IMPROVING AN OPEN SOURCE QUESTION ANSWERING SYSTEM

CS 297 Report

Presented to Dr. Chris Pollett

Department of Computer Science

San Jose State University

In Partial Fulfilment

Of the Requirements of CS 298

By

Salil Shenoy

Contents

1 Problem Statement

In todays world, a large amount of data is available on the world wide web. Over the years, multiple techniques have been developed to process the data and retrieve useful information from it. The Question Answering System [1] is one such system which can be used to retrieve information. Question Answering System is a computer science discipline within the fields of information retrieval and natural language processing, which is concerned with building systems that automatically answer questions posed by humans in a natural language. A Question Answering System can be implemented using different approaches like Regular Expressions (regex) [2], Natural Language Processing [3].
The project uses the open system search engine Yioop which is developed by Dr.Chris Pollett, San Jose State University. The Yioop summarizer creates a summary for each of the documents crawled. The Question Answering System in Yioop is designed so that it will use the summary and extract information and form triplets of the form [Subject-Predicate-Object]. The triplet format may be different depending on the language for which the Question Answering System is developed. The goal of the project is to integrate an existing patch for Question Answering System developed by Niravkumar Patel [4], make it support internationalization and develop a similar system for Indian languages like Hindi, Marathi, etc. The system will include utilizing various aspects of natural language processing to extract the triplets.The triplets extracted will be stored in the index to make the retrieval process efficient.
In the report, I will discuss my deliverables for the CS 297. In the next section, I will discuss about the integration of the existing patch on Question Answering System in Yioop. The next section discusses the implementation of a Hindi Part of Speech Tagger and creation of a Hindi lexicon for Yioop. In the next section I will discuss how I refactored the question answering module code and made it locale specific. I will conclude the report by discussing how the work done in this semester will help me make progress in CS 298.

2 Deliverable 1 : Get existing patch to work with the current version of Yioop

The goal of this deliverable was to integrate a existing patch on Question Answering System into Yioop. To execute this task it was necessary to understand the Question Answering System its implementation and the way it functions.
For any information retrieval system to be effective it is necessary that it has a comprehensive database. As part of this deliverable I had to understand how this database is formed in Yioop. Yioop has its own summarizer which creates short summaries of the documents it has crawled. The summaries are then broken down and indexed. This information is used by the Question Answering System to answer the question posed by the user. The existing Question Answering System patch was developed to answer 'wh' questions posed in Yioop. In this deliverable, I worked on modifying the patch so as to optimize it and also to make it inline with the coding guidelines set aside for Yioop.
The goal of the deliverable was to get the existing patch, optimize it and make it work with the latest version of Yioop at that time. I downloaded the latest version of yioop at the time and applied the patch on it. Although the patch was working I had to make some changes to make it comply with Yioop coding guidelines. The changes I made included using variables to store value returned from a function, which helped reduce the number of function calls made by the Question Answering System. I also worked on refactoring some of the code wherein I removed some repeated code which may have had impact on the performance of the module. On completeing the refactoring I submitted a new version of the patch to Mantis. This patch was verified by Professor Pollet and after some more modifications has now been integrated with the latest version of Yioop. With the Question Answering System module now in Yioop we observe that it is able to answer 'WH' questions like 'who' efficiently.

3 Deliverable 2: Literature Review

The goal of this deliverable for me was to have a conceptual understanding of the working of a Question Answering Sytems. As part of this deliverable, I read papers on Question Answering Systems, Part of Speech Taggers, Triplet Extraction, etc.
My first goal was to know the functioning and architecture of a Question Answering System. As explained in [5] a Question Answering System has the following modules:

3.1 Query Processing

In a Question Answering System, the query processing module accepts as input a question in natural language. The function of the query processing module is to process and analyse the input question. The output is the classification of the question as belonging to any of the different types of questions supported by the system.

3.2 Query Generation

In query generation different techniques like a recursive decent parse tree or regex expressions are used to express the input question in a format understandable by the question answering system.

