UB at the NTCIR-13 STC-2 Task: Exploring Syntactic Similarities and Sentiments

Jianqiang Wang
Department of Library and Information Studies
University at Buffalo, the State University of New York
Buffalo, NY 14260, U.S.A.
jw254@buffalo.edu

ABSTRACT

The University at Buffalo (UB) team participated in the STC-2 Chinese task at the NTCIR-13, working on the retrieval-based subtask. We investigated the use of manually crafted rules for improving resulted returned by an Okapi BM25 IR system. Comments that are too syntactically similar to the query post are first excluded from the result set. We then raised the ranks of those comments that contain positively sentimental/opinionated words if the test post also contains any positively sentimental/opinionated word. We also tried a method of first retrieving posts from the collection for a new post and then extracting comments that corresponded to these posts. Finally, we tested the effectiveness of combining the ranked lists from these runs. The official evaluation results show that while our baseline IR approach is effective, the usefulness of other techniques that we tried is limited. Future research directions are discussed.

Team Name

UB

Subtasks

STC-2 Retrieval-Based Method (Chinese)

Keywords

NTCIR-13, Short Text Conversation, Sentiment Analysis, Opinion Analysis, Re-Ranking

1. INTRODUCTION

Natural language conversational systems use computer programs or agents to converse with humans in a coherent manner. Such systems can be particularly useful in many application areas, including business, education, healthcare, government, and entertainment where fast and accurate responses to a potentially large number of user inquiries are much desired. The user input to such a system and the output generated by the system can be text, speech, graphics, haptics, etc. Apple's Siri and Facebook's Messenger Bots are just two of the many familiar examples of natural language conversational systems. Such human-computer conversational systems often consist of three key components, namely natural language understanding, dialogue management, and natural language generation [8]. The development of such systems often requires techniques of natural language processing and understanding, machine learning, reasoning, dialog modeling, information extraction, knowledge base development, and also automatic speech recognition and text-to-speech synthesis in the cases of speech-based systems. While much success has been achieved with task-oriented dialogue systems in constrained domains, there remain major problems with open domain dialogue systems, largely because of factors including variations and unpredictability of the user input to such systems [7].

A line of interesting research in this area focuses on social media interaction [4]. In recent years, social media websites like Twitter and the Chinese Sina WeiBo have become a major platform for online users to connect to each other with shared interests. Short burst messages are used for a variety of purposes, including broadcasting news of current events, releasing product information, marketing for business, promoting education, and sharing personal interests, opinions, and activities; users often respond/follow with messages in similar lengths. These websites have expanded significantly in terms of the number of participating users, the scope of subjects, and the data being generated. For example, Sina WeiBo as one of the largest microblogging websites in China has reached 132 million active daily users in 2016, which generates more than 100 million message every day. On average each of the 18 most popular subject areas is visited more than 10 billion times in a month. The most popular hash tags of user interest include joke, film and television, media, beauty, shopping, music, fashion, food, and the Internet [1].

The research, coined as "Short Text Conversation," aims at developing and studying computer systems that retrieve previous messages or automatically generate new messages (known as *comments*) in response to a user's messages (known as *posts*). This task is on *short text* because it focuses on application areas like Twitter and WeiBo where posts and comments are usually very short. The NTCIR's Short Text Conversation (STC) task was developed in this context. Rather than full-fledged conversational systems, however, the NTCIR STC task focuses on one-round conversation between a user and a computer system. That is, the user submits a post and the system responds with an appropriate comment, which is either retrieved from accumulated data of post-comment pairs or newly generated by the system.

The University at Buffalo (UB) team is a first-time participant to the NTCIR's STC task. We chose to work on the Chinese subtask using the retrieval-based method. We investigated the usefulness of manually crafted rules for improving resulted returned by an Okapi BM24 IR system. Comments that are too syntactically similar to the query

[1] http://data.weibo.com/report/reportDetail?id=346

post are first excluded from the result set. We then raised the ranks of those comments that contain some positively sentimental/opinionated word if the test post also contains any positively sentimental/opinionated word. We also tried a method of first retrieving posts from the collection for a new post and then extracting comments that corresponded to these post, which is based on the post-comment pair information contained in the official data set. Finally, we tested the effectiveness of combining the ranked lists from these runs. The official evaluation results show that while our baseline IR approach is effective, the usefulness of other techniques that we tried is limited.

