An Approach for the Chinese Question-answer System based on Document

Benyou Wang[1], Jiabing Niu[1], Liqun Ma[1], Yuhua Zhang[1], Lipeng Zhang[1], and Peng Zhang[1]

Tianjin Key Laboratory of Cognitive Computing and Application, School of Computer Science and Technology, Tianjin University, Tianjin, China

Abstract. Question Answering system has gradually become a new trend within the field of information retrieval and NLP. It outperforms the conventional search engines, for the system is able to answer users' questions automatically and accurately. Question Answering system based on English corpus has developed rapidly, whereas the Chinese corpus based Question Answering system still has some problems remains to be solved. Thus, developing a new Question Answering model, which is characterized by dealing with features of Chinese corpus is extemely essentail. Different to the current deep learning model, our model uses the semantic and syntactic information in Chinese corpus and bases on the linearity of Chinese texts. Finally, our model turns out to perform better than other methods through experiments.. . .

Keywords: Question Answer, DBQA, semantic matching, Chinese QA

1 Introduction

Quesion Answering (QA) have catched a great attention with the development of Natural Language Processing and Information Retrieval techniques. One of the typical tasks named document-based quetion answer (DBQA) concentrates on finding the answers from the question's given documents. Comparing to the tranditional document retrieval task, DBQA system usually use a fluent natural language to express the query intent and a accurate result with discarding most unmatching candidates is needed.

Due to the shorter text in QA task, the term independency hypothesis and data sparsity become a more serious problems than the tranditional retrieval task. The relevance-based IR method like TFIDF or BM-25 cannot solve well this precise matching problems. Thus, word embedding technology [1] is considered to widely applied in some English QA system as well as the Chinese QA system. Moreover, The question text is natural language with complete syntax structures instead of some keywords in document-retrieve task. The sentence of a question should be considered as a sequence structure or a tree structure, which can pay different attention on different parts. In summary, a QA system should consider such problems simultaneously.

1) match the semantics-similar texts which may be synonymous paraphrased.

2) take the sequantial information of the question text into consideration, instead of a unordered set of words.

For the first problem, enumerating all the paraphrase rules of English or Chinese seems to be impossible. When we adopt the distributed representation, two words which is similar in semantics may have a closed embedding representations. Based on the term independency hypothesis, for a query *a cute pet*, *a dog* or *a cat* cannot be found in the return list, but the embedding-based method can. In Chinese, many similar words may share the same character based on the specific word formation of Chinese. For exmaple the two words "表演" or "演出" have the same meaning as "perform or show" in English. Some character-based technology may help a lot. In the Chinese expression habit, people are more likely to firstly elaborate the premise and then ask the related issues under such premise. In the bag-of-words model, a unordered set of words in questions will lose the information to distinguish between the premise and issues and focuse on the issues.

This paper elaborate an approach for the Open Domain Questin Answering shared sub-task of Document-based QA task in NLPCC-ICCPOL 2016. We conbine the count-based method and embedding-base method with a ensemble learning strategy. In order to adapt to the Chinese expression habit, we integrate the features of Chinese into both the count-based method and embedding-based method, which achieves significant improvement upon baselines in the final evaluation.

The aim of the task is to build a system for selecting the answers of thousands of question. The target answers are only supposed to be selected from the question's given document contains a set of answer sentences. The results will finally be evaluated by the evaluation metrics to figure out the performance of our system. For example, for the given question "俄罗斯贝加尔湖的面积有多大?" in training set, the participants should find the correct answer "贝加尔湖长636公里，平均宽48公里，最宽79.4公里，面积3.15万平方公里" from the candidate answers. The model should generate a set of relevance scores between the question and each answer sentence including the correct answer in the test set. The evaluation toolkit which has test sets labeled golden answer annotations to questions will rank all answer sentences according to MRR scores and MAP scores.

The structure of this paper can be listed as follows. Sec. 3 introduces the methods including features and model which we use to complete the shared task. Sec. 4 presents the experiments and results we get. Sec. 5 are discussions and conclutions.

