BFQA: A Bengali Factoid Question Answering System

Somnath Banerjee, Sudip Kumar Naskar, and Sivaji Bandyopadhyay

Department of Computer Science and Engineering,
Jadavpur University, India
s.banerjee1980@gmail.com
{sudip.naskar,sbandyopadhyay}@cse.jdvu.ac.in

Abstract. Question Answering (QA) research for factoid questions has recently achieved great success. Presently, QA systems developed for European, Middle Eastern and Asian languages are capable of providing answers with reasonable accuracy. However, Bengali being among the most spoken languages in the world, no factoid question answering system is available for Bengali till date. This paper describes the first attempt on building a factoid question answering system for Bengali language. The challenges in developing a question answering system for Bengali have been discussed. Extraction and ranking of relevant sentences have also been proposed. Also extraction strategy of the ranked answers from the relevant sentences are suggested for Bengali question answering system.

Keywords: BFQA, Question Answering (QA), Bengali Factoid QA

1 Introduction

A QA system is an automatic system capable of answering natural language questions in a human-like manner: with a concise, precise answer. Generally questions can be classified into five broad categories ([1,2]): factoid questions, list questions, definition questions, complex questions and speculative questions.

As there exists no Bengali QA system till date and the development of Bengali QA system is at its nascent stage, our aim for the work reported here was to address the factoid questions for the following factors:- (i) a considerable percentage of the questions actually submitted to a search engine belongs to factoid questions. (ii) The percentages of factoid questions are increased each year in TREC due to frequent occurrences in daily usage. (iii) Sophisticated state-of-the-art approaches to open-domain QA use named entity recognition as a core process for detecting candidate answers.

2 Related Work

Designing a QA system for European languages particularly for English is not new in natural language processing. A number of QA systems have been developed since the 1960s. Two such early QA systems were BASEBALL[3] and LUNAR[4].The most notable QA system available to date is IBM Watson [2] which was developed under IBM's DeepQA project. Research in QA received significant boost when a shared task on factoid QA was included in the 8thText REtrieval Conference (TREC).

P. Sojka et al. (Eds.): TSD 2014, LNAI 8655, pp. 217–224, 2014.
This is preprint prepared by Proceedings editor for Springer International Publishing Switzerland.

A number of QA systems were developed for European languages particularly for English ([5,6,7]), Middle Eastern languages ([8,9,10]) and Asian languages, e.g., Japanese ([11,12]) Chinese ([13,14]), etc. The aforesaid systems are capable of providing answers with reasonable accuracy. However, for Bengali, which is a widely spoken language in India and among the most spoken languages in the world, very little work [12,13,14] have been reported so far in QA research like other Indian languages.

3 Challenges

To the best of our knowledge, there exists no QA system till date for Bengali. Developing a QA system for low resource language is very much challenging. Several issues were confronted for developing the system which includes-

- *Presence of many interrogatives*: Unlike English there are many interrogatives present in the Bengali. A study [15] identified a total of twenty six interrogatives and classified them into three categories – Unit Interrogative (UI), Dual Interrogative (DI) and Compound/Composite Interrogative (CI).
- *Interrogative position*: A Bengali interrogative can appear in all potential positions, i.e., three positions (first, in between, last) of a question text [12]. This makes it difficult to propose rule-based question analysis.
- *Resource scarcity*: The language processing tools for Bengali are either under development phase or not developed yet. Even a fully-fledged parser has not been developed yet and no NER system is publicly available for Bengali. Besides, gold-standard corpora for QA research are not developed yet.

