University of Indonesia at TREC 2011 Microblog Track

Samuel Louvan, Mochamad Ibrahim, Mirna Adriani, Clara Vania, Bayu Distiawan, and Metti Z. Wanagiri

Faculty of Computer Science
University of Indonesia
Depok Campus, Depok 16424, Indonesia

{samuel.louvan, mochamad.ibrahim, mirna, c.vania, b.distiawan, mz.wanagiri} @cs.ui.ac.id

ABSTRACT

In this paper we describe our submission to the TREC2011 MicroblogTrack. Our run combines different methods namely customized scoring function, query reformulation, and query expansion. We apply query expansion from dataset with different weighting scheme. Furthermore, we do an initial experiment to incorporate timestamp of the *tweet* document in order to improve search performance. We found the query expansion utilizing external search result combined with re-tweet value in the customized scoring function was the most effective.

1. INTRODUCTION

The Microblog track[1] is a brand new track which replaces the previous blog track in TREC. One of the reasons to have this new track is the emerging popularity of Twitter[2]. Twitter is a microblogging environment which allows users to post certain text, limited up to 140 characters, called a *tweet*. Within Twitter, users can subscribe to other user's tweet by *following* them. In addition, users can post reply to a tweet, send private messages and also perform a *re-tweet* when they find an interesting tweet. Given the myriad size of Twitter data and intensive information flow, Twitter has drawn a lot of attention from researchers in various fields. Many research works study the data on Twitter and try to extract knowledge from it. For example, by utilizing the social network nature of Twitter we may identify who is influential in certain topic [5]. Specific from Information Retrieval discipline, the microblogging search is challenging as its characteristics are different from ordinary document retrieval e.g., the size of the document (tweet) is much shorter, informal/abbreviated words often occur in the document. The tweet is considered relevant if it is current in terms of time.

Massoudi et.al. apply probabilistic framework combined with quality indicators such as re-tweet value [6] while Nagmoti et.al. use Twitter's network feature such as follower number in their microblog ranking mechanism [2].

This year University of Indonesia participate in the TREC 2011 Microblog track for the first time.

In this microblog track, we investigate various mechanisms to reformulate our query in order to retrieve more relevant documents. Approaches such as proximity search, keyword weighting, and phrase detection are explored. We also apply query expansion technique by adding terms using pseudo-relevance feedback method. In order to perform re-scoring of the retrieved, we implement additional scoring function in our retrieval framework.

This paper is structured as follows; we first begin describe related works and the objective of the TREC Microblog track. We then outline our approaches in the submitted runs and present and discuss the results. The conclusion summarizes our findings and possible future works.

2. MICROBLOG TRACK

2.1 Tasks

The task of the Microblog Track is to retrieve relevant *tweet* documents for each query at a specific time. Therefore, for each query the information retrieval system should retrieve all relevant tweet documents that are ordered from the newest to the oldest, and all tweets must be created before the query is issued.

2.2 Dataset

The size of the corpus is approximately 16 million tweets, which span over a two-week period (24 January 2011 – 8 February 2011). However, we can only collect around 14 million tweets because apparently many of the tweet users set their Twitter accounts into *protected mode* or change their account names so that we are not able to download their tweets. The corpus was downloaded using a twitter-academia-corpus downloader.

[1] Https://sites.google.com/site/microblogtrack/

[2] Https://www.twitter.com

2.3 Evaluation

The relevant tweets is judged based on the specific information need and also the "interestingness" aspect. All non-English is judged as not relevant tweets. Plain re-tweeted documents, which do not add any informative content is also considered as not relevant. The evaluation measure used for the task is P@30.

3. RETRIEVAL APPROACH

We conduct our experiment using an open source information retrieval system Apache Lucene[3] to index the documents and perform basic search functionality. The essential idea of our approach is to reformulate the topic query, perform search and then do re-scoring for the retrieved documents.

3.1 Query Expansion

We apply several weighting schemes for the query expansion namely TF-DF scheme and a combination of TF-DF with time parameter. In particular, we use TF-DF weighting since it yields more encouraging results on our preliminary study. We apply pseudo-relevance feedback on the tweet corpus and snippets from Google search results [4].

3.2 Proximity Search

We use built in proximity operator from Lucene to incorporate proximity search. Here, we use a distance value, which indicates the number of terms than can present between query terms. Fewer terms between query terms will give higher score for a document. As default value, we choose 10 words distance for the proximity operator. Proximity search will also detect every probability of term combination appears in tweet, for example "world war"~10.