The rest of this paper is organized as follows. In section 2 we describe briefly the STC-2 task setup including the test collection and evaluation measures. The techniques that we developed and used for the generation of our officially submitted runs are introduced in Section 3. We describe our experiment setup, including document/query processing, IR system, and the five official runs in Section 4. Section 5 presents the official evaluation results of our runs and analyzes the results. We conclude the paper with Section 6 by identifying the limitations of this study and highlighting several research directions that we are interested in exploring in the future.

2. TASK DESCRIPTION

In this section, we give a brief description of the dataset, the official evaluation measures, and the requirements of officially submitted runs. More details can be found in the task overview paper [5].

2.1 Dataset

The Chinese subtask of the STC-2 task at the NTCIR-13 reuses the document collection of STC-1 at the NTCIR-12, which was gathered and prepared by the task organizers from Sina Weibo. The collection contains 4,433,949 Weibo post-comment pairs. Posts and comments are written in Chinese although sometimes foreign words (mostly English) may show up. Since a post may have multiple responding comments and occasionally a comment may appear to respond to different posts, there are in total 219,174 *unique* posts and 4,305,706 *unique* comments in the collection. Each post is assigned with a unique Post ID and so is each comment with a unique Comment ID. For training purpose, 11,535 labeled post-comment pairs wre provided, which consist of 769 posts and 11,535 comments. Each label indicates the relevance level of a comment with regard to a post, taking one of the three possible values: L2, L1, and L0 (explained below). For the official evaluation, 100 new posts are provided. The task is for each of these 100 posts, a ranked list of comments are either retrieved (for retrieval-based method) or generated (for generation-based method). In the rest of this paper, we focus our discussion on retrieval-based method since that is the task that our team chose to work on. Each team can submit up to five runs and for each post in a run, no more than 10 comments should be included.

2.2 Relevance Levels

The relevance or more precisely, the *appropriateness*, of a comment to a given post is officially defined as having one of the three possible levels, namely *L0*, *L1*, and *L2*, which are roughly interpreted as being inappropriate, neutral (partially appropriate), and appropriate, respectively. Since the term "relevance" is more widely used in the IR field, throughout this paper we use it interchangeably with the term "appropriateness." However, readers should be aware that being topically relevant does not always guarantee being appropriate, as discussed and illustrated below.

To judge the level of appropriateness of a comment for a post, four criteria are considered:

1. *Fluent*: the comment is acceptable as a natural language text;

2. *Coherent*: the comment should be logically connected and topically relevant to the original post (i.e. the comment makes sense in the eye of the originator of the post);

3. *Self-sufficient*: the assessor can judge that the comment is appropriate by reading nothing other than the post-comment pair;

4. *Substantial*: the comment provides new information in the eye of the originator of the post;

We randomly checked a few hundred comments in the collection and found the majority of them are fluent. This is not surprising because these comments are all made by humans (although occasionally some comments appear to be consisting of random characters or symbols). In other words, being fluent does not seem to be a challenging criterion for a comment to meet. The second criterion – being coherent – can be roughly viewed as being relevant in the traditional IR evaluation practice. The other two criteria, however, could be beyond the scope of typical IR endeavor. For example, a comment that basically repeats the post would be viewed as relevant but usually not substantial, and hence at the best it could only have L1 appropriateness (being neutral).

It should be noted that these criteria are somewhat different from those used in STC-1 task, which used the criteria of *Coherent, Topically Relevant, Context-Independent, and Non-Repetitive*. It appears that "Coherent" and "Logically Relevant" criteria used in STC-1 are combined into in STC-2's "Coherent" while a new criteria "Fluent" is added; the other two criteria are basically the same (i.e., STC-1's "Context-Independent" as STC-2's "Self-Sufficient" and STC-1's "Non-Repetitive" as STC-2' "Substantial"). Because of this great similarity, the labeled data from STC-1 evaluation are still valuable for training purpose in this year's task.

Table 1 describes how the three levels of appropriateness of a comment to a post are decided. If a comment is fluent, coherent, self-sufficient, and substantial, then it should have an appropriateness level of L2. Otherwise, it can only have either L1 or L0 appropriateness, the difference being whether the comment is both fluent and coherence or not. Obviously it is more efficient for an assessor to judge the fluency and coherence first before the other two criteria.

