Question Answering System using PLSA

CS-365 Project Report

By

AnuragGautam ,HarshitMaheshwari
Advisor :Dr. Amitabha Mukherjee
Dept. of Computer Science and Engineering
IIT-Kanpur ,India
{ganurag, harshitm, amit}@iitk.ac.in

April 15, 2012

Abstract

Question answering systems is a wide and active area, with much scope of further research. In our project we look into the semantics of the web documents (taken from Wikipedia) to answer the questions, asked from the user. The principal algorithm used is PLSA to find the sentences, in which the topic is similar to the query. Then we use cosine similarity algorithm to rank the answers.

1 Introduction

In order for the computers to interact with the users more naturally,the computer must understand and infer what the user wants to say from what he actually says(the latent meaning). Question Answering System (referred to QAS henceforth) is basically a search program that strives to give the best possible answers to the questions asked from it by the users from the knowledgebase it has. The knowledgebase could be local or stored in a remote machine. Probabilistic Latent Semantic Analysis (PLSA) is a statistical technique for the analysis of two-mode and concurrent data. PLSA evolved from latent semantic analysis, adding a sounder probabilistic model. PLSA has applications in information retrieval and filtering, natural language processing, machine learning from text, and related areas.

There is a two-fold complexity of the question which must be considered, before answering a question:

1. Acquire user’s true intentions from question
2. The answer should be complete in best possible way and at same time non-redundant.

We use Wikipedia as the web knowledge base as this specific knowledge database usually contains the entire information neededto answer a complex question.

2 Question Answering System

Our project was to use Wikipedia as the knowledge base for all questions. Hence, some part of our project code is Wikipedia specific. Our approach is described below:

2.1 Extraction of useful terms from the query

This was one of the challenging parts. The accuracy of the answer lies heavily on this process. Since, we extract the concept from the question asked and search for the same on Wikipedia if ever we made a mistake in extracting the relevant concepts, then we will not get the correct document from Wiki, forget about answering. So, the first step wastagging. It is done to find the group of words to be considered together. For example terms like ‘Artificial Intelligence’ should be considered together rather than two different words. We used nltk library for tagging. The words tagged with NN* were extracted. Contagious words tagged with NNP are considered to be one concept. Second step was to remove all the stop words. For this purpose we have a list of the stop words from ftp://ftp.cs.cornell.edu/pub/smart/english.stop.

2.2 Finding the knowledge database for answer retrieval

After the extraction step, we search for all the concepts on Wikipedia and download the web pages. From the numerous pages generated the first k pages are considered for further processing.In our program, we have taken the value of k to be 1, because on most counts the other pages generated were not relevant at all, and they adversely affected the quality of our answers. Though, this ‘k’ is configurable.

2.3 Parsing the html pages for the removal of html tags

This step involves the removal of the html tags from the generated web pages. We used BeautifulSoup html parsing library for this. This generates a plain text document without any tags. This is done so that PLSA algorithm can be applied effectively, and it also immensely reduces the dimension.

2.4 Term-Document Matrix Generation

A term document matrix (T-D matrix) is generated from the plain text document generated in the above step. First we tokenize the document into sentences. Every sentence is considered is to be separate document in its own regard. The T-D matrix is a 2 dimensional matrix whose row contains the terms and the columns contains the documents containing them.We also map the query in this matrix by considering it also a separate document in itself. This T-D matrix is then passed to the PLSA algorithm.

2.5 PLSA algorithm

PLSA is a statistical model which has been called aspect model. It introduces a conditional independence assumption, namely that d (for documents) and w (for words in them) are independent conditioned on the state of the associated latent variable. The basic equation for the model is

$$\boldsymbol{P(d,w) = \sum_{z \in Z} P(z)P(d|z)P(w|z)}$$

computationally PLSA involves Model Fitting with EM Algorithm.

The E-Step equation is

$$P(z|d,w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z' \in Z} P(z')P(d|z')P(w|z')}$$

The M-Step formulae

$$P(w|z) \propto \sum_{d \in D} n(d,w)P(z|d,w)$$

$$P(d|z) \propto \sum_{w \in W} n(d,w)P(z|d,w)$$

$$P(z) \propto \sum_{d \in D} \sum_{w \in W} n(d,w)P(z|d,w)$$

The T-D matrix is fed into the PLSA Algorithm. It returns the ZxD matrix, where Z is the number of topics sorted. Here we get the document vectors mapped into the concept space.

2.6 Answer Ranking

For ranking we use the standard cosine similarity algorithm. It simply calculates the dot product between the various document vectors and the query vector. The document vectors giving maximum value are considered to be the most relevant answer.

$$Sim(s_1, s_2) = \frac{\sum_{w \in W} P(w|s_1)P(w|s_2)}{\sqrt{\sum_{w \in W} P(w|s_1)^2 \sum_{w \in W} P(w|s_2)^2}}$$

2 Implementation Details

The *engine* of our program was coded in Python and we used JavaFX for developing the interface.

