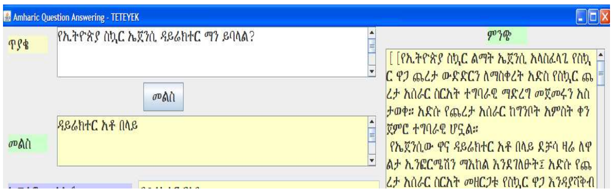
ጥያቄ: የኢትዮጵያ ሥኳር ዔጀንሢ ዳይሬክተር ማን ይባላል?

መልስ

መልስ: No Answer

ምንጭ

Figure 6: Before charcter normalization



Amharic Question Answering - TETEYEK

ጥያቄ: የኢትዮጵያ ስኳር ኤጀንሲ ዳይሬክተር ማን ይባላል?

መልስ

መልስ: ዳይሬክተር አቶ በላይ

ምንጭ

[[የኢትዮጵያ ስኳር ልማት ኤጀንሲ አላስፈላጊ የስኳር ዋጋ ጨረታ ውድድርን ለማስቀረት አድስ የስኳር ጨረታ አሰራር ስርአት ተግባራዊ ማድረግ መጀመሩን አስታወቀ። አድሱ የጨረታ አሰራር ከግንቦት አምስት ቀን ጀምሮ ተግባራዊ ሆኗል።

የኤጀንሲው ዋና ዳይሬክተር አቶ በላይ ደቻሳ ዛሬ ለዋልታ ኢንፎርሜሽን ማእከል እንደገለፁት፤ አድሱ የጨረታ አሰራር ስርአት መዘርጋቱ የስኳር ዋጋ እንዲያሻቅብ

Figure 7: after character normalization

Figure 6 shows that the question “የኢትዮጵያ ሥኳር ዔጀንሢ ዳይሬክተር ማን ይባላል?” (Who is the director of Ethiopian Sugar Industry Agency?), has no answer as the characters ሥ, ዔ, and ሢ are not normalized to reflect same character representations as of the document in the corpus. We can clearly see the impact of the character normalization in figure 7 where the same question returns the correct answer.

The effect of document normalization is shown in table 2.

Document	Before normalization		After normalization	
	Precision	Recall	Precision	Recall
Sentence	53.3%	60.3%	66.6%	82.4%
Paragraph	55.4%	63.1%	63.7%	80.6%
File	51.4%	63.9%	55.3%	75.6%

Table 2: Effects of Document normalization

Table 2 shows that document normalization has a performance gain of 7% for precision and 12% for recall.

The Question Answering system correctly classified 89% of the questions using the rule based classification, while 62% are correctly classified by the IR based question classification technique where question sets and answer sets are indexed so that an unseen question will be matched with the help of document similarity computations later.

Index type	Correct answer particles present	Wrong answer particles present
Sentence	465 (93 %)	35 (7%)
Paragraph	477 (95.4 %)	23 (4.6 %)
File	486 (97.2 %)	14 (2.8 %)

Table 3: document retrieval performance

Our document retrieval component also shows an excellent performance as shown in table 3.

Table 4 shows our Named-entity-based answer selection performance for person and numeric question types. The pattern based answer selection outperforms the named entity based answer selection technique as the named entity based answer selection technique fails to address all possible answer particles

Document	Number of correct answers	Number of wrong answers	Number of No Answers	Missed Answers	Precision	Recall	MRR
Sentence	60 (56.6 %)	30(28.3 %)	16 (15.1%)	11	0.566	0.845	0.493
Paragraph	72 (67.9%)	20 (18.9%)	14 (13.2%)	8	0.679	0.900	0.575
File	60 (56.6%)	34 (32.1 %)	12 (13.3%)	6	0.566	0.909	0.438

Table 4 Gazetteer based correct answer performance

7. CONCLUSION AND FUTURE WORK

This paper attempted to identify the basic language specific issues in question answering for Amharic. The first task we have tackled is normalizing the document so that a standard document is indexed and matching relevant documents during searching are maximized. We have also identified proper question particles as well as question focuses that will help in classifying the question. Gazetteer based and pattern based answer selection algorithms have been developed to maximize correct answer selection. Our algorithm first identifies all possible answer particles in a document. Once the answer particles are identified, the distance of every question particle toward the question terms is calculated. The one with the minimum distance from the query terms is considered the best candidate answer of that document. Once candidate answers are selected from every document, candidate answers which have been repeated more than once (i.e., appeared in more than one document) are given higher ranks.

Candidate answers with maximum number of query terms matched in a document are given higher priority in case a similar rank is given for two or more candidate answers. The evaluation of our system, being the first

Amharic QA system, shows very well performance. The rule based question classification module classifies about 89% of the questions correctly. The document retrieval component shows greater coverage of relevant document retrieval (97%) while the sentence based retrieval has the least (93%) which contributes to the better recall of our system. The gazetteer based answer selection using a paragraph answer selection technique answers 72% of the questions correctly which can be considered as promising. The file based answer selection technique exhibits better recall (91%) which indicates that most relevant documents which are thought to have the correct answer are returned. The pattern based answer selection technique has better accuracy for person names using paragraph based answer selection technique while the sentence based answer selection technique has outperformed the performance in numeric and date question types. In general, our algorithms and tools have shown good performance compared with highly resourced language QA systems such as English.