In computing the evaluation measures (described below), the appropriateness levels of L0, L1, and L2 are assigned a numerical value of 0, 1, and 2, respectively.

Here some specific examples to explain these three levels of appropriateness:

- **Post**: *Oh, I am so hungry.*

Fluent	N	-	Y	Y	Y
Coherent	-	N	Y	Y	Y
Self-Sufficient	-	-	N	-	Y
Substantial	-	-	-	N	Y
Relevance	L0	L0	L1	L1	L2

Table 1: Relevance (appropriateness) decision based on the four criteria. "N" means the criterion is not satisfied; "Y" means the criterion is satisfied; "-" means the criterion is indifferent, i.e., it does not matter whether this criterion is satisfied or not.

- **Comment 1**: *xxxyyy...* This comment is not fluent, so it has L0 relevance.
- **Comment 2**: *You are my boss.* This comment is fluent but not coherent because it does not relate to the original post. Therefore, it has L0 relevance.
- **Comment 3**: *yes you are hungry.* This comment is fluent, coherent, and self-sufficient but not substantial because it does not provide new information. Therefore, it has only L1 relevance.
- **Comment 4**: *Maybe it's time for my boss to eat something.* This comment is fluent, coherent, and substantial but not self-sufficient because even if the original poster is the commenter's boss, the reader cannot know that fact solely based on this post-comment pair. Therefore, the comment has only L1 relevance.
- **Comment 5**: *Would you like some cookies?* This comment is fluent, coherent, self-sufficient, and substantial so it has L2 relevance.

A natural conversation system, as is what the STC task focuses on, is different from a traditional IR system in that the task here is to find an appropriate comment as quickly as possible – ideally it is the first comment included in a ranked list. Accordingly, evaluation measures should value more to the relevance of the top ranked comments. Three measures, including their variants based on the unanimity-aware gain approach, are proposed and used in the official evaluation of the STC task. Detailed definitions and explanations of these measures can be found in the STC-2 overview paper [5].

3. TECHNIQUES

Many factors can influence the appropriateness/relevance of a comment to a new post, as our discussion of the criteria above shows. How to model these factors and how to combine them to achieve optimal retrieval results are the focus of our research on the problem of short text conversation. Our approach is to first use an IR system to retrieve a ranked list of comments and then re-rank them by applying various syntactic and semantic rules that we developed based on our analysis of the labeled training data. For our officially submitted runs, we tried mainly the following three techniques.

3.1 Syntactic Similarity Analysis

Through browsing the test collection and analyzing the labeled training data, we found sometimes a comment merely repeats the original post, with little or no syntactic variation. For example:

- **Post**: *I don't like fast food because it is not healthy.*
- **Comment 1**: *I don't like fast food since it is no good to health.*

Clearly this comment, while topically relevant to the post (and likely retrieved by an IR system), does not contain any new information; instead, it merely repeats what the post says, with a slight syntactic variation. Tools do exist to compute the syntactic similarity between two text strings, e.g., using the edit distance, although they may or may not be applicable for the task here because they may help in some cases but hurt in other cases. For example, the comment below to the same post above would be appropriate because it reads as a logistical and coherent response to the post and contains new information. However, a rule based on merely syntactic similarity in this case will likely exclude the comment from the retrieved set of comments.

- **Post**: *I don't like fast food because it is not healthy.*
- **Comment 2**: *I don't like fast food made of frozen meat because it is not healthy.*

For this reason, we developed and applied a simple rule that will exclude any comment that is a substring of the original post.

3.2 Sentiment/Opinion Analysis

Lots of comments express sentiments or opinions toward the original posts, such as agreeing, liking, praising, sympathizing, hoping, and confirming or disagreeing and contradicting. We found through our analysis of the training data that if a post is written in a positive tone, comments that are also in a positive tone tend to be more appropriate in most cases. On the other hand, a negative tone of a comment does not seem to be related much to its appropriateness to a post. In this study, we model the positive tone in a message as whether it contains any positively sentimental or opinionated word. Based on these observations, we improve the ranks of positive comments in a ranked list that is initially returned by the IR system *if* the post also contains any positively sentimental or opinionated word.

