**Introduction**

Basic QA System is a system to answer a question given in a test Json file. The system contains 3 main parts, which are sentence retrieval, entity extraction and answer ranking. The report will first introduce methods the authors use to build Basic QA System, following by error analysis. Then, the report will introduce a new system called Enhanced QA system as an improvement over Basic QA System. Lastly, the report will end with evaluation of the new system and future improvement.

**Basic QA System**

The base QA system correctly answers 1,028 questions out of 8,463 questions on **development** set, so there are a room for improvement. (Kaggle score: 0.15681)

1. **Sentence Retrieval:**

The idea of this part is to get a sentence in an article which has the most similarity score to the question. TF\*IDF is used to achieve this. Sklearn package is used. Three types of model has been launched on test model, the first one is traditional TF\*IDF ways to calculate term frequency and inversed document frequency product as a scores, and by looping the query term and add all score in TF-IDF matrix rank out the highest score sentences as the query sentences.

The second way is Okapi BM25, beyond IDF, there anther three parameter k1 b k2, which considered the term frequency saturation and filed-length normalisation it is more reasonable since it limited the term frequency when it’s too high in sentences but acutely it is not true answers and also checked the importance of sentence average. The aggregate will be the scores to rank the sentences and query.

The third way, its implement by language model, the main method its used Dirichlet smoothing way to smooth the probability on the term occurrence probability over each document , there a parameter alpha which are smoothing parameter , and give the sentence the probability of each query terms product will be the score for this sentences.

In this Project, the final method is BM25, Advantage using BM25 because the speed the accuracy are the best out of those three. Therefore the test data sets are based on BM25 method to impute the relevant sentence

1. **Entity Extraction:**

The idea of this part is to identify entity for words in a sentence so that it can be used to determine which part of the sentence should be the answer to the question. Stanford NER package is used to identify the tag. The tags are PERSON, LOCATION, NUMBER, and OTHER. The label of the tag NUMBER has been ensured to assign to number.

1. **Answer Ranking:**

The idea of this part is to get part of the sentence as an answer. Rule-based classifier is selected. The rule is defined as follow.

1. Question type must be determined.

* Question structure must be determined.
  + WH word structure (e.g. ‘Where is the museum dedicated to Berliner located?’)
  + WH word with words behind structure (e.g. ‘What technology is used by night-vision devices?’)
  + Non WH word (e.g. ‘Did the FBI target MLK?’)
* Based on the structure, question type, lookup word, and focus word can be determined. Question type can be identify from WH word if the WH words are not what, which, or how; otherwise, the question type will be regarded as OTHER. Focus word is the word that follow WH word. Examples are as below.
  + Where is the museum dedicated to Berliner located?
    - Question type: LOCATION, Focus word: Berliner, museum, locate
  + What technology is used by night-vision devices?
    - Question type: OTHER, Focus word: technology night-vision, device
  + Did the FBI target MLK?
    - Question type: OTHER, Focus word: FBI, target, MLK

1. Question type and focus word will be used to determined which part of the sentence should be the answer.

* If the question type is not OTHER, the same entity words in the sentence, including every word with NN as part of speech (POS) in the sentence, excluding words in focus words in the question, are selected and put in X list.
* If the question type is OTHER, every word with NN as POS in the sentence, excluding words in focus words in the question, are selected and put in X list.

1. Each word in X list is scored by the distance to the focus words and the least-scored word in the list is selected as the part of answer (K).
2. POS of K is used to construct a noun phase for the complete answer. (e.g. One of the grammars is NP = NP NNP , NP = NNP NNP, NP = NP IN CD. If K is Act which is NNP, ‘Guam(NNP) Organic(NNP) Act(NNP) of(IN) 1950(CD)’ can be constructed.

**Basic QA System’s error analysis**

Errors from the basic QA system are the results that are not the same as the answers in the training set. Errors are found to be from each part of the basic QA system.

1. **Errors on Sentence Retrieval:**

Error: Retrieved sentences does not contain the correct answers. Sometimes, the sentences cannot be retrieved. Based on the **development** set, 55% of the total retrieved sentences (4,654 sentences) contains correct answers. Moreover, TF-idf is time consuming.

Possible reason: TF-idf cannot get correct sentences since the questions may be completely different from the article. For example, when the question ‘who was the runner up’ is queried against the article, there is no sentence retrieved from the article because there is no ‘runner’ in the article.

Potential improvement: The improvement can be done by changing algorithm. The new algorithm includes BM25 combining language model and word semantic similarity. This method should outperform TF-idf in terms of time consumption and precision.

1. **Errors on Entity Retrieval:**

Error: Stanford NER gives wrong entity to a word. For example, ‘1980’ should be recognised as NUMBER but it is recognised as OTHER. Another example is ‘0.9–14’. The entity should be NUMBER; however, the entity is recognised by Stanford NER as OTHER. Another example is ‘Christian Dior’. The entity should be ORGANIZATION; however, the recognised entity is PERSON.

Possible reason: Stanford NER alone cannot correctly recognises the entity.

Potential improvement: Stanford NER entity recognition can be improved by rule. The rule is created by human looking through the entity. This way is inefficient and time-consuming.

1. **Errors on Answer Ranking:**

Since the rule selects only one word that should be a part of the answer and construct complete answer, there are 2 errors that may occur.

Error: The word is not in the answer.

Possible reason: There is no semantic structure considered; therefore, the closest word to the focus words is selected.

Potential improvement: Word2Vec can be used to determine which word is closest to the question and change the focus words to the word then reconsider the answer word.

Error: The word is in the answer but the construction of the answer is wrong.

