TETEYEQ: Amharic Question Answering For Factoid Questions

"TETEYEQ: sistema de respuesta a preguntas factoides en lengua amárica"

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

Mulugeta Libsie
Addis Ababa University, Ethiopia
mlibsie@aau.edu,
mlibsie@aau.edu

ABSTRACT: The number of Amharic documents on the Web is increasing as many newspaper publishers started 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 the Question Answering systems emerge as best solution to get the required information to the user with the help of information extraction techniques. 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 on un-normalized ones. A distinct technique was used 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

Amharic documents on the web increases gradually as many newspaper agencies provide their service electronically. 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 a 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 the Web. The main initiative behind QA system 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 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. Our QA system has been given a 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 question answering system depends on both 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?" and so on. But this same question will have 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 as (**የኢትዮጵያ ጠቅላይ ሚኒስትር ማን ይባላሉ? -** *ye-ethiopia Teqlay minister man yibalal*). This question needs a special consideration to exactly return the correct answer, which is very different from English and other languages question answering techniques. There is no QA system developed for Amharic so far. In this study, we will investigate the problem and limitations of an Amharic search engine, the effect of developing QA system, analyze the strengths and weaknesses of QA with search engine and try to develop an Amharic question answering system.

2. 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, including Amhara and the multi-ethnic Southern Nations, Nationalities and Peoples region. Outside Ethiopia, Amharic is the language of millions of emigrants (notably in Egypt, 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. Table 1 shows some of the Amharic question particles.

Question word	Transliteration	Description
ማን	*man*	Who related questions
ለማን	*leman*	to whom …
ማነው	*manew*	Who is …..
የት	*yet*	Where …
ስንት	*Sint*	How many ….
ለምን	*Lemin*	Why
…	…	…

Table 1: Amharic Question Particles

3. 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.

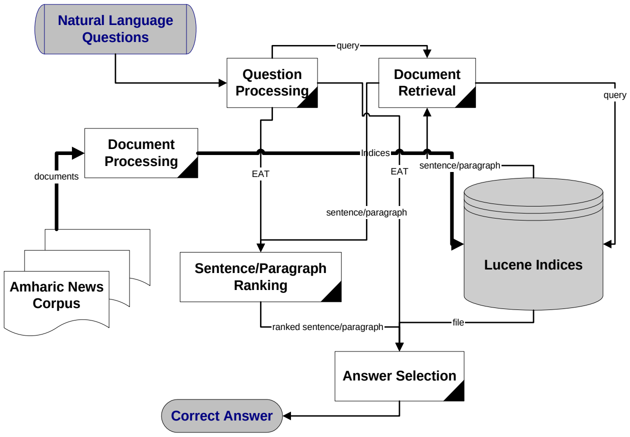


Figure 1: Architecture of the system

4. IMPLEMENTATION

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 nearly five equivalent characters possess the same reading style, and people use it interchangeably with it, that are ሃ, ሐ, ሓ, ኀ, and ኃ and all occurrences these characters should be replaced with ሀ (*ha*). The research shows that document processing improves the performance of our system nearly by 12 percent (see Section 5). Besides character normalization, we also did number normalization. Numbers in Amharic represented in Arabic, Ethiopic, and alphabetic ways. The number normalization help as to detect all possible numeric answer particles (expected answers) in the document which otherwise left un-matched. To delimit documents in to 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, we use a frequently sentence finishing words such as (**ነው**-*newu*, **ታውቋል**-*tawuquwal*, **ተባለ**-*tebale*, **ገልጿል**-*geltsuwal*, etc.). Similarly paragraphs are detected by the normal paragraph separator (new line followed by a blank line) or an average number of sentences that can make up a paragraph. Once the document is normalized, sentences and paragraphs are delimited; then, our final task is create sentence, paragraph and file indexes using Lucene.

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 will be used to determine the expected answer type. The question processing module also generate the proper IR query that will be submitted to the document retrieval component of AQA.

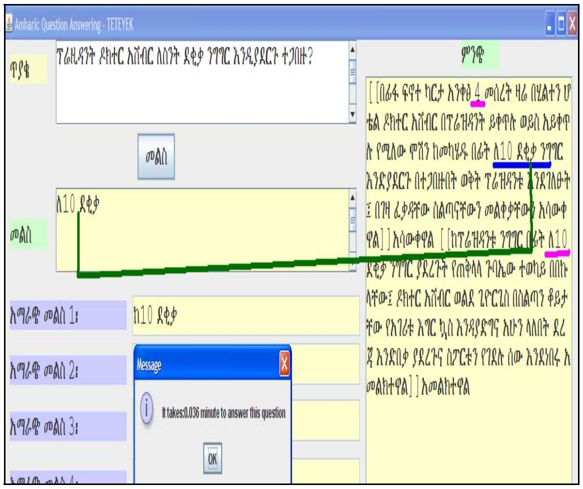
The **document retrieval** component retrieves relevant documents so that the sentence/paragraph re-ranking module will process on it. For document retrieval module, different techniques were used. SpanNearQuery and RegexQuery, the Lucene contribution packages, have been used to maximize retrieval of 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 presence 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 term varies from 4 to 6, at least 3/4 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 the query terms to be considered relevant. The rules designed indicate that as more number of query terms is present in a document, it is considered as 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 retrieve the total file and present the document directly 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 receive the least ranking weight.

The **answer selection** module selects the best top 5 answers from the previously ranked documents. Beside 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 of the other is the short form of it. 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.