4 BFQA Architecture

Our proposed factoid QA system for Bengali language, named BFQA, has a pipeline architecture having three components, namely question analysis, sentence extraction and answer extraction. The question analysis module accepts natural language question in Bengali as input posed by the user. The question analysis step processes in five stages, namely question type (QType) identification, expected answer type (EAT) identification, named entity identification, question topical target (QTT) identification and keyword identification. The valid keywords are 'AND'ed together to form the query. Sentences are extracted from the passages based on the query and are ranked based on the answer score value. Finally, extracted answers are validated using the EAT module. The architecture of the proposed model is depicted in Figure 1

5 Question Analysis

Question analysis plays a crucial role for an automatic QA system. Acquiring the information embedded in a question is a primary task that allows the QA system to decide the appropriate strategy in order to provide the correct answer to the question. [18] stated that when the question analysis module fails, it is hard or almost impossible for a QA

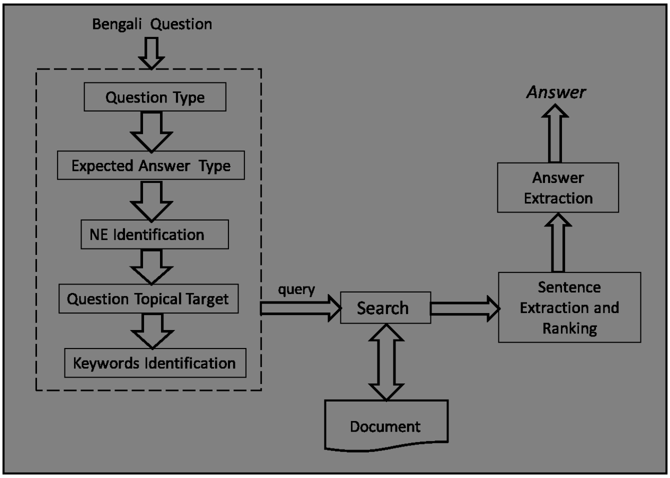

Fig. 1. BFQA Architecture

system to provide correct answer. In this work, question analysis task is divided into five parts: question type identification, inferring the expected answer type, obtaining the question topical target, named entity (NE) extraction and identification of keywords.

Question Type: Question Type (QType) identification or question classification is an important component of every Question Answering System (QAS). Depending on the classification strategy, the task of a question classifier is to assign one or more predefined class labels to a given question written in natural language. The set of predefined categories which are considered as question classes is usually referred to as question taxonomy or answer type taxonomy. The single layer taxonomy with nine coarse-grained classes proposed by [12] is the only standard taxonomy available in Bengali QA research. We followed the approach of [14] which is the best reported work (91.65% accuracy) yet for Bengali QType identification task.

Expected Answer Type: Prager ([16]) defined the Expected Answer Type (EAT) as the class of object or rhetorical type of sentence required by the question. Another way EAT may be defined is as the semantic category associated with the desired answer, chosen from a predefined set of labels. A set of hand-crafted rules written as regular expressions (RE) were used by many sophisticated QA systems ([17],[18],[19]) for finding the EAT. Every RE is associated with an EAT that is to be assigned to a question if it matches its pattern. Though hand-crafted rules are very specific to the particular domain, but those are very useful at the initial stage of the development. In this work, hand-written rules were used to determine EAT of a question.

Named Entity Identification: NEs play a crucial role in question answering. NEs present in the question are believed to be present in the sentence that contains the expected answer. Again, the answer of a factoid question is a NE. Therefore, NE identification in the question is very much necessary. However, unfortunately any openly available NER system is not available for Bengali. Therefore, we implemented the NER system proposed by [20] which reported 65.95% accuracy.

Question Topical Target: Knowing the question type alone is not sufficient for finding answers to questions [21]. Question Topical Target (QTT) (sometimes also referred to as question focus [22], or question topic [16]) corresponds to a noun or a noun phrase that is likely to be present in the answer. Proper identification of QTT benefits the question answering process, since QTT terms or their synonyms are likely to appear in a retrieved sentence that contains the answer. Due to varying position of Bengali interrogatives, it is very challenging to separate the QTT when the interrogative appears within the noun phrase. To follow the same strategy for all cases, Bengali QTT is defined as a comma separated named entity list in this work.