We compiled a list of 3,351 positive words (in Chinese) based on words we gathered from several places online as well as our additions and deletions. In our experiments that we describe later, we checked if each comment in a ranked list contains any word from this list to decide if the comment should be re-ranked, again if the post also contains any word in this nature.

3.3 Exploring Post-Comment Relationship

As described earlier, the test collection used for the STC task consists of about 4.5 million post-comment pairs. Although it is not always the case that the comment in each pair is appropriate for the post, many such comments are indeed appropriate to the post. Therefore, a different approach to the task here is, for a new post, to first retrieve the most relevant(old) posts from the collection and then for each of these posts, to extract comments that responded to it.

3.4 Combining Evidence for Re-Ranking

Different techniques may find different relevant comments. Therefore, combining the results based on these techniques can potentially improve the retrieval effectiveness. In this study, we generated runs of applying the syntactic similarity rules, the sentiment/opinion rules, and the post-comment relationship rules, respectively and also runs of combining these rules. It should be noted that combining evidence gained from these techniques was operated at a post-retrieval stage, i.e., the initial ranked lists of retrieved comments were re-ranked by applying the rules defined by the above techniques. See the section below for more detailed descriptions.

4. EXPERIMENTS

In this section, we describe our implementation of the techniques described above, document processing and indexing, query formulation, and the IR system used in our study. Our methods involve: (1) creating separate full-text indices of comments and posts, (2) generating ranked lists of comments or posts for a new post, and (3) applying these techniques as described in previous section to improve the initial ranked lists.

4.1 Query/Document Processing

Since we chose to use word-based IR, we had to segment all texts in this experiment into words. Rather than writing our own, we used the Stanford NLP group's Chinese segmenter [2]. Manual inspection of a sample of posts and comments produced by this segmenter shows the result to be satisfactory. After both the document collection and the test post set were segmented, we converted them in hexadecimal codes for easy handling by the IR system. Finally we created two independent indices: one containing all unique posts and the other containing all unique comments. It should be noted that we did not perform the removal of stopwords, punctuation, special characters and symbols, and non-Chinese words (mostly English but occasionally also Japanese) contained in the test collection.

All our experiments were run using the Perl Search Engine (PSE), a document retrieval system based on Okapi BM25 weights. Previous IR experiments using PSE showed reasonable retrieval effectiveness [6]. In the Okapi BM25 formula [3], we used $k_1 = 1.2$, $b = 0.75$, and $k_3 = 7$ as has been commonly used. We did not run any experiments of tuning these parameters.

4.2 Official Runs

We submitted five runs for the official evaluation, as described below.

4.2.1 UB-C-R1

This was a baseline run. 20 comments were retrieved using the comment index for each of the 100 posts although only 10 of these comments were included in the officially submitted file. For the ease of discussion, let's say the ranked list with top 20 comments is "base1.rlist." For this baseline run, we simply took the top 10 comments for each post.

4.2.2 UB-C-R2

This run was the result of applying the following rules on base1.rlist (in this specific order) – it takes into consideration the syntactic similarity between a retrieved comment and the search post, as well as positive sentiments or opinions if amy:

1. Remove each retrieved comment that is a substring of the query post;

2. If the post contains any word in the positive sentimen/opinion word list, divide the 20 initially retrieved comments into two sets: (1) Set_Pos_A, each comment in which contains at least one word in the positive sentiment/opinion word list, and (2) Set_Pos_B, each comment in which does not contain any such words;

3. Re-rank the 20 comments so that the comments in Set_Pos_A appear before the comments in Set_Pos_B while retaining the relative ranks within each set as they are initially retrieved. Ranking from 1 to 20 is then reassigned. Finally only the top 10 comments are kept the official submission.

Here is an example further illustrating the process:

1. The IR system returns the following ranked list: c_1, c_2, ..., c_i, ..., c_{20}, where c_i represents a comment that is ranked number i in the ranked list.