5. 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 2 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

[[በሬፋ ፍኖት ካርታ እንቅፅ 4 መሰረት ዛሬ በሂልተን ሆቴል ዶክተር አሸብር በፕሬዝዳንት ይቀጥሉ ወይስ አይቀጥሉ የሚለው ጥያቄን ከመከሄዱ 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 እንዲያድግ አሁን ላለበት ደረጃ እንደበቃ ያደረጉና ስፖርቱን የገደሉ ሰው እንደነበሩ አመልክተዋል]] አመልክተዋል

Figure 2: 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 criterion 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 formula for these criteria is as follows:

Recall: It is calculated as the total number of correct answers over the total of correct and missed answers.

Recall= $\frac{correct}{correct + missed\ answers} X100\%$

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

Precision =

$$\frac{correct}{correct\ + wrong\ + No\ \ answers} X100\%$$

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

Percentage =

$$\frac{correct}{total\,answers}, \frac{wrong}{total\,answers}, \frac{No\,answers}{total\,answers}$$

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

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 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 processing module correctly classifies 89% of the questions using the rule based classification, while 62% are correctly classified by the IR based question classification techniques 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 techniques as the named entity based answer selection technique fails to address all possible answer particles.

6. CONCLUSION

This research work attempted to identify the basic language specific issues in question answering. The first task we have tackled is normalizing the document so that a standard document will be indexed and matching relevant documents during searching will be maximized. We have also identified proper question particles as well as question focuses that will help in classifying the

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	56.6%	84.5%	49.3%
Paragraph	72 (67.9%)	20 (18.9%)	14 (13.2%)	8	67.9%	90.0%	57.5%
file	60 (56.6%)	34 (32.1 %)	12 (13.3%)	6	56.6%	90.9%	43.8%

Table 4: Gazetteer based correct answer performance

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 particles toward the question terms will be calculated. The one with the minimum distance from the query terms will be 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) will be given higher rank. Candidate answers with maximum number of query terms matched in a document will be 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 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. 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.

7. CONTRIBUTION OF THE WORK

The main contributions of this thesis work are summarized as follows:

✓ The study has adopted the efforts made towards English QA systems techniques to Amharic.

✓ The study has paved the way to identify language dependent components specific to Amharic question answering.

- ✓ The study identified key components of Amharic QA systems which can be considered a framework for factoid questions.
- ✓ The study showed the strategy, algorithms, and techniques in developing Amharic QA system.
- ✓ The study showed how questions in Amharic can be classified hierarchically (coarse and fine grained based), what are the specific question focuses for different questions, and the function of question particles to determine question type and expected answer types.
- ✓ This study also showed how information extraction can be accomplished in Amharic based on the standard off-the-shelf information retrieval techniques available.
- ✓ The study identified basic challenges in developing Amharic QA systems and the possible strategies to solve those challenges.

8. FUTURE WORK

Question answering is a very complex task, which consumes more time, and needs a number of different NLP tools. Hence, there are a number of rooms for improvement and modification for Amharic question answering. Below are some of the recommendations we propose for future work.

- Developing automatic named entity recognizer: The gazetteer we have used has limitations such as usage of a single named entity for multiple entities (such as person and place). Developing an automatic named entity recognizer will help the QA system to automatically detect the expected answer.
- Incorporating a parser and part of speech tagger: The NER will detect named entities in a document. A sentence parser will further help the QA system to know the structure of the question and the expected answer sentence. Besides, there is no POS tagger available publicly to integrate with our QA system. Integrating POS tagger will help the answer processing component of the QA system so that wrong answer particles, such as considering a verb as proper noun, will be eliminated.
- Developing Amharic WordNet: Word synonym, hyponym, antonym, metonym, meronym and so on help to match wider number of relevant documents. By using Amharic synonyms and the like, we believe that Amharic WordNet is very beneficial.
- Enhancing the Amharic stemmer: The stemmer that we have used brought some drawbacks both for document retrieval and answer processing algorithms. It will be better to develop a state-of-the-art stemmer which we believe will bring a significant change to the performance of QA systems.
- Incorporating Machine learning and statistical Question classifications: the rule based and IR based question classifications have some limitations. The rule based approach does not include all possible patterns of questions and the IR approach also does not help as the number of questions and question types indexed are very small. The machine learning and statistical approaches show better performance for other QA systems such as

English [60] and we hope it will also help for Amharic QA systems as well.

- Integrating with other search engines: for this research work, documents have been collected manually with the help of third party tools such as **DownThemAll** of Firefox and **WinHTTrack website copier**[1]. It will be better to incorporate a crawler component which will interact with the main search engines (Google, Yahoo, etc.) and Amharic Websites for collecting relevant documents.
- Extending to other question types: This research work shows that, even with minimal NLP tools, it would be possible to handle other question types such as list, define, and so on. Extending this work to other question types will be beneficial for wider applications where only a piece of information is not sought.
- Incorporating Amharic spell checker: most of the wrong answers and wrong documents returned are due to spelling errors. Incorporating spell checker will enhance the performance of our system.

Implementing for specific applications: The QA system can be easily implemented to satisfy the needs of some organizations for specific projects. It can be developed for customer service support such as e-commerce and e-governance.

REFERENCE

[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

[1] HTTrack is a free (GPL, libre/free software) and easy-to-use offline browser utility, http://www.httrack.com/

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

[10] **ባየ ይማም፤**1987 **የአማርኛ ሰዋሰው፤ ት.መ.ማ.ማ.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 TechnologyCambridge, 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