Possible reason: There is no rule to construct the answer.

Potential improvement: The rules can be constructed by extracting POS of the answer in conjunction with the question type.

**Enhanced QA system**

Enhanced QA system consists of 3 parts: Sentence Retrieval, Sentence Partitioning, Question Type Detection, Answer Ranking. (Kaggle score: 0.20231)

1. **Sentence Retrieval:**

BM25 is used instead of TF-idf. Based on the **development** set, 65% of the total retrieved sentences (5500 sentences) contains correct answers. This is a big improvement since the increase is 846 sentences. Therefore, the number of answers is likely to increase.

1. **Question Type Detection:**

First, training set is processed into feature and result columns. Feature column contains all question in the training set. Result column contains results from Stanford NER on actual answer. This new data is used to train Logistic Regression Classifier. Below is the confusion matrix for the classifier.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| LOCATION | 0.40 | 0.37 | 0.38 | 607 |
| NUMBER | 0.74 | 0.80 | 0.77 | 2076 |
| OTHER | 0.72 | 0.76 | 0.74 | 4143 |
| ORGANIZATION | 0.43 | 0.17 | 0.24 | 897 |
| PERSON | 0.49 | 0.62 | 0.55 | 740 |
| AVG/TOTAL | 0.65 | 0.67 | 0.65 | 8463 |

Based on the opinion of the authors, average F1-score of 0.65 is reasonable.

1. **Sentence Partitioning:**

The idea is partitioning potential noun phases off a target sentence. The sentence is first tagged using Stanford POStagger and Stanford NERtagger. Then, using nltk’s RegexpParser, POS rules are created to capture noun phases (NPs). NPs of a sentence are put into an answer pool for each question, which is passed onto the next part of the Enhanced QA system. Development set is used to test the rules and there are 4,128 out of 8,463 answer pools, which contain correct answers, given that the question types are known. NER tags are used in the 4th part of the system.

1. **Word2Vec Similarity score:**

This part is used to create semantic scores for NPs and the question.

1. **Answer Ranking:**

This part is separated into 4 scoring processes.

1. Penalty score is given to the NPs that contains exactly same words appeared in the question.
2. Rewarding score is given to the NPs that locate closer to focus words (words that appears in the question)
3. Rewarding score is given to the NPs that has high **Word2Vec** similarity score to question core words (question core words are words between WH word and 1st verb in the sentence).
4. Rewarding score is given to the NPs that has same entity type as the question type.

For example, given the below question.

Question: “How many people currently call Guam home as of 2015?”

1. From sentence retrieval of the system, the answer sentence is given below.

Answer sentence: “In 2015, 161,785 people resided on Guam.”

1. The system detects question type from the question.

Question type: NUMBER

1. The system finds noun phases in the sentences.

NPs: [“167,785”, “2015”, “people”, “Guam”]

1. The system tries to rank the NPs.

* 1st process: Penalty score to NPs with same words

[(“167,785”, 0), (“2015”, -1), (“people”, -1), (“Guam”,-1)]

* 2nd process: Rewarding score to NPs close to focus words

Score is calculated by summation of the distance from focus words divided by total distance from every word

[(“167,785”,0+), (“2015”, -1+), (“people”, -1+), (“Guam”,-1+)]

* 3rd process: Rewarding score to NPs with **Word2Vec** similarity score to core words.

[(“167,785”,0.23+0), (“2015”, -0.73+0), (“people”, -0.81+0), (“Guam”, -0.69+1)]

* 4th process: Rewarding score to NPs with same entity as question type.

[(“167,785”,0.23+1), (“2015”, -0.73+1), (“people”, -0.81+0), (“Guam”,0.31+0)]

Therefore, the final score is [(“167,785”,1.23), (“2015”,0.27), (“people”, -0.81), (“Guam”, 0.31)]. The NP “167,785” has the highest score and it should be the correct answer.

**Evaluation**

Evaluation is done on each part of the system.

1. **Sentence Retrieval:** 3 methods including TF-idf, BM25, and Language model tested on **development** set

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tf-idf | BM25 | Language model |
| Ratio between correct retrieved sentences and total questions | 0.584308164953 | 0.647879002718 | 0.638071605814 |

1. **Question type detection:** 2 methods of **supervised machine learning model** including LogisticRegression, and RandomForestClassifier trained on **training** set and tested on **development** set

|  |  |  |
| --- | --- | --- |
|  | LogisticRegression | RandomForestClassifier |
| F1-score | 0.65 | 0.63 |

1. **Answer Ranking:** 2 methods including Basic QA system and Enhanced QA system tested on **development** set and given that the correct sentence is retrieved and question type is known.

|  |  |  |
| --- | --- | --- |
|  | Basic QA’s Answer Ranking | Enhanced QA’s Answer Ranking |
| Ratio between correct answers and total questions | 0.1715703651187522 | 0.1857497341368309 |

Based on the above BM25 is selected to be the main method for sentence retrieval, LogisticRegression is selected as a question type classifier, and Enhanced QA’s Answer Ranking is used.

**Future Improvement**

Deep learning can be used to improve the system. To apply deep learning to the system, redesigning the QA system may be needed. One of the state-of-art methods which the authors are interested in is Convolutional Neural Network (CNN). According to Feng et al (2015), CNN gives minimum accuracy score of 0.592 which is relatively high considering Kaggle’s baseline score of 0.15578.

Reference

Feng, M., Xiang, B., Glass, M. R., Wang, L., & Zhou, B. (2015). Applying Deep Learning to Answer Selection: A Study and An Open Task.