2. Assume that c_1 and c_2 are substrings of the test post, so they are eliminated from further consideration;

3. After applying the rule of positive sentiment/opinion words, let's assume we end up with: Set_Pos_A: c_3, c_4, c_7, c_9, c_{15}, c_{20}, and Set_Pos_B: c_5, c_6, c_8, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}, c_{16}, c_{17}, c_{18}, c_{19}

4. The final UB-C-R2 would be: c_3, c_4, c_7, c_9, c_{15}, c_{20}, c_5, c_6, c_8, c_{10} (ranked from 1 to 10).

4.2.3 UB-C-R3

This run was created by taking into consideration the post-comment pair relationship and the syntactical similarity between the search post and the retrieved comments (but not sentiments or opinions), as follows:

1. Search the index of the original *posts*, which returned 20 posts for each test post. Let's say the resulting ranked list is base2.rlist;

2. Find all responding comments for each post in base2.rlist. Notice that a post could have multiple responding comments in the collection and they are not ranked. Let's say the final set of these comments is Set_P;

3. Traverse the initial ranked list of comments (i.e., base1.rlis) to divide the comments in it into two sets: (1) Set_Pst_A, each comment in which also appears in Set_P, and (2) Set_Pst_B, each comment in which does not appear in Set_P;

4. Re-rank base1.rlist so that those comments in Set_Pst_A appear before those in Set_Pst_B. Finally, take the top 10 comments in this re-ranked list as the final result of this run.

Here is an example further illustrating the process:

1. The base1.rlist is the same as above and c_1 and c_2 are substrings of the test post, so they are eliminated from further consideration;

[2] The Stanford Chinese Segmenter is publicly available at: https://nlp.stanford.edu/software/segmenter.html#Download

2. Let's assume Set_P is: c_3, c_4, c_5, c_7, c_8, c_{23}, c_{25}, c_{120} (note: these are not ranked). Again, these comments are obtained by first searching the index of all posts and then collecting for each retrieved post, all comments in the collection that responded to it (i.e., based on the post-comment pair relationship);

3. Then Set_Pst_A would be: c_3, c_4, c_5, c_7, c_8 and Set_Pst_B would be: c_6, c_9, ..., c_{20} (15 comments)

4. So the final UB-C-R3 would be: c_3, c_4, c_5, c_7, c_8, c_6, c_9, c_{10}, c_{11}, c_{12}.

4.2.4 UB-C-R4

The remaining two official runs take into consideration all three factors, i.e., the syntactical similarity between the search post and the retrieved comments, sentiments or opinions of them, and the existing post-comment pair relationship. In a sense, they are each the result of combining UB-C-R2 and UB-C-R3. The difference between these two runs lies in whether the initially retrieved comments are re-ranked first based on sentiment/opinion information and then re-ranked within either group based on the post-comment pair relationship, which is how UB-C-R4 is created, or the oppositive way, which is how UB-C-R5 is created.

Take the above examples, we now have:

1. Set_Pos_A (retrieved comments containing positive sentiment/opinion words): c_3, c_4, c_7, c_9, c_{15}, c_{20};

2. Set_Pos_B (retrieved comments not containing positive sentiment/opinion words): c_5, c_6, c_8, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}, c_{16}, c_{17}, c_{18}, c_{19};

3. Set_Pst_A (retrieved comments also appearing in comment-post pairs of retrieved posts): c_3, c_4, c_5, c_7, c_8;

4. _Pst_B (retrieved comments also appearing in comment-post pairs of retrieved posts): c_6, c_9, c_{10}, ..., c_{20} (15 comments);

The resulting ranked list of UB-C-R4 run would be: c_3, c_4, c_7, c_9, c_{15}, c_{20}, c_5, c_8, c_6, c_{10}.

4.2.5 UB-C-R5

Still with the above example of the four sets of comments, the resulting ranked list of UB-C-R5 run would be: c_3, c_4, c_7, c_5, c_8, c_6, c_{10}, c_{11}, c_{12}, c_{13}.

5. EVALUATION RESULTS

	UB-C-R1	UB-C-R2	UB-C-R3	UB-C-R4	UB-C-R5
MnG@1	0.4103	0.406	0.3792	0.3978	0.3858
MP+	0.5104	0.5105	0.498	0.5106	0.4932
MnERR@10	0.5445	0.5484	0.5314	0.5473	0.5334
MUnG@1	0.409	0.4075	0.384	0.4041	0.3904
MUP+	0.505	0.5065	0.4952	0.506	0.4911
MUnERR@10	0.5498	0.5553	0.5403	0.5567	0.5424

Table 2: Official evaluation results.