2 Related Work

QA task focuses on automatically understand natural language questions and select or generate a answer (or more) which can match semantically the question. Due to the shorter text than the tranditional task of document retrieval, structured semantic information and the lexical gap are two key points for QA

system. For the first point, tree [2] or sequential [3] structure have been proposed to utilize the syntactic information instead of a unorderd bag-of-word model. Some efforts like lexical semantics [4], probabilistic paraphrase or translation [5] have been made to alleviate the problem of lexical gap. Moreover, feature-based ensemble method [6] try to combines both the semantic and syntactic information to rank the answers. Recently, the end-to-end strategy motivates us to build a deep symantic matching model which can also model the sequential text. With the development of the embedding-based neural network, deep learing [7] [8] have achieved a good performance in the QA task. Severyn et.al. propose a shallow convolutional neural network (CNN) which combines the ordered overlapped information into the hidden layer [9]. Recurrent neural network(RNN) and following long short-term memory neural network (LSTM) [10] [11] which can model the sequential text is also applicable for the represention and matching of Quetion and answers. Santos et.al. proposes an attentive pooling networks with two-way attention mechanism for modelling the interactions between two sequential text and easily integrating a CNN or RNN network [12]. In Chinese, we need a more

中文QA 的发展 我们的方法

3 Methods

We adopt a classical process for a competition task, with the successive steps from data exploration (in Sec 3.1), data preprocess (in Sec 3.2), feature extraction (in Sec 3.3) to model selection (in Sec 3.4).

3.1 Data Exploration

In this section, we will exploration the characters of the train data and test data.

basic statistics There are 181882 quetion-answer pairs with 8772 questions in the training set, and 122532 question-answer pairs with 5779 questions in the testing set. Every question have 20 candidate answers in the training set and 21 candidate answers in the testing set.

Table 1. the basic information of the training and testing set.

	training set	**testing set**
number of qa pairs	181882	122532
number of questions	8772	5779
average number of candidate answer	20.7	21.2
average length of the questions (charater)	46.3	46.2
average length of the answers (charater)	106.0	106.5
average number of correct answers	1.05	1.06

question types Different questions will have different information need. Question sentences' type is usually a kind of vital signal. It is a prerequisite but not sufficient condition when distinguishing whether the given sentence is a correct answer or not. For example of the question "中央大学的首任校长是谁?", a candidate answer can be correct only if it appears a name entity of person. We divide the question into the following categories:

Table 2. the number of different types of question.

	training set	**testing set**
time	950	861
number	2135	1433
person name	1049	526
place name	583	396
organazation name	185	137
others	3870	2646
whole		

word-level and character-level overlap For a question like "佛罗伦萨什么时候降水比较多？", the unique answer "降水主要集中在冬季" will share the same component "降水". In most Chinese word segmentation tools, "降水" can be segmented as a single word, which is usually considered as the minimal granularity of semantic unit. Except for some paraphrased cases, the answer will covers the issues of the question by overlapping some words with the question. In the whole training test, we get the trend as showed in the Fig. 1. It is easily found in the range from 0 to 13 of the x-axis, the more overlapped words between the question-answer pair, the more likely the pair matches. Moreover, the information of character-level overlap will cover many paraphrased patterns of Chinese. For example of the question "“年”字有多少笔？", the correct answer "笔划：6" have three words ["笔划"，":","6"] and do not have any overlapped word with the question. Fig. 2 shows the word-level overlap have the similar trends to be the correct answers.

Sequential structure informations Tranditional IR model like TF-IDF or BM25 model treats a query or a documents as a bag of words, in which the sequential information of structure are ignored. In the scenario of QA system with a shorter length of question and answer, sequential information may help a lot for the matching of the question-answer pairs and a more elaborate model which take the sequential information into consideration is needed. Roughly speaking, the word in different posistion of a sentence may reflect the different syntactic and semantic structures. For the example of the question "中央大学的首任校长是谁？"， the word "中央大学" in the forward position is the limited premise of

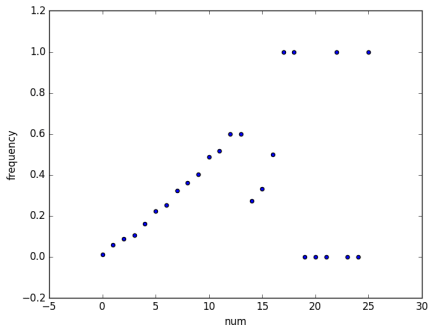

Fig. 1. x-axis means the number of overlapped words in both question and answer sentences. y-axis refers to the probable occurrence frequency of the target answer. Data scatters after the number of 14 for the lack of words samples.