3.3 Phrase Identification and POS Tagging Rule

We consider that phrase is important if it appears on the tweets. So we identify phrases based on their POSTAG. We use Stanford POS tagging [4] tools to identify the POSTAG on the query terms.

Based on our simple observations and rules from [9] , we define two set of rules to detect phrases.

The first set of rules are:

- For each NNP term, if previous term is NNP, IN, DT, or JJ, append the term with previous term.
- For each NN/NNS term, if previous term is JJ or DT, append the term with previous term. If previous term is NN or NNS and the term before previous is not JJ, append the term with previous term.
- For each JJ term, if previous term is DT append the term with previous term.
- For each IN term, if previous term is NN or NNS and term before previous is not JJ, append the term with previous term.
- For each DT term, if previous term is IN, append the term with previous term.
- For each term, if previous term is IN or DT, append the term with previous term.

The second set of rules are:

- If there is an NNP in the original topic query then we pair it up with all non NNP terms in the original topic query.
- If there is no NNP, find NNS and then we pair it up with all non NNS terms in the original topic query.
- If there is no NNP/NNS, find JJ and then pair it up with all non JJ terms in the original topic query.

3.4 Keywords Weighting

We use built in boosting operator from Lucene to increase the weight of certain term during retrieval process. In other words, we boost terms that we consider more important. We apply different boosting value for different kinds of terms. The order of the boosting value, from the highest to the lowest is as follows:

- Original query (as one phrase), boosting value : 5
- Phrase detected using first approach, boosting value : 3
- Phrase detected using second approach, boosting value : 2
- Terms (we give score for each term), boosting value : 1

All different kinds of terms above are combined using OR operator. Note that, each term in a query can be written once minimum, for example one in phrase and another in the single term.

3.5 Re-Tweet Value

We use Re-Tweet(RT) information to indicate highly discussed topics. We assume the higher the value of RT, the more important is the tweet. We obtain the RT value from the HTML page of the tweet.The HTML contains the identifier of the *original* tweet and the number which indicates how many times the *original* tweet has been retweeted. As the original tweet is considered ~~as~~ relevant, therefore we associated the RT value to it.

We simply use the RT value in our scoring function by normalizing the RT value, therefore the value range is between 0 to 1.

3.6 Language Detection

Since this track only considers English tweet as relevant tweets, then we use language detection to filter non-English tweets from the retrieved tweets.

We use common language detection, LangDetect, known to detect many languages (including English) with more than 99% accuracy.

3.7 Scoring Function

For scoring function, we make some modification on Lucene scoring function as it tends to give higher score to a document which has more query keywords in its content. For example, if we

[3] http://Lucene.apache.org

[4] http://code.google.com/apis/customsearch

have a query “fifa 2022”, the tweet “fifa 2022 video game fifa #fifa” will obtain a higher score than the tweet “fifa 2022 in qatar”. In addition, as the length of a tweet is very limited (140 characters), it is more reasonable if we only consider the unique words from a tweet. Otherwise, the problem above will occur.

Second, we consider that the relevant tweets on query are appearing on the same timescale. We cluster tweet by the time (days), and give each cluster score based on the number of relevant tweets found in that cluster. More relevant tweets gives will give higher cluster score. We assume that a tweet relevant if it presents in top retrieved results.

4. SUBMITTED RUNS

Our group submits four submissions run with some combination on query and scoring function. There are:

1. FASILKOM01
   This run does not include any future evidence e.g. tweet data after query tweet time cutoff and external resource. We only utilize query expansion from internal dataset and proximity search.
2. FASILKOM02
   This run uses phrase identification, query expansion from external resource (Google snippets) and internal dataset, customized scoring function (RT value added), proximity search, keywords weighting, and language detection.
3. FASILKOM03
   This run uses phrase query identification, query expansion from internal dataset, customized scoring function (without RT value added), proximity search, keywords weighting, and language detection.
4. FASILKOM04
   This run uses phrase identification, query proximity search, keywords weighting, and language detection. We use query expansion from Google snippets with customized scoring function (without RT value added).

5. RESULTS

The TREC 2011 Microblog results are basically divided into two sets of judgment namely the set that only consider highly relevant tweets and the set which consider all highly relevant and relevant tweets as relevant documents. The exact P@30 measurements of our runs are shown in

1 and Figure 2. From our observation, for the all relevant tweets result, our runs are above the median. However, for the highly relevant tweets, our approach suffers as the number of relevant tweets is significantly smaller and makes it harder to retrieve the right ones.