Table 2 shows the official evaluation results of our five submitted runs in terms of $nG@1$, $P+$, and $nERR@10$ and the same measures that were computed following the unanimity-aware gain approach. Overall these results are comparable with each other – there is no statistical difference between any pair of the results, verified by running Student's t-tests. Therefore, the techniques that we explored did not seem to have any noticeable improving effect.

Comparison of Runs	Tells us
UB-C-R2 vs UB-C-R1	The effect of using syntactical similarity and sentiments/opinions
UB-C-R3 vs UB-C-R1	The effect of using post-comment pair relationship
UB-C-R4 vs UB-C-R1	The effect of using syntactical similarity, sentiments/opinions and post-comment pair relationship
UB-C-R4 vs UB-C-R5	The better combination of using sentiments/opinions and post-comment pair relationship

Table 3: The rationale of comparing runs.

We notice that UB-C-R1, UB-C-R2, and UB-C-R4 are consistently slightly better than UB-C-R3 and UB-C-R5 based on the average of each of the six measures reported in the official results (see Table 2). As a reminder UB-C-R3 and UB-C-R5 somehow gave more weight to retrieved comments that appear in comment-post pairs of retrieved posts. In our preliminary analysis, we found that most of the top 20 retrieved posts were actually not so appropriate to the query post and hence it actually hurt to raise the ranks of their corresponding comments. This perhaps explains why these two runs are somehow inferior to the other three runs.

We further conducted post-by-post comparisons of these runs to see on which topics our techniques helped, did not have any effect, or hurt. Table 3 describes what we are looking at by performing each pair-wise post-by-post comparison of two runs. Notice that among the six measures used in the official evaluation, $nG@1$ and $UnG@1$ test the effectiveness of the system at top-1 rank. We compared UB-C-R1 run and UB-C-R2 run in terms of $nG@1$ and found for only nine posts a difference is observed. More specifically, our techniques had a positive effect on five of these nine posts while a negative effect on the rest four posts. By further checking and comparing the top-1 comment contained in UB-C-R1 and that contained in UB-C-R2 for each post, we found that for the five posts (they are posts 10790, 10110, 10640, 10010, and 10330) that our techniques helped, all posts have a positive tone and all top-ranked comments in UB-C-R2 also have a positive tone.

On the other hand, the top-1 comment in UB-C-R2 for Post 10040 contains the word " spirit," which could be either a noun with a neutral sense (which is the meaning in this comment) or an adjective in the sense of "energetic." Since our techniques did not apply any sort of word sense disambiguation or part-of-speech tagging, the comment was mistakenly boosted based on this word as being positive while it is not. Post 10090, which is another post that our techniques had a negative effect on, presented another interesting case. The post talks about "just finished the New Year's Eve dinner and feel very full" while the UB-C-R2 top-1 comment responded cynically with "great and happy doomsday!" Here again, the comment was regarded as being sentimentally positive due to the words "great" and "happy" while the comment as a whole does not have a positive tone.

Next, we conducted post-by-post pair-wise comparisons of runs in terms of $nERR@10$ measure, which tests the system's effectiveness on top 10 retrieved comments. Figure 1, 2, and 3 each shows the comparison of two runs, in which we only display those posts for which the absolute difference of the $nERR@10$ values between the two runs is 0.2 or larger. In other words, we deem our techniques had little or no effect if the difference in $nERR@10$ is less than 0.2.

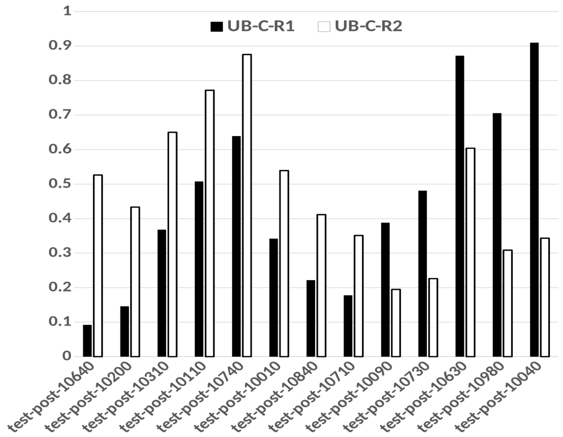

Figure 1: Comparison of UB-C-R2 and UB-C-R1 on $nERR@10$. The figure shows only those posts whose absolute difference of $nERR@10$ between the two runs is at least 0.2.