3.3 Database Search

The question generated from the query generation module is then given to the database in order to search possible results stored in the database. The related results which satisfy the given query with selected keyword and rules are sent to the next stage.

3.4 Related Documents

The results which are generated by the previous stage are stored in a document. This document is then used by the display answer module to extract the short answer to the

user question.

3.5 Display Answer

The result is stored as a document. The result is then converted into required text which is required by the user and displayed to the user.
After understanding the architecture of a Question Answering System, the next task was to understand the different components like Part of Speech Tagger [6] and Triplet Extraction [7]. For CS 297, the plan was implement a Part of Speech tagger for an Indian language like Hindi or Marathi, I started going through papers related to implementing a Hindi Question Answering System [8] and PoS tagger.

4 Deliverable 3: Create a part of speech tagger for Hindi

One of the main aspect of a Question Answering System is the ability to identify the different components of a given sentence. A Part of Speech (PoS) tagger is a module which helps identify the which part of speech each word in sentence belongs to. There are several approaches to developing a PoS using rule based approach, neural networks, HMM, etc. [9]. But the accuracy for each of these approaches is almost same whereas the complexity increases in that order. So, I have used the rule based approach mentioned in [10] to implement the Hindi PoS tagger.
The rules described in the paper are as follows:

- NOUN IDENTIFICATION

  RULE 1: If the previous word tagged is a Adjective / Pronoun / Postposition then the current word is likely to be a noun

  RULE 2: If the current word is a verb then the previous word is likely to be a noun

RULE 3: If the current tag is a noun then next / previous is likely to be a noun

- DEMONSTRATIVE IDENTIFICATION

  RULE 1: If the current and previous words are tagged as pronouns then the previous word is likley to be a demonstrative

  RULE 2: If current word is a noun and previous word is a pronoun then the current word is liklely to be demonstrative

- PRONOUN IDENTIFICATION

  RULE: If the previous word is unknown and cuurent word is a noun then the previous word is most likely to be a pronoun

- NAME Identification

  RULE: If we get two words which are untagged the most probably they form a name and will be tagged as noun

- ADJECTIVE IDENTIFCATION

  RULE: If the word ends with ¡tar¿, ¡tam¿, ¡thik¿ then we tag it as a Adjective

- VERB IDENTIFICATION

  RULE: If the current word is tagged as Auxilary verb and previous word is tagged as Unknown then most likely that the previous word is a verb

In order to build a data set, I selected short sentences from different sources. The main aim for the dataset was to test if the rule based Hindi POS Tagger implemented in Deliverable 3 is able to tag most of the words correctly. The observation is that although msost of the words are tagged correctly some of the words remain tagged as unknown. The reason for this may be the absence of well built lexicon to support the Hindi POS Tagger.
So for example, when I used a sentence as shown in Figure 1
The tagger output was as shown in Figure 2

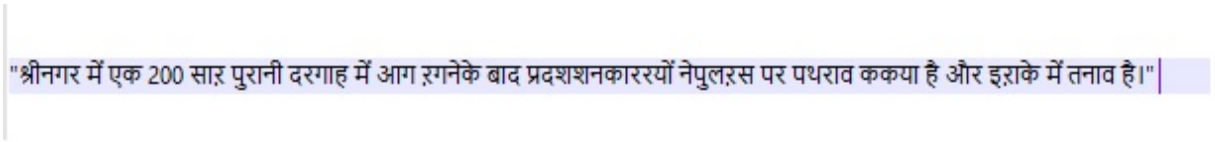

Figure 1: A Hindi Sentence.

श्रीनगर~UNKNOWN में~case एक~NN 200~NN साऱ~NN पुरानी~AJ दरगाह~UNKNOWN में~case आग~NN ऱगनेके~UNKNOWN बाद~NN प्रदशशनकाररयों~NN नेपुलऱस~NN पर~CONJ पथराव~NN ककया~UNKNOWN है~VB और~Q इऱाके~UNKNOWN में~case तनाव~NN है।~NN

Figure 2: Tagger Output.