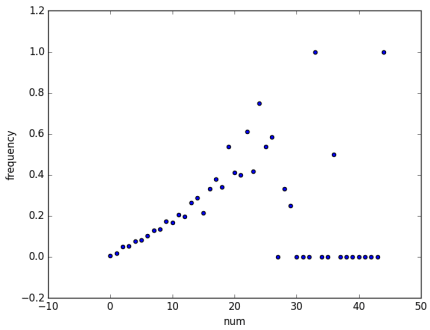

Fig. 2. x-axis means the number of overlapped characters in both question and answer sentences. While y-axis refers to the probability of becoming the target answer. Data are dispersed in the range between 22 and 50 because the samples are not enough.

the issues of the latter words "首任校长". The rearward word "首任校长" may be more related to the issues, which will contribute more to question-answer matching. In the training and testing set, we find the statistical correlation between the overlapped position and its corresponding probability of question-answer matching in Fig. 4 for word-level overlap and Fig. 4 for character-level overlap.

3.2 Data Preprocessing

Due to the lack of the obvious boundaries of Chinese text, we use the pynlpir package [1] [13] to segment both the question sentences and the answer texts. Stopwords [2] are removed for dropping the useless high-frequency words which are not discriminative and have little semantic meaning. In order to get the classfication information of answer, we adopt the LTP online API [3] for Named Entity Recognition (NER). The 300-dimention word embedding is provided by the NLPCC competition. Moreover, we have trained a embedding model of some crawled pages in Baidu Baike [4].

3.3 Feature Extraction

Questions' categories and Answers' classfication As mentioned in the Sec. 3.1, The questions can be divided into 5 categories. 3 of them are concerned

[1] https://github.com/tsroten/pynlpir
[2] stopwords in http://tcci.ccf.org.cn/conference/2016/pages/page05_evadata.html
[3] http://www.ltp-cloud.com/
[4] http://baike.baidu.com/

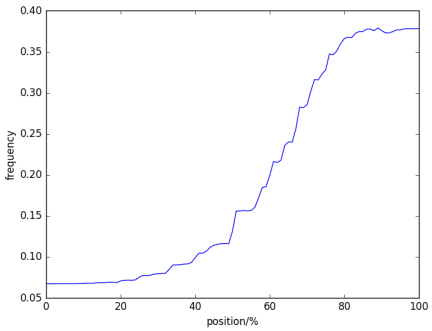

Fig. 3. x-axis refers to the position of the overlapped characters in question sentences. x=0 means that the overlapped character is on the front of a sentence.x=100% means the overlapped character is on the back of a sentence.y-axis means the occurrence frenquency of correct answers.

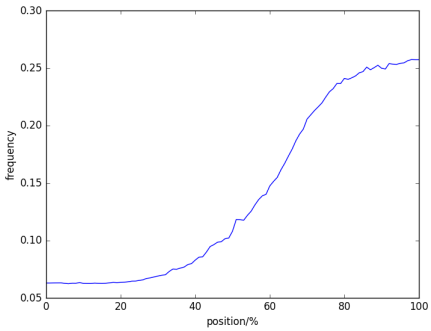

Fig. 4. Similar to Fig. 3, this figure shows the relationship between the position of the overlapped words in question sentences and the occurrence frenquency of correct answers.

with name entity, which are *person*, *place*, *organization*. The remaining two are *time* and *number*. Due to the lack of large-scale labeled datas, we can not adopt a learning-based question classifiers [14]. Alternatively, a template-based question classifiers can covers most cases for a simplified taxonomy. Answers' classfication may be a multilabeled task, which means an answer can belong to many categories. The first three NER-based categories can be recognized by the LTP online API. Meanwhile, we use templetes of regular expression to distinguish the types of *number* and *time*.