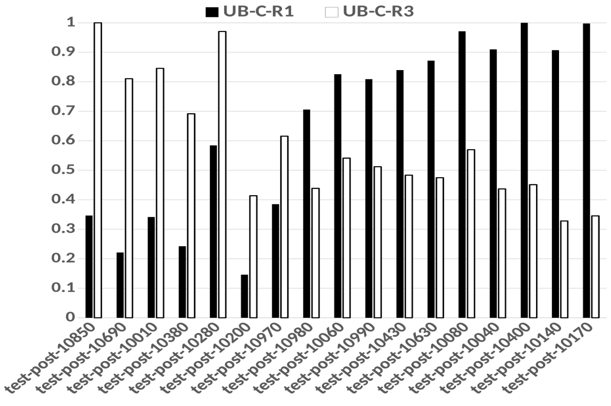

Figure 2: Comparison of UB-C-R3 and UB-C-R1 on $nERR@10$. The figure shows only those posts whose absolute difference of $nERR@10$ between the two runs is at least 0.2.

From these comparisons, we have the following observations. First, our techniques had an effect (positive or negative) on about 10-20% of the test posts, which is not a negligible number. Secondly, our techniques helped improve the retrieval of appropriate comments for some posts while hurt some other topics. In the cases of improved retrieval effectiveness, we did see the techniques helped in the way we expected them to. In the cases of degraded effectiveness, the causes of the problem could be due to (1) incorrect detection of a comment's sentiment or opinion (just like in the case of post 10090), or (2) drift of the topic as a result of utilizing existing post-comment pair relationships.

Comparing Figure 1 and Figure 2 reveals that among the seven posts that our techniques had a positive effect in UB-C-R3, only two of them also appear in the eight posts that saw improved effectiveness in UB-C-R2. This indicates that exploring the existing post-comment pair relationships did contribute differently from utilizing sentiment/opinion information. Not surprisingly, combining these techniques (which is what UB-C-R4 is about) could lead to more posts whose $nERR@10$ was improved.

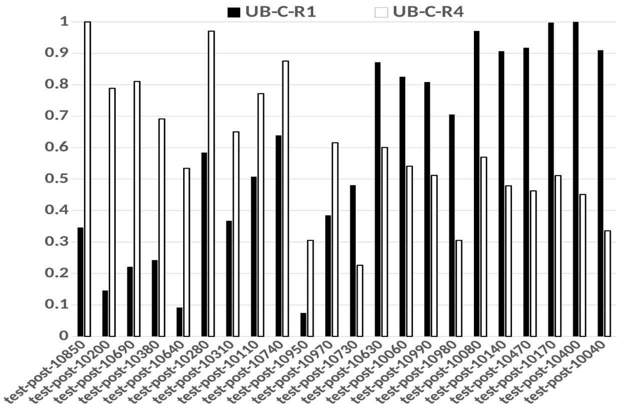

Figure 3: Comparison of UB-C-R4 and UB-C-R1 on $nERR@10$. The figure shows only those posts whose absolute difference of $nERR@10$ between the two runs is at least 0.2.

Additionally, as compared to UB-C-R5 post-by-post, we found that UB-C-R4 had 16 posts with higher $nERR@10$ and four posts with lower $nERR@10$ although only three posts were at least 0.2 higher while one post at least 0.2 lower. This somehow indicates that when combining evidence learned from sentiments/opinions and post-comment pairs, the former should be given more weight.

Finally, we also compared these runs based on other official measures. The relatively performance is quite consistent as what has been discussed above; the sets of posts for which our techniques helped or hurt are also similar between these measures. Therefore, we are not including further discussion of comparisons based on these other measures.

6. CONCLUSIONS AND FUTURE WORK

For our participation in the STC-2 Chinese task, we used the retrieval-based method. To further enhance it, we designed and experimented with several techniques, namely, the use of syntactic similarities between each retrieved comment and the query post, the use of positively sentimental or opinionated heuristics, and the consideration for the post-comment pair relationship that are included in the test collection. While our experiments show that the retrieval-based method using Okapi BM25 weighting is effective, the techniques that we have tried have overall quite limited effect on the task. Our post-by-post comparisons of the submitted runs indicate that these techniques were effective on some posts while degraded the performance on some other topics.