Keywords Identification: The keywords of the posed natural language question are used to form query which is used to extract passages that might contain the expected answer. Shallow parser for Bengali is used to parse the question in the absence of any full-fledged parser. The nouns, proper nouns i.e., NEs, verbs and adjectives are taken as legitimate keywords and other words are considered as stop words. As the NEs are already included in QTT, query is formed by merging QTT with nouns, verbs and adverbs present in the question. The valid keywords are 'AND'ed together to form query.

6 Sentence Extraction and Ranking

The developed corpus for this work has the constraint that all the questions are related to a particular document only. So, we imposed some constraints on the searching technique: the answer to the posed question is searched from a single document and each of the paragraphs of the document is treated as individual documents while searching. The task of this module is to extract the relevant sentences that contain the expected answer from different paragraphs of a single document. The relevant sentences are searched using the query. The relevant sentences are searched taking into account the imposed constraints.

The extracted sentences that contain the expected answer are initially passed to the same NE system that tagged the NEs appearing in the question text. The extracted sentences are then ranked on the basis of answer score. EAT and similarity score are used to calculate the answer score. The answer score is calculated based on – (i) syntactic similarity, (ii) name proportions, and (iii) paragraph relevancy.

Syntactic Similarity: if Q_t is the natural language (NL) question and is composed of n words, then the question text Q_t can be expressed as:

$Q_t = Q_1 Q_2 Q_3 Q_4 ... Q_{n-2} Q_{n-1} Q_n$

Let $V_Q = Q_1, Q_2, Q_4, Q_7, Q_8, ... Q_n$, and $V_{Stop} = Q_3, Q_5, Q_6, Q_9, ...$;

Then $Q_t = V_Q \cup V_{Stop}$ and $V_Q \cap V_{Stop} = \emptyset$; where V_Q and V_{Stop} are the two word vectors, namely content and stop words respectively.

V_S is the sentence vector which contains all the sentences in the document of p sentences. i.e. $V_S = V_{S_1}, V_{S_2}, V_{S_3}, V_{S_4}, ... V_{S_{P-1}}, V_{S_P}$; where V_{S_k} represents the k^{th} sentence in the document and contains c words.

$V_{S_k} = w_1, w_2, w_3, w_4, ..., w_{c-1}, w_c$

In similarity measure, we only consider the part-of-speech (POS): verb (VB), noun (NN), adjective (ADJ) and proper noun (NE). Let {VB, NN, ADJ, NE ∈ POS}. Four weights λ_{vb}, λ_{nn}, λ_{adj} and λ_{ne} have been defined corresponding to the verb, noun, adjective and named entity, respectively. We have set $\lambda_{vb} = 0.2$, $\lambda_{np} = 0.3$, $\lambda_{adj} = 0.1$ and $\lambda_{ne} = 0.4$ (so that the four weights add up to 1).

i.e., $\sum_{Pos \in \{vb,np,adj,ne\}} \lambda_{Pos} = 1$

So, the similarity of a NL question Q_t and a sentence S_l is calculated by the following formula: $Similarity_{(Q_t,S_l)} = \sum_{K=1}^{n} frequency_Q . w_K$;

where, $w_K = Q(\lambda_{Pos})$ and $frequency_Q$ is the number of occurrence of question word Q in the sentence S_l.

Name proportions(nprop): Jaccard similarity coefficient is used to measure name proportion. Jaccard similarity coefficient is a similarity measure that compares the similarity between two feature sets. In name proportion measure, it is defined as the size of the intersection of the named entities in the question and a sentence normalized by the size of the union of the named entities in the question and the sentence.