Question answering is a very complex task, which consumes time and needs a number of different NLP tools. Hence, there are a number of rooms for improvement and modification for Amharic question answering. Some of the research works that we have planned to undertake in the future include:

- Developing automatic named entity recognizer
- Incorporating a parser and part of speech tagger
- Developing Amharic WordNet
- Enhancing the Amharic stemmer
- Incorporating Machine learning and statistical Question classifications

REFERENCES

[1] Hu, H. Jiang, P. Ren, F. Kuroiwa, S. 2005. Web-based Question Answering System for Restricted Domain Based of Integrating Method Using Semantic Information Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE '05. Proceedings of 2005 IEEE International Conference.

[2] Anne-Laure Ligozat, Brigitte Grau, Anne Vilnat, Isabelle Robba, Arnaud Grappy, 2007. Towards an automatic validation of answers in Question Answering, LIMSI-CNRS 91403 Orsay CEDEX France.

[3] Dongfeng Cai Yanju Dong Dexin Lv Guiping Zhang Xuelei Miao, 2004. A web based Chinese Question Answering System, Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE '05. Proceedings of 2005 IEEE International Conference.

[4] Shouning Qu, Bing Zhang, Xinsheng Yu, Qin Wang, 2008. The Development and Application of Chinese Intelligent Question Answering System Based on J2EE Technology, Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop.

[5] Zheng-Tao Yu Yan-Xia Qiu Jin-Hui Deng Lu Han Cun-Li Mao Xiang-Yan Meng, 2007, Research on Chinese FAQ questions Answering System in Restricted Domain, Machine Learning and Cybernetics, 2007 International Conference.

[6] Jignashu Parikh, M. Narasimha Murty, 2002. Adapting Question Answering Techniques to the Web, Proceedings of the Language Engineering Conference (LEC'02).

[7] Sameer S. Pradhan, Valerie Krugler, Wayne Ward, Dan Jurafsky and James H. Martin, Using Semantic Representations in Question Answering, Center for Spoken Language Research University of Colorado Boulder, CO 80309-0594, USA.

[8] http://en.wikipedia.org/ wiki/Amharic_language, last accessed on October 1, 2008

[9] ጌታሁን አማረ፣ 1989 የአማርኛ ሰዋሰው በቀላል አቀራረብ - *Getahun Amare, 1989 Ye-Amarigna sewasew beqelal aqerareb.*

[10] ባየ ይማም፤1987 የአማርኛ ሰዋሰው፤ ት.መ.ሚሚ.ድ. – *Baye Yimam, Ye-Amarigna sewasew, T.M.M.M.D.*

[11] http://www.lonweb.org/link-amharic.htm, last accessed on March 30, 2009.

[12] http://jrgraphix.net/research/unicode _blocks.php?block=31, last accessed on March 31, 2009.

[13] Matthew W. Bilotti, Boris Katz, and Jimmy Lin, 2004, What Works Better for Question Answering: Stemming or Morphological Query Expansion?, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

[14] Cheng-Wei Lee, Cheng-Wei Shih, Min-Yuh Day, Tzong-Han Tsai, Tian-Jian Jiang, Chia-Wei Wu, Cheng-Lung Sung, Yu-Ren Chen, Shih-Hung Wu, Wen-Lian Hsu, 2005, ASQA: Academia Sinica Question Answering System for NTCIR-5 CLQA, Proceedings of NTCIR-5 Workshop Meeting, December 6-9, 2005, Tokyo, Japan.

[15] Tomek Strzalkowski and Sanda Harabagiu, 2008, Advances in Open Domain Question Answering, Published by Springer, ISBN 978-1-4020-4746-6, The Netherlands.

[16] Mark A. Greenwood, 2005, AnswerFinder: Question Answering from your Desktop, Department of Computer Science University of Sheffield Regent Court, Portobello Road Sheffield S1 4DP UK

[17] Otis Gospodnetic, Erik Hatcher, 2005, Lucene in Action, Manning Publications Co., 209 Bruce Park Avenue, Greenwich, CT 06830, ISBN 1-932394-28-1.

[18] Xiaoyan Li and W. Bruce Croft, 2002, Evaluating Question-Answering Techniques in Chinese, In NIST Special Publication: The 10th Text Retrieval Conference.

[19] Praveen Kumar, Shrikant Kashyap, Ankush Mittal, Sumit Gupta, 2005, A Hindi Question Answering system for E-learning documents, Proceedings of the 2005 3rd International Conference on Intelligent Sensing and Information Processing.

[20] Seid Muhie Yimam and Mulugeta Libsie, “TETEYEQ: Amharic Question Answering For Factoid Questions”, In Proceedings of Information Retrieval and Information Extraction for Less Resourced Languages Conference (IE-IR-LRL), Donostia, Spain, September 7th, 2009.