Matthew Paras
IEMS 308
1/13/2019
Homework 4

Question and Answer System Report

Executive Summary

Question and answer systems can be used to quickly gain access to information given a body of text. For this report, we have prepared a question and answer (QA) system built off of articles taken from the Business Insider website (from 2013 to 2014). We have processed these articles in order to identify percentages, companies, and CEOs. From there, a QA system was built on top of Elasticsearch in order to generate answers to the following questions:

- Which companies went bankrupt in month X of year Y?
- Who is the CEO of company X?
- What affects GDP?
    - From here, be able to answer the following question based off an answer from the GDP question:
        - What percentage increase or decrease is associated with X?

The system can be used easily by leveraging the Elasticsearch offline client. It can run easily from the command line, where questions are taken as input. In the future, the model can be improved by performing more entity recognition in order to create more complex questions. In addition, more articles would only lead to better precision on the answers to the questions since the model could be improved.

The system that we used today does a very good job at answering the fact-based questions, however does a mediocre job at answering open ended questions, such as the "what affects GDP" question. In addition, identifying the percentages associated with the responses from the open-ended question is difficult, because the chosen method of selecting answers was naïve and not sophisticated.

Methods

In order to query the corpus, we employed Elasticsearch, which is an open-source software used for document retrieval. This made searching for documents easy. In order to search for documents, queries needed to be built from the questions.

Classifying Questions

Here, we used a very naïve rule-based approach in order to classify the kind of question. While we recognize that using a machine-learning/model-based approach to classify the questions would have been far more successful, we were limited in the fact that we did not have a training set to build off of. In addition, we were only expected to build a QA system to answer 4 different kinds of questions with fairly specific guidelines. The rules we used are as follows:

- Question has the word “bankrupt” or “bankruptcy”
    - Bankruptcy question
- Question has the word “CEO”
    - CEO of company question
- Question has “GDP”
    - “what affects GDP” question
- Anything else:
    - Follow-up question to GDP

The rules are not amazing by any stretch of the imagination, however they work for straightforward queries. They can easily be broken by bad queries, however we expect the user to properly input queries.

Querying Elasticsearch

Once the input question is classified, we parse the question (once again, naively and in a not sophisticated manner) and generate queries using Lucene query strings. From here we are easily able to query Elasticsearch and pull the top 50 documents. For bankruptcy questions, we search for articles from the given month and year that contain ‘bankrupt’ or ‘bankruptcy’. For CEO questions, we search for articles that contain the phrase ‘CEO’ and the given company name. For GDP questions, we search for articles that contain the phrase GDP and any variant of ‘effect’ or ‘affect’. For the follow up question, we search for the articles that contain the phrase GDP and the given property that would affect GDP.

Finding Answers from Query Results

Bankruptcy

We go through the top 50 articles and gather all the sentences that have the string “bankrupt” in it. From here, we identify candidate companies from those sentences similar to how we did in the homework 3 text analytics (and we use the same code from before). We then return the company that was referenced the greatest number of times.

CEO

We follow a similar process here, going through the top 50 articles and identifying sentences that have “CEO” in them. From here, we identify candidate CEO names from those sentences similar to how we did in the homework 3 text analytics (and we use the same code from before). We then return the CEO name that was referenced the greatest number of times.

What factors affect GDP?

This question will just return the same thing every time based on how its classified. In order to identify what factors affect GDP, we used tf-idf on single and double words (unigrams and bigrams), then select the top 50 and remove stop words from there.

Factor’s impact on GDP

From here we follow a similar pattern as before. We identify all of the sentences in the article that contain “GDP”. From there, we identify all of the percentages in the sentences and return the most common one.