In practice, two methods are adopted. The first one is dummp Number, 5 categories are viewed as 5-dimention binary feature vectors which default value is unmatching. If the question is identified as one kind of category, the corresponding feature will be filled with matching or not. The second the methon is adopting one-dimention feature, which relect that the category of question-answer pair matched or not.

Overlap In Sec 3.1, there are a statistical correlation between the matching probability and the overlapped information. In these features, the removation of stop words seems to be vital due to the ignorance of the meaningless words. For Chinese text, it's hard for us to find the dictionay which can contains all the near-synonym pairs. An alternative approch is to use the charcter-based metric due to the fact that many synonymous paraphrased pairs shares the same charcter in Chinses. We calculate both the word-level and character-level score of overlap

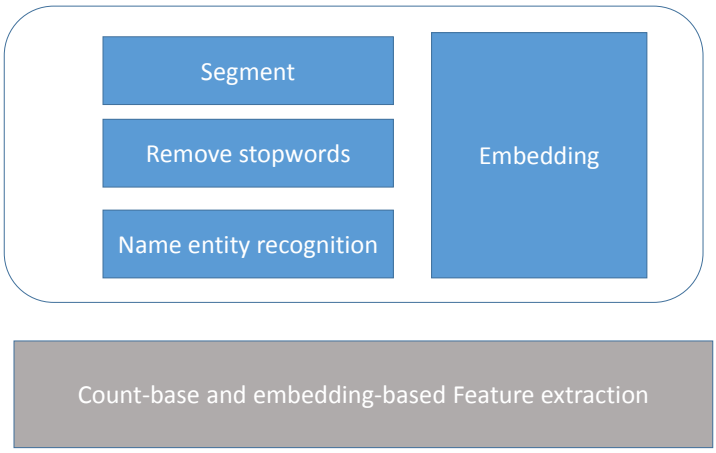

Fig. 5. the basic structure of the data prepare

as follow:

$$Score_{overlap}(Q, A) = \sum_{q_i \in Q}^{n} freq_A(q_i) \cdot weight(q_i) \tag{1}$$

Where the question sentence have n words(characters) and the answer sentence have m words(characters). The weighted model is based on the position of q_j in the sentence. $freq_A(q_i)$ is denoted as the smoothed frequency of the q_i in the answer A.

TF-IDF model and BM25 model Some tranditional IR methods which are based on bag-of-words model are also implemented.

$$Score_{bm25}(Q, A) = \sum_{q_i \in Q}^{n} IDF(q_i) \cdot \frac{f_i \cdot (k_1 + 1)}{f_i + k_1 \cdot (1 - b + b \cdot \frac{Length_A}{Lenght_{avg}})} \tag{2}$$

where f_i is the frequency of the q_i in the answer A. k_1, b are parameters which can be adjusting for the specific task. $Length_A$ and $Lenght_{avg}$ are the length of the answers A and the average length of the whole answers.

Other features Edit distance is usually used to measure the length of similar strings in English. While Jaccard index gives the similarity of morphemic sets between the questions and answers. The length of answer sentence is often considered as a significant feature.

Embedding-based approach Embedding technology embed words into a high dimensional semantic space, which makes it easier to find the relationship between words. Sentence is simply considered to have been lapped by words linearly. Different words can contribute different weights for the whole meaning of a sentence, which depends on their position, semantic structure and IDF. We get the representation of a word or Chinese character by the following approach.

$$Representation(S) = \frac{\sum_{i=0}^{n} weight(s_i) \cdot \overrightarrow{embedding(s_i)}}{\sum_{i=0}^{n} weight(s_i)} \tag{3}$$

s_i is the character or word in a sentence(question or answer), and $embedding(s_i)$ is the corresponding embedding vector. Then we calculate the inner product between the represention of question and answer as the final score.

Besides these linear combination of the inside word embedding, we have build a CNN model to test how much the question-answer pair matches.

