WUST System at NTCIR-13 Short Text Conversation Task

Maofu Liu, Yifan Guo, Yan Yu
School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, China
liumaofu@wust.edu.cn, guoguoyifan@foxmail.com

Han Ren
Laboratory of Language Engineering and Computing, Guangdong University of Foreign Studies, Guangzhou 510006, China
hanren@whu.edu.cn

ABSTRACT

Our WUST team has participated in the Chinese subtask of the NTCIR-13 STC (Short Text Conversation) Task. This paper describes our approach to the STC and discusses the official results of our system. Our system constructs the model to search the appropriate comments for the query derived from the given post. In our system, we hold the hypothesis that the relevant posts tend to have the common comments. Given the query *q*, we firstly adopt the framework to extract the topic words from *q*, and retrieve the initial set of post-comment pairs, and then the post-comment pairs are used to match and rank to produce the final ranked list. The core of the system is to calculate the similarity between the responses and the given query *q*. The experimental results using the NTCIREVAL tool suggest that our system should be improved by combining with knowledge and features.

Categories and Subject Descriptors

I.2.7 [Artificial Intelligence]: Natural Language Processing - text analysis.

I.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing - linguistic processing.

General Terms

Experimentation

Team Name

WUST

Subtasks

Short Text Conversation (Chinese)

KEYWORDS

Short Text Conversation, Information Retrieval, Vector Space Model

1. INTRODUCTION

The STC is a core task of NTCIR -13. Given a new post, this task aims to reuse an appropriate comment from a large post-comment repository. The dialogue is one of the most difficult problems in artificial intelligence. A lot of related technologies have been developed, such as dialogue state tracking, natural language generation, and the STC is a much simplified version of this problem, one round of conversation formed by two short texts, with the former being a message from human and the latter being a response to the message from the computer [1].

In NTCIR-12, the STC is defined as an IR (Information Retrieval) problem to search for the appropriate comments matching the given query *q*, derived from the given post, from the dataset. Compared to NTCIR-12, besides the retrieval-based method, NTCIR-13 considers the generation-based method to generating new comments. The generation-based methods for STC fall into two categories, the statistical machine translation (SMT) method, and the generation models (e.g., recurrent neural network, RNN) [2].

The SMT-based method treats the response generation as a translation problem. The method is intrinsically unsuitable for response generation, because the response are not semantically equivalent to the post as in translation. And the SMT-based method often yield responses with grammatical errors and in rigid forms [3].

Recent progress in deep learning has raised the possibility of realizing generation-based STC in a purely neutralized way. The model based on the neural network is trained in an end-to-end fashion, and thus there is no need in building the system using linguistic knowledge [4].

We take STC as an IR problem. Given the new post, we assume the effectiveness of comments depends on the similarity between the new post and the old comment, or the similarity between the new post and the old post.

In this paper, we propose the model based on similarity to estimate whether the post and the given query *q* are relevant. We adopt the framework to extract the topic words from *q*, and retrieve the initial set of responses, and then the retrieved post-comment pairs are used to match and rank to produce the final ranked list. We have also trained a Word2Vec [5] model to generate more features. A series of strategies have been applied to selecting the best comment.

The experiment uses the NTCIR-13 retrieval repository, and the official evaluation measures of the STC task are graded relevance IR evaluation measures for navigational intents and compute these evaluation measures using the NTCIREVAL tool [2].

The remainder of this paper is organized as follows. Section 2 discusses our system architecture in details. Section 3 describes our evaluation results of the formal run on the test collection of STC task. Finally, we conclude our paper in section 4.

2. SYSTEM DESCRIPTION

Our system includes three main stages, i.e. data preprocessing, matching and ranking. Figure 1 illustrates our system architecture in detail.

2.1 Data preprocessing

There are some traditional Chinese in raw text. First, we convert traditional Chinese to simplified Chinese. Chinese word is the meaningful linguistic basic element, and we use Ansj [6] to split Chinese text into a sequence of words. This algorithm is based on Google semantics model and conditional random fields (CRF). The model produces the set *T* that contains topic words of the given query *q*, being prepared for retrieving the similar post from the dataset. We get the topic words by HanLP (Han Language Processing) [7] and TF-IDF (Term frequency–Inverse Document Frequency).

On the other hand, the system builds the inverted index table of the posts and words separately. The inverted index table of words records the frequencies and positional index locating which post contains the word. In the corpus of millions sentences, it is difficult to retrieve the most appropriate comment candidates for a new post quickly. In this way we do not need to scan all posts for each input, but instead retrieve a limit number of posts efficiently.

Finally, we trained a Word2Vec model [5] to generate most features. The continuous skip-gram model is an efficient method for learning high-quality distribution vector representations that capture a large number of precise syntactic and semantic word relationships.