Results

User interface

The interface for the user is such that they can run the python script and input questions. The response for each question will be sent back out. In order to run the python script, the user should download and run elasticsearch, then run the python script. Here is an example of the input that the user would see:

```
##########################################
##WELCOME TO QA SYSTEM IEMS 308 VERSION 1.0##
-----Source code at github.com/mattwparas----
##########################################


Enter a question: Who is the CEO of Facebook?
Mark Zuckerberg


Enter a question: What company went bankrupt in july of 2013?
Detroit


Enter a question: This is a test question??
```

Sample Question Results

Bankruptcy

Companies going bankrupt at date

```
print(classify_and_parse_question("What company went bankrupt in April of 2014?"))
print(classify_and_parse_question("What company declared bankruptcy in November of 2014?"))
print(classify_and_parse_question("What company went bankrupt in February of 2013?"))
print(classify_and_parse_question("What company went bankrupt in July of 2013?"))
print(classify_and_parse_question("What company went bankrupt in August of 2013?"))
print(classify_and_parse_question("What company went bankrupt in September of 2013?"))
```

```
Apple
Apple
Fisker
Detroit
Detroit
Detroit
```

Here, we can see that the system does not properly return the result for the first two questions. However, it was able to correctly identify that the company Fisker went bankrupt in February of 2013. In addition, while Detroit is not explicitly a company, the city of Detroit did declare bankruptcy in July of 2013 (and was apparently much discussed in the news in the following months). While this is a flaw in the system, in particular in the way we calculate and return the answer since it is just based on quantity of references, it is interesting to note that Detroit was correctly returned as an answer.

CEO

CEO of companies

```
print(classify_and_parse_question("Who is the CEO of Goldman Sachs?"))
print(classify_and_parse_question("Who is the CEO of Facebook?"))
print(classify_and_parse_question("Who is the CEO of Apple?"))
print(classify_and_parse_question("Who is the CEO of Apple Inc?"))
print(classify_and_parse_question("WHo is the CEO of Tesla?"))
```

```
Lloyd Blankfein
Mark Zuckerberg
Tim Cook
Chief Executive
Elon Musk
```

The system does correctly classify the CEO for 4 out of the 5 questions, however struggled to identify the CEO of Apple Inc. While unfortunate, in general the system was able to identify correctly the CEO of some very large and popular companies.

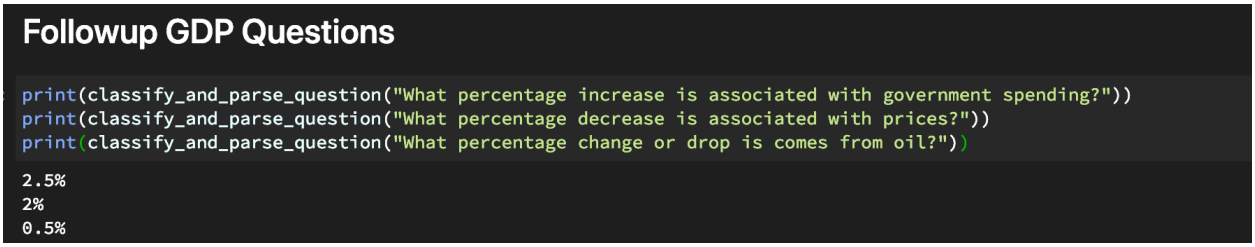
What factors affect GDP?

For convenience, here are the factors returned:

‘japan', 'policy', 'debt', 'weather', 'us', 'government', 'said', 'fed', 'percent', 'spending', 'prices', 'year', 'would', 'economic', 'economy', 'gdp', 'in the', 'growth', 'of the', 'oil'

Obviously, “said”, “would”, and “of the”, should be removed from this output list, so the responses should have been more properly cleaned of stop words. In addition, some of these terms may not directly impact the GDP, but this is an extremely broad questions and the results are not completely terrible.

Factor’s impact on GDP



Here, it is difficult to say if the QA system is properly returning good results. These are high level economic questions, but the system does return percentages associated with the given factors based off of the number of occurrences in the articles.

Conclusions and Next Steps

Conclusions

- It is difficult to build an all-encompassing QA system, so focusing on more narrowed questions yields fairly good success
- More advanced selection methods may work better, but for a simple QA system following “the most common” selection method for picking CEO and companies worked fairly well.
- The system does a better job of answering questions that have direct answers. The question “What factors affect GDP”, does not yield amazing results since it is such a broad question.

Next Steps

- Add more documents into the system and use a better entity recognition model to identify CEOs and companies, rather than just the regex that I used.
- Build parsers and queries for other fact-based questions that could be answered
- Build a training set of questions so that we can avoid using the rule-based approach that we used for this model (that performs poorly).