Paragraph relevancy: Relevancy of a paragraph to a question is measured by counting the presence of query words in that paragraph. The synonyms of the query words are also considered during relevancy count. To distinguish between original query word and synonymous words a *relevancy weight* is assigned to each appearance.

$$relevancy\ weight(r_w) = \begin{cases} 1.0 & \text{if original query term appears in the paragraph} \\ 0.9 & \text{if synonym appears in the paragraph} \\ 0.0 & \text{neither query word nor any synonym} \end{cases}$$

Each of the words in the paragraph is considered for paragraph relevancy calculation. Paragraph relevancy of a word is calculated as follows.

$$R_w = frequency \times relevancy\ weight$$
$$= f_w \times r_w$$

Therefore, if a paragraph contains k distinct words, then the *paragraph relevancy* for that paragraph can be measured using the following formula.

$$R_p = f_{w1} \times r_{w1} + f_{w2} \times r_{w2} + f_{w3} \times r_{w3} + \ldots + f_{n1} \times r_{w_n}$$
$$= \sum_{i=1}^{k} f_{w_i} \times r_{w_i}$$

= sum of the paragraph relevancy for each distinct word in the paragraph

Finally, the score for the three metrics are summed up to arrive at the answer score.

answer score = syntactic similarity + name proportion + paragraph relevancy

7 Answer Extraction

Answer Extraction is the final module of the QA pipeline architecture. After the extracted sentences are ranked based on answer score, the answer to the natural language question is determined by the NE which is suggested by EAT in the question analysis module. Here, three cases are possible -

i) Only a single word in the sentence is of suggested NE type by EAT.

ii) Multiple words having the same NE tag suggested by EAT.

iii) No word in the sentence having NE tag suggested by EAT.

In the first case, the answer extraction is trivial and the NE word suggested by EAT is the answer to the question. However, the second case is a bit ambiguous and we need to apply some extra effort. We use a novelty factor to solve this ambiguity, i.e., choose the NE as candidate answer which is not present in the query. In the third case, the QA system simply fails to answer the question.

8 Experiments

As mentioned earlier, the work presented here is focused on factoid questions. Also we had to build our own corpus for experimentation. Corpus preparation and experimentation are described in the following subsections.

Corpus: As per our knowledge, there is no Bengali corpus available for QA research. So we had no other choice but to build our own corpus for experimentation. Fourteen documents from the geography and agriculture domains were acquired from the Wikipedia. Twenty Bengali language experts were involved in this small corpus development work. A total of 184 factoid questions were prepared and annotated according to three question answering based levels, namely Question Class (kappa - 0.91), Expected Answer Type (kappa - 0.85) and Question Topical Target (kappa - 0.89).

Results: Mean Reciprocal Rank (MRR) metric is used to evaluate the QA system. MRR is formulated as follows:

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

where, $rank_i$ represents the best rank of the correct answer of the i^{th} question and $|Q|$ is the number of test questions. Corpus statistics and evaluation results are shown in Table 1 and Table 2, respectively.

Table 1. QType Statistics

Type	Geography	Agriculture	Overall
Person	22	0	22
Organization	4	6	10
Location	6	11	17
Temporal	47	13	60
Numerical	60	15	75

9 Conclusion

This paper presents the first attempt to build a factoid QA system for Bengali. We propose an architecture to address the scenarios common to low-resource languages particularly for Indian languages. Also, we discussed the major challenges of developing

Table 2. Corpus Statistics and System Evaluation

Domain	#Documents	#Questions	MRR
Geography	10	139	0.34
Agriculture	4	45	0.31
Overall	14	184	0.32

a QA system for Bengali. We proposed a sentence ranking strategy for the BFQA system. However, it was observed from the experiments that the accuracy of the system is not at par with those for the European languages. The probable reasons for the somewhat poor performance of the system can be attributed to the low accuracies of the shallow parser and the NER system as the accuracy of factoid QA system is largely dependent on the performance of the NER component and the parser. In absence of any gold standard test set, we also prepared our own test set to evaluate the system.

Acknowledgements We acknowledge the support of the Department of Electronics and Information Technology (DeitY), Ministry of Communications and Information Technology (MCIT), Government of India funded project *"CLIA System Phase II"*.