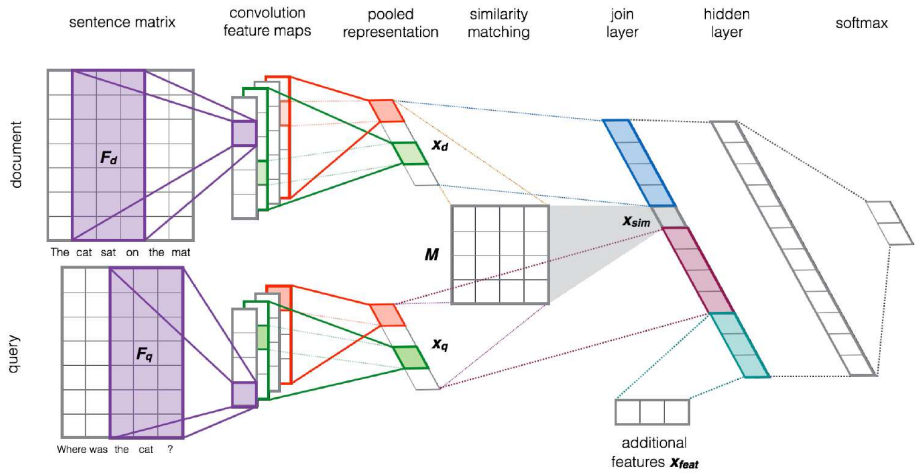

Fig. 6. The picture need to be repainted

3.4 Model Selection

We have presented various fundamental features in last chapter,which will directly effect the degree of how question and answer matches. We adopt a lin-

ear regression model and learn-to-rank model [15] to integrate those features [5]. While some ensemble learning methods like boosting and bagging models are also used. [16] [6]

4 Results

4.1 Dataset

The provided datasets [7] of the document-based quetion answer task contains a training data which have the label for the ground truth and a testing data which do not have the correct label. In the training dataset, each question has many candidate answers. The structure of the dataset can be illustrated by Tab.3, where 1 in the last column means the correct answer. The provided test set only contains questions and their candidate answers. Each submission should only includes a column of scores which will be evaluated by the evalutin toolkit.

Table 3. Training Data Structure

question	candidate answers	label
俄罗斯贝加尔湖的面积有多大?	贝加尔湖是世界上最深和蓄水量最大的淡水湖。	0
俄罗斯贝加尔湖的面积有多大?	它位于布里亚特共和国(Buryatiya)和伊尔库茨克州(Irkutsk)境内。	0
俄罗斯贝加尔湖的面积有多大?	湖型狭长弯曲，宛如一弯新月，所以又有“月亮湖”之称。	0
俄罗斯贝加尔湖的面积有多大?	贝加尔湖长636公里，平均宽48公里，最宽79.4公里，面积3.15万平方公里	1
俄罗斯贝加尔湖的面积有多大?	贝加尔湖湖水澄澈清冽，且稳定透明（透明度达40.8米），为世界第二。	0

4.2 Evaluation Metrics

Our Question Answering system will be evaluated by MRP and MAP. Mean Reciprocal Rank(MRR) mainly indicates that whether the recall results good or not depend on the rank of the correct answer, namely the higher the correct answer ranked, the better the results are.

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{4}$$

$|Q|$ stands for the total number of questions in the evaluation set. For the i^{th} question Q_i, $rank_i$ represents the position of the first correct answer in the generated answer set C_i. $\frac{1}{rank_i}$ equals 0, if C_i doesn't have the correspond answer with the golden answers A_i for Q_i.

Another evaluation metric is Mean Average Precision(MAP), which can be defined as follows.

[5] https://sourceforge.net/p/lemur/wiki/RankLib/
[6] http://xgboost.readthedocs.io/en/latest/
[7] http://tcci.ccf.org.cn/conference/2016/pages/page05_evadata.html

$$MAP = \frac{1}{|Q|}\sum_{i=1}^{|Q|} AveP(C_i, A_i)\ AveP(C, A) = \frac{\sum_{k=1}^{n}(P(k)\cdot rel(k))}{min(m,n)} \quad (5)$$

MAP mainly refers to the average precision of the results. If the correct anwser ranks high in the retrieved answer set, the value of MAP will be high accordingly. m denotes the number of correct answer sentences and n is the number of retrieved answer sentences. $P(k)$ is the precision at cut-off k, which is the rank in the sequence of retrieved answer sentences. $rel(k)$ equals 1 if the item at rank k is an answer sentence, and 0 if not.