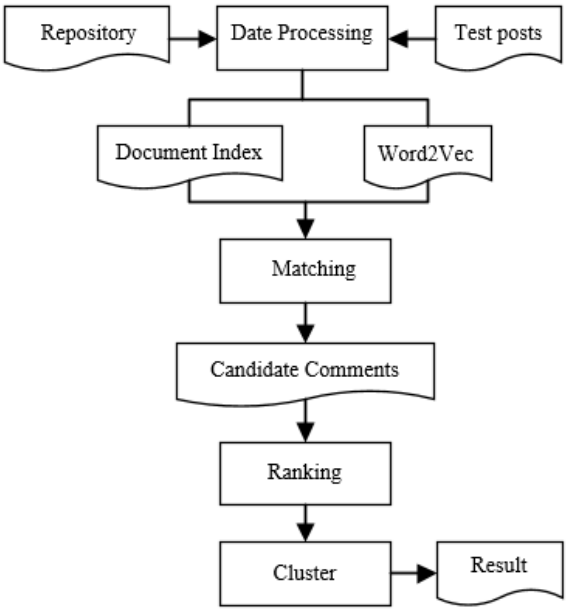

Figure 1. System overview

2.2 Matching

The main work of this stage is to find appropriate post-comment pairs for the given query *q*. The basic idea here is to find the posts being similar to the query *q* and use their comments as the candidates [1].

The system firstly retrieves a number of candidate posts by the topic words set *T*. And then, the system computes similarities between the query and posts, returns the top-N most similar posts, and produces the set $P_q^{reduced}$ of reduced candidate posts.

In our system, we use a simple VSM (Vector Space Model) for measuring the similarity between the query *q* and the candidate post *p* in the $P_q^{reduced}$. The score is more close to 1 if the two texts are more similar.

$$sim_{Q2P}(q,p) = \frac{\vec{q}^T\vec{p}}{\|\vec{q}\|\|\vec{p}\|} \quad (1)$$

Where $\vec{q}$ and $\vec{p}$ are the TF-IDF vector of *q* and *p* respectively. The assumption here is that if a post *p* is similar to the query *q*, its associated comments might be appropriate for *q* [1]. Moreover, the system further picks post-comment pairs from $P_q^{reduced}$. Our system measures the similarity between *q* and each of comments in post-comment pairs, and then produces the candidate comments set *C*.

We also use the VSM for measuring the similarity between the query *q* and the candidate comment *c* in *C*.

$$sim_{Q2C}(q,c) = \frac{\vec{q}^T\vec{c}}{\|\vec{q}\|\|\vec{c}\|} \quad (2)$$

Where $\vec{q}$ and $\vec{c}$ are the TF-IDF vector of q and c respectively.

The similarity methods are based on some textual features. We use some other simple matching features as follows.

(1) Inverse document frequency (IDF): the sum of the IDF of the common words in the sentence, based on a large set of Chinese sentences.

(2) Longest Common String (LCS): length of the longest common substring between the query *q* and the candidate response *r*.

(3) Overlapping similarity: The overlapping of some topic words in the two sentences. We use HanLP to extract the topic words. This algorithm is based on the TextRank.

$$\text{overlap}(q,c) = \frac{|q \cap c|}{\min(|q|,|c|)} \quad (3)$$

(4) Word2Vec similarity: the Word2Vec similarity is the cosine similarity between two word vectors, generated from the sentence word vectors with the additive synthesize method. The word to vector is an efficient tool for computing continuous distribution representations of words. Since every short text post or comment consists of several words, and it is available to combine vectors of words in these short texts to gain a representation of the whole text. Word2vec learns vector representations from training text corpus with skip-gram architecture.

2.3 Ranking

After all the features are obtained, the system uses a ranking function defined in Formula (4) to further evaluate all the comments in C, and assigns a ranking score to each candidate comment.

$$rank(q,c) = lcs * idf * keyoverlap * word2vecSim \quad (4)$$

Then, the system ranks the candidate comments based on their scores and returns the comments with top-200 comments to the given post. We also use cluster to get diversity answers.

3. EXPERIMENTAL RESULTS

Table 1 shows the statistics of the post-comment pairs repository and test dataset provided in the task. There are 219,174 Weibo posts and the corresponding 4,305,706 comments. There are 4,433,949 post-comment pairs. So each post has 21 different comments on average, and one comment can be used to response to multiple different posts.

Table 1. Statistics of the dataset for Chinese STC task

Retrieval Repository	posts	219,174
	post-comment pairs	4,433,949
Test Data	query posts	100

There are 100 posts being used for testing. In this paper, our work focuses on the retrieval-based method and then the system is asked to provide a ranked list of ten results for each given post. The comments must be those from the repository [2].