References

1. Zheng, Z.: AnswerBus question answering system. In: the second international conference on HLT Research,pp. 399–404. Morgan Kaufmann Publishers Inc.(2002)
2. Ittycheriah, A., Franz, M., Zhu, W. J., Ratnaparkhi, A., Mammone, R. J.: IBM's Statistical Question Answering System. In: TREC.(2002)
3. Green Jr, B. F., Wolf, A. K., Chomsky, C., Laughery, K.: Baseball: an automatic question-answerer. In: western joint IRE-AIEE-ACM computer conference, pp. 219-224, ACM. (1961)
4. Woods, W. A., Kaplan, R. M., Nash-Webber, B.: The lunar sciences natural language information system: Final report. BBN Report 2378, Bolt Beranek and Newman Inc. (1972)
5. Mohammed, F. A., Nasser, K., Harb, H. M.: A knowledge based Arabic question answering system (AQAS). ACM SIGART Bulletin, 4(4):21-30.(1993)
6. Kanaan, G., Hammouri, A., Al-Shalabi, R., Swalha, M.: A new question answering system for the Arabic language. American Journal of Applied Sciences, 6(4), 797.(2009)
7. Hammo, B., Abu-Salem, H., Lytinen, S.: QARAB: A question answering system to support the Arabic language. In: ACL-02 workshop on Computational approaches to semitic languages,pp. 1–11.(2002)
8. Sakai, T., Saito, Y., Ichimura, Y., Koyama, M., Kokubu, T., Manabe, T.: ASKMi: A Japanese Question Answering System based on Semantic Role Analysis. In: RIAO,pp. 215–231).(2004)
9. Isozaki, H., Sudoh, K. Tsukada, H.: NTT's japanese-english cross-language question answering system. In Proceedings of the NTCIR Workshop 5 Meeting. pp. 186–193. (2005)
10. Yongkui, Z. H. A. N. G., Zheqian, Z. H. A. O., Lijun, B. A. I., Xinqing, C. H. E. N.: Internet-based Chinese Question-Answering System. Computer Engineering, 15, 034. (2003)
11. Sun, A., Jiang, M., He, Y., Chen, L., Yuan, B.: Chinese question answering based on syntax analysis and answer classification. Acta Electronica Sinica, 36(5), (2008)

12. Banerjee, S., Bandyopadhyay, S.: Bengali Question Classification: Towards Developing QA System. In: SANLP-COLING 2012, IIT,Mumbai,India. (2012)
13. Banerjee, S., Bandyopadhyay, S.: An Empirical Study of Combining Multiple Models in Bengali Question Classification. In: IJCNLP. Nagoya, Japan. (2013)
14. Banerjee, S., Bandyopadhyay, S.: Ensemble Approach for Fine-Grained Question Classification in Bengali. In: PACLIC-27. National Chengchi University, Taiwan. (2013)
15. Moldovan, D., Pasca, M., Harabagiu, S., Surdeanu, M.: Performance issues and error analysis in an open-domain question answering system. ACM Trans. Inf. Syst., 21:133-154.(2003)
16. Prager, J.: Open-Domain Question-Answering. In: Foundations and Trends in Information Retrieval. Now Publishers (2007)
17. Mollá Aliod, D.: AnswerFinder in TREC 2003. In: TREC. pp. 392–398.(2003)
18. Chen, J., Diekema, A.R., Taffet, M.D., McCracken, N., Ozgencil, N.E., Yilmazel, O., Liddy, E.D.: Question answering: CNLP at the TREC-10 question answering track. In: TREC. (2001)
19. Hovy, E. H., Gerber, L., Hermjakob, U., Junk, M., Chin-Yew Lin.: Question Answering in Webclopedia. In TREC. (2000)
20. Saha, S. K., Chatterji, S., Dantapat, S., Sarkar, S., Mitra, P.: A Hybrid Approach for Named Entity Recognition in Indian Languages. In: IJCNLP-08 workshop on NER for South and South East Asian Languages. Hyderabad,India. pp. 17–-24. (2008)
21. Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R., Girju, R., Goodrum, R., Rus, V.: The structure and performance of an open-domain question answering system. In: ACL. pp. 563–570. (2000)
22. Monz C.: From Document Retrieval to Question Answering. Ph.D. thesis, University of Amsterdam. (2003)