The Back End

For parsing the html tags we use Beautiful Soup html/xml parsing library. We use NLTK library for tagging, tokenizing and stemming. There was also a need for SciPy and NumPy libraries because of the PLSA algorithm. Apart from them PyYAML was another need.

The Front End

Front End is developed in Java, using JavaFX 2.0 libraries. There is no other external dependency.

We also, display an image on the program. This image is the first image produced by the first Wikipedia page. Usually this picture is also, quite relevant to the question asked and helps the user facilitate in understanding the answer generated.

3 Some Experimental Results

Some successful results of the queries were:

1. Who was the first man sent on moon by NASA?
2. How high is Mount Everest?
3. What is photosynthesis?
4. When did India became independent?

4.1 Success cases

Some success cases with TREK’08 are also listed:

1. When was London's Docklands Light Railway constructed?
   ANS: The station first opened up in 1987.The residents lost the case.

2. How far is Yaroslavl from Moscow?
   ANS: Yaroslavl (Russian: ??????????) is a city and the administrative center of Yaroslavl Oblast, Russia, located 250 kilometers (160 mi) northeast of Moscow. The historic part of the city, a World Heritage Site, is located at the confluence of the Volga and the Kotorosl Rivers.

4.2 Failure cases

There were also many **incorrect** results mainly because of the following reasons:

1. We were not able to take data from the correct Wikipedia page because of the complexity of the question and the words to be searched after step 2.1 were not able to generate the correct Wikipedia pages. This is mainly because the query extraction in our system is syntax driven.
2. The relevant data was not available on Wikipedia.
3. PLSA was not able to map the topics in the web knowledge database to the query correctly.

Some queries from the TREK’08 database for which our program failed along with the reason of failure are

1. What was the monetary value of the Nobel Peace Prize in 1989?
   There was no data available on Wikipedia for the question.

2. Who was President Cleveland's wife?
   PLSA was not able to correctly map the answer to the query.

3. What was the name of the US helicopter pilot shot down over North Korea?
   In this case we have 2 proper nouns: US and North Korea, which are related with each other by the incident of the pilot being shot down. Here the system fails because it is syntax driven and it searches for US and North Korea and nothing related to the incident.

4. What two US biochemists won the Nobel Prize in medicine in 1992?
   Similarly in this case also, there were many nouns like: US, biochemists, Nobel Prize and medicine. The system was not able to find the correct web page.

4 A different approach

Our program is different from most of the other question answering systems as they use a static knowledge base while our system dynamically searches for the query on Wikipedia and downloads the relevant pages. It can be easily modified to take data from any other web source.

Acknowledgements: The authors would like to acknowledge Matthieu for his PLSA code that is modified and used for this project. The Sarvagya logo is inspired by Google logo. We will also, like to thank Dr. Amitabh Mukherjee for stimulating discussions and helpful hints.

Codes and Libraries Used

nltk library	http://www.nltk.org
Beautiful Soup	http://www.crummy.com/software/BeautifulSoup/
PyYaml	http://pyyaml.org
NumPy	http://numpy.scipy.org
SciPy	http://scipy.org
Matthieu's Log	http://www.mblondel.org/journal/2010/06/13/lsa-and-plsa-in-python/
JavaFX2.0	http://www.oracle.com/technetwork/java/javase/downloads/index.html

References

[1] Han Ren, DonghongJi, Chong Teng, Jing Wan (2011).A Web Knowledge Based Approach for Complex Question Answering
M.V.M. Salem et al. (Eds.): AIRS 2011, LNCS 7097 pp. 470–478, 2011.
http://www.springerlink.com/content/9646253882343438/

[2] Protima Banerjee and Hyoil HanModeling Semantic Question Context for Question Answering: Proceedings of the Twenty-Second International FLAIRS Conference (2009)pp 9-15 ,2009 http://www.aaai.org/ocs/index.php/FLAIRS/2009/paper/viewFile/31/235

[3] Thomas Hoffmann. Probabilistic Latent Semantic Indexing Published in: SIGIR'99 Proceedings of the 22 annual international ACM SIGIR conference on Research and development in information retrieval 1999. pp 50-57 http://dl.acm.org/citation.cfm?id=312624.312649

[4] Thorsten Brants, Francine Chen, IoannisTsochantaridis. Topic -Based Document Segmentation with Probabilistic Latent Semantic Analysis Published in: CIKM'02 Proceedings of the eleventh international conference on Information and knowledge management (2002).pp 211-218 http://dl.acm.org/citation.cfm?id=584792.584829

[5] TREC question answer database http://trec.nist.gov