So as we can see the result is a mix of 'UNKNOWN' and correctly tagged words.

5 Deliverable 4: Refactor the tokenization code for English QA system

The existing code for Question Answering System is maintained in its own class QuestionAnswerExtractor. The goal of this deliverable was to refactor the code so that it starts living in the Tokenizer file of any given locale. This would make it easy to add language specific functionality to Yioop as we would only need to change the Tokenizer of that locale and other modules of Yioop would not be affected. I implemented the refactoring in two steps as follows:

1. The question triplet formation while indexing
2. Answering the question asked in Yioop

My refactoring affected the following files:

1. Deleted the QuestionAnswerExtractor.php file as I was able to move all the code to the Tokenizer

2. Added all the business logic for the QA system ranging from question triplet formation and answer retrieval to Tokenizer

3. Minor change in PhraseModel and PhraseParser as the references to QuestionAnswerExtractor had to be changed to point to Tokenizer

I have uploaded a patch for this in Mantis: Issue Id: http://www.seekquarry.com/mantis/view.php?id=174 Note Id: 0000684

6 Conclusion and Future Goal

In CS 297, my first task was to understand the concept of Question Answering System and get an existing patch for a Question Answering System developed for Yioop to work with the lastest version of Yioop. My literature review included reading papers telling about different ways of information retrieval, different approaches for developing a Question Answering System, reading about how similar systems were developed for Indian languages like Hindi, Marathi, etc.I read about how a Part of Speech tagger can be developed using Rule Based Approach, Neural Networks, HMM, etc. As a follow up to my understanding of a Question Answering System and Part of Speech tagger, I implemented a rule based Hindi POS which I tested with small hindi corpus. My next task was to refactor the existing code for Question Answering System in Yioop, I did this by moving most of the code to language specific locales, the goal being to eliminate extra layers in code, to make adding similar system for other languages easy and also going ahead this may help improve the efficiency. For future work, I will work on making the hindi lexicon more robust and add more rules to Hindi Part of Speech tagger. I will also work on implementing a parse tree generator for Hindi. I also plan to work on implementing a similar system for another language.

References

[1] Question Answer System: http://www.sciencedirect.com/science/article/pii/S1319157815000890

[2] Keelj, Vlado, and Anthony Cox. "DalTREC 2004: Question Answering using Regular Expression Rewriting."

[3] Cooper, R. J., and Riiger, S. M. A Simple Question Answering System. In AUTHOR Voorhees, Ellen M., Ed.; Harman, Donna K., Ed. TITLE The Text REtrieval Conference (TREC-9)(9th, Gaithersburg, Maryland, November 13-16, 2000). NIST Special Publication. INSTITUTION National Inst. of Standards and Technology, Gaithersburg, MD.; Advanced Research Projects Agency (DOD), Washington, DC. (p. 208).

[4] Existing work done on Question Answer System Question-Answer System patch for Yioop by Niravkumar Patel, 2015.

[5] http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.681.843&rep=rep1&type=pdf

[6] Part of Speech Tagger http://research.ijcaonline.org/volume34/number8/pxc3875993.pdf

[7] Triplet Extraction From Sentences by Delia Rusu, Lorand Dali, Bla Fortuna, Marko Grobelnik, Dunja Mladeni. 2007.

[8] S. Sahu, N. Vasnik, and D. Roy, "PRASHNOTTAR: A HINDI QUESTION ANSWERING SYSTEM," International Journal of Computer Science and Information Technology (IJCSIT), vol. 4, no. 2, pp. 149-158, Apr. 2012.

[9] https://pdfs.semanticscholar.org/1b4e/04381ddd2afab1660437931cd62468370a98.pdf

[10] http://www.aclweb.org/anthology/C12-3021.pdf