There are several limitations of the study reported in this paper. First, all these techniques were applied on the top 20 comments initially returned by the IR system and eventually only 10 of them (after re-ranking) were included. It would be interesting to see how these techniques perform if the initial ranked list is expanded to a larger number because it is possible that some appropriate comments are ranked below top 20 and our techniques could bring them up to top 10. Second, even though our initial examination of the training data revealed that positive sentiments or opinions tend to have more effect than negative ones, it would be beneficial to take both into consideration in re-ranking comments. Our discussion of Post 10090 in an earlier paragraph in Section 5, of which the top ranked comment contains both a positively sentimental word and a negatively sentimental word, seems to corroborate this idea. Thirdly, our approach reported in this study consists of two steps, namely an initial step of applying Okapi BM25 weight to retrieve potentially appropriate comments and a follow-up step of re-ranking based on those techniques. A different approach would be to combine these two steps in one-pass of term weighting and document ranking. The benefit of this alternative approach is to overcome at least some of the problems of retrieval-based method. For example, some relevant comments may be ranked very low in the two-step approach and hence excluded from further consideration while in the one-pass approach their relevance score could potentially be boosted. Finally, the IR system that we used in this study focuses strictly on topical relevance and basically ignores other aspects of a comment's *appropriateness*. If we can model *fluency*, *self-sufficiency*, and *substantiality* prior to or during the retrieval process, it could potentially improve the performance. For example, by applying some basic linguistic rules it is possible that comments that are not fluent can be eliminated from the retrieval stage.

Short text conversation is a challenging task in that IR-based techniques, which are usually built upon strict term matching, fail to return appropriate comments that contain no keywords that appear in the query post. This is actually a fundamental problem in the IR field although with longer documents, this problem can be often lessened to a great extent. With short messages like the Sina Weibo posts and comments, however, the problem of word mismatch could be more severe. Unless some kind of reliable knowledge bases are used, we suspect that IR-based approaches to STC can only have limited success. For example, a comment talking about taxation officers could very well be appropriate to a post mentioning public servants, but unless the system knows taxation officers are one type of public servants, such a comment will most likely not be retrieved. For this reason, we plan to explore the usefulness of thesauri or synonyms for the STC task in our future work.

Short text also provides unique opportunities in that the knowledge structure contained in each message is often simple, as compared to longer documents. Therefore, it is possible to extract the knowledge structure form both the post and the comment and then judge the appropriateness of the latter to the former based on a comparison of their knowledge structures. It would be interesting to see how similarities of such knowledge structures can contribute to improving the STC task performance.

In a long run, other features of posts and comments, such as the reputation of posters and commenters and the subject areas where they post more frequently, which are not included in the current test collection but usually an integral part of social media, can be valuable sources of information. This is somewhat similar to the ideas behind Google's PageRank algorithm [1] and citation analysis [2]. These are also some of the areas that we hope to explore in the future.

7. ACKNOWLEDGEMENT

The author would like to thank Dagobert Soergel and Ying Sun at the Department of Library and Information Studies, University at Buffalo for their discussions of some of the ideas reported in this paper. Our thanks also extend to the NTCIR STC organizers for their timely responses to our inquiries about the task as well as their valuable comments on an earlier draft of the paper.

8. REFERENCES

[1] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, pages 107–117. Elsevier Science Publishers B. V., 1998.

[2] E. Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178(4060):471–479, 1972.

[3] S. E. Robertson and K. Sparck-Jones. Simple proven approaches to text retrieval. Cambridge University Computer Laboratory, 1997.

[4] L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1577–1586, 2015.

[5] L. Shang, T. Sakai, H. Li, R. Higashinaka, Y. Miyao, Y. Arase, and M. Nomoto. Overview of the NTCIR-13 short text conversation task. In *Proceedings of NTCIR-13*, 2017.

[6] J. Wang and D. W. Oard. Combining bidirectional translation and synonymy for cross-language information retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 202–209. ACM Press, 2006.

[7] N. G. Ward and D. DeVault. Ten challenges in highly-interactive dialog systems. In *AAAI 2015 Spring Symposium on Turn-taking and Coordination in Human-Machine Interaction*, 2015.

[8] S. Young, M. Gasic, B. Thomson, and J. D. Williams. POMDP-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, 2013.