4.3 Results

The competition have provided four baseline as follow:

Table 4. the four baseline.

method	**MAP**	**MRR**
Average Word Embeding	0.4610	0.4610
Machine Translation	0.2410	0.2412
Paraphrase	0.4886	0.4906
Word Overlap	0.5114	0.5134

Due to that the above methods are based on the bag-of-word model and do not have a learn-based mechanism, the performance is a little poor. In our approach, we get the result as follow:

Table 5. our approach.

method	**MAP**	**MRR**
BM25	0.4610	0.4610
weighted word overlap	0.2410	0.2412
weighted character overlap	0.4886	0.4906
weighted word embedding	0.5114	0.5134
weighted character embedding	0.5114	0.5134
word-based NN	0.5114	0.5134
character-based NN	0.5114	0.5134
Emsenble learning of all features	0.5114	0.5134

4.4 discussion

analsys our results

5 Conclusion and Future Work

In this paper, we report technique details of our approach for the sub-task of NLPCC 2016 shared task Open Domain Question answering. Some traditional methods and neural-network based methods have been proposed. In our approach, we conbine the characters of Chinese text with our models and achieve a good performance by a emsenble learning strategy. In our opinions, a effective repersentation which contains the sequencial(or tree-based) information of short text and the corresponding effective semantic matching are the two key factors of the QA system. An end-to-end system which is specifically applicable for Chinese may be the trend for Chinese Question-Answering system.

[17]

References

1. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *Computer Science*, 2013.
2. X. Yao, B. V. Durme, C. Callison-Burch, and P. Clark, "Answer extraction as sequence tagging with tree edit distance," in *Conference of the North American Chapter of the Association for Computational Linguistics*, 2013.
3. Z. Wang and A. Ittycheriah, "Faq-based question answering via word alignment," 2015.
4. W. T. Yih, M. W. Chang, C. Meek, and A. Pastusiak, "Question answering using enhanced lexical semantic models," in *Meeting of the Association for Computational Linguistics*, 2013, pp. 1744–1753.
5. G. Zhou, L. Cai, J. Zhao, and K. Liu, "Phrase-based translation model for question retrieval in community question answer archives," in *The Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, Usa*, 2011, pp. 653–662.
6. A. Severyn, "Automatic feature engineering for answer selection and extraction," in *EMNLP*, 2013.
7. L. Yu, K. M. Hermann, P. Blunsom, and S. Pulman, "Deep learning for answer sentence selection," *Computer Science*, 2014.
8. M. Feng, B. Xiang, M. R. Glass, and L. Wang, "Applying deep learning to answer selection: A study and an open task," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2015.
9. A. Severyn and A. Moschitti, "Learning to rank short text pairs with convolutional deep neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2015, pp. 373–382.
10. D. Wang and E. Nyberg, "A long short-term memory model for answer sentence selection in question answering," in *Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, 2015.
11. M. Tan, B. Xiang, and B. Zhou, "Lstm-based deep learning models for non-factoid answer selection," *Computer Science*, 2015.
12. C. D. Santos, M. Tan, B. Xiang, and B. Zhou, "Attentive pooling networks," 2016.

13. T. Liu, W. Che, and L. I. Zhenghua, "Language technology platform," in *COLING 2010, International Conference on Computational Linguistics, Demonstrations Volume, 23-27 August 2010, Beijing, China*, 2010, pp. 13–16.
14. X. Li and D. Roth, "Learning question classifiers," *Coling*, vol. 12, no. 24, pp. 556–562, 2003.
15. T. Y. Liu, *Learning to Rank for Information Retrieval.* Now Publishers,, 2009.
16. T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," 2016.
17. Y. Li, W. Li, F. Sun, and S. Li, "Component-enhanced chinese character embeddings," *Computer Science*, 2015.