The official evaluation measures of the STC task are graded-relevance IR evaluation ones for the navigational intents. The official computes these evaluation measures using the

NTCIREVAL tool, which are nG@1, nERR@10 and P+ [2]. All the results of all the runs from teams are pooled to imitate manual annotation by the NTCIREVAL tool. The comments are labeled with L0 (inappropriate), L1 (appropriate in some contexts), and L2 (appropriate) by multiple judges.

Query q	移动设备改变生活，如今聚餐时确是这场景，我们成了信息的奴隶。 Mobile devices have changed life, and now dinner is really this scene and we become slaves of information.
Comment c1	都沦落成科技的奴隶了。科技改变生活、科技也限制生活 We became slaves of technology. Although science and technology changes life, they limits life.
Comment c2	果真是移动改变生活了。智能机的魅力。 It is amazing that intelligence machine really changed life.

We submitted two results of our system for STC task in Chinese by retrieval-based method. The runs have been sorted by Mean nG@1, P+, and nERR@10 respectively. Table 2 only lists the official evaluation results of our group. There are another 120 official evaluations from 22 teams for Chinese subtask, most of which better than ours. Comparing with the other runs, our system fails to produce the desired result.

Table 2. Part of official STC results

Run	WUST-C-R1	WUST-C-R2
Mean nDCG@1	0.071	0.094
Mean P+	0.0984	0.1409
Mean nERR@10	0.0927	0.1349

In most cases, a good response has many common words as the query. For example, in Table 3, when the query and the candidate responses both have "围脖(microblog)","喜欢(love)" and "几米(JiMi)", it is a strong signal that they are about similar topics. This measure, requiring no learning works on infrequent words, is easy and helpful in finding relevant responses.

Table 3. Post and comments holding common words

Query *q*	[泪]很喜欢几米每天分享的漫画围脖，大爱这个围脖！ I love the cartoons JiMi share every day on microblog.
Comment *c1*	真心喜欢这个围脖，真的很好！大爱 I really love the microblog. It is very good.
Comment *c2*	这个围脖的图都很赞。 The cartoons on the microblog are very good.
Comment *c3*	很喜欢几米，一个爱画画爱说故事的童话人 I like JiMi, a fairy tale who loves drawing and telling stories.

We find out that there are only 52 test posts returned appropriate comments by analysing the offical results of our system. So the average score is very low, and even many posts have zero score.

One reason is that our model uses the simple VSM rather than semantic similarity for measuring query-posts and query-comments similarities. If there are no words overlapping between the candidate comments and the query, the comments will rank very low even they are appropriate.

Table 4. An example of suitable comments ranking

Table 4 shows an example of the no words overlapping between the query and the comments. Two candidate responses are suitable to the query, while their ranks are very low. The main reason is the no words overlapping between the candidate responses and the query.

In addition, the high dimensionality and the sparsity are the characteristic of the short text. Some of the meaningful words maybe regarded as the stop ones. As a consequence, the segmentation results far away from achieving ideal effect, which weak the system more or less.

4. CONCLUSIONS

In this paper, we have described our model based on VSM for STC task in Chinese. We also analyzed our submitted experimental results and adjusted parameters which outperformed than former.

We think that there are two important ways to improve the efficiency of our model for STC task. We need to enhance the accuracy by combining other models, such as the topic-words, and to consider matching between query and response in terms of semantic relevance, speech act, and entity association.

In the future, besides on the basis of information-retrieval, we also would like to generate the appropriate and human-like response derived from what we searched from the post-comment pair repository.

ACKNOWLEDGMENTS

The work presented in this paper is partially supported by the Major Projects of National Social Foundation of China under Grant No. 11&ZD189 and Natural Science Foundation of China under Grant No. 61402341.

REFERENCES

[1] Z. Ji, Z. Lu, H. Li. 2014. An information retrieval approach to short text conversation. Eprint Arxiv

[2] L. Shang, T. Sakai, Z. Lu, H. Li, et al. Overview of the NTCIR-13 Short Text Conversation Task. Proceedings of NTCIR-13, 2017.

[3] L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. In Proceedings of ACL 2015, 2015:1577–1586.

[4] J. Yin, X. Jiang, Z. Lu, L. Shang, H. Li. And X. Li. Neural Generative Question Answering. International Joint Conference on Artificial Intelligence (Vol.27, pp.2972-2978). AAAI Press.

[5] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.

[6] https://github.com/NLPchina/ansj_seg

[7] http://hanlp.linrunsoft.com/

[8] H. Wang, Z. Lu, H. Li, and E. Chen. A dataset for research on short-text conversations. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013:935–945.