Our results show that, although FASILKOM02 is the best run for all relevant tweets, it gives the worst result for highly relevant tweets. Since in FASILKOM02 we add more terms through query expansion, consequently it retrieves more relevant tweets than the other runs. However, it is possible that adding too many terms can lower down the rank of the highly relevant tweets, hence it may yield lower precision (P@30). Given that a user can only post 140 characters for one tweet, terms used by users to post about a topic usually are not really different. Furthermore, there are so many nonstandard terms and abbreviations used in internal dataset, so that the expansion terms are not quite good compared to Google snippets.

As for FASILKOM04, it obtains higher precision on highly relevant tweets because the query expansion from external source (Google snippets) apparently can capture some keywords that appear in the highly relevant tweets. Snippets from Google usually come from news articles that use good standard language. This can help us to get richer terms for each query topic.

Figure 1. Result for All Relevant Tweets

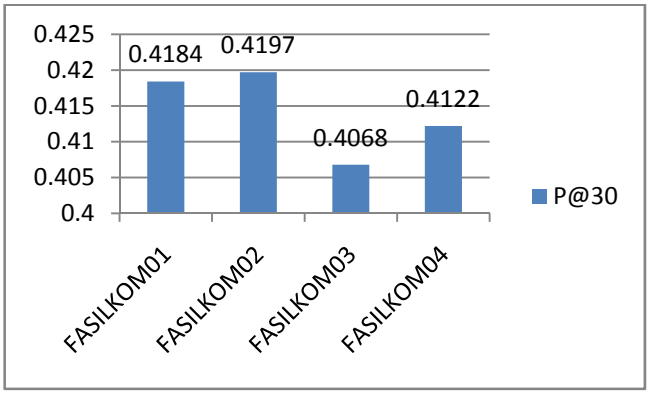
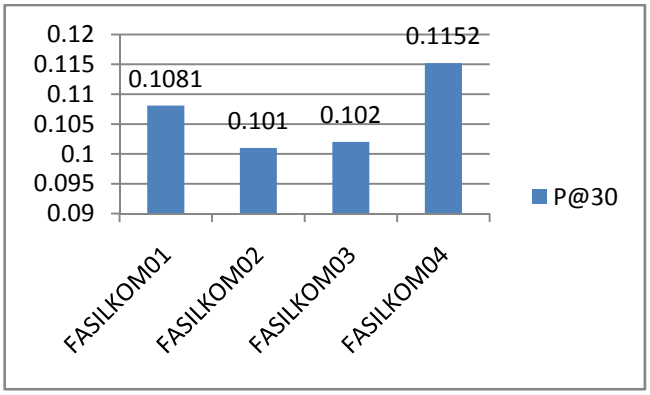


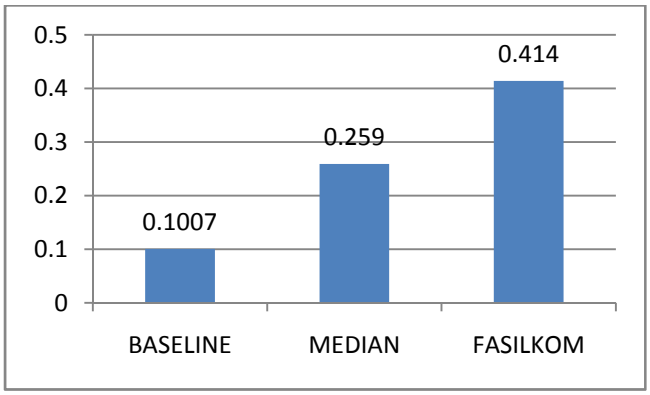
Figure 2. The Results for Highly Relevant Tweets



For this experiment, we assume that RT value can indicate the importance of a tweet. We use RT value in our customized score function for FASILKOM02 run. In reality, our second run cannot reach higher score in case of highly relevant tweets. This shows that the importance of a tweet does not only depend on RT value.

If we compare our performance to the retrieved documents as the baseline (resulted by information retrieval system) and median scores from other TREC participant, our result is higher than the average median value. This means generally our method is quite successful to retrieve relevant tweets at P@30.

Figure 3. Average P@30 Compared to Baseline and Median



6. CONCLUSION AND FURTHER WORK

In this paper we describe our approach in the TREC 2011 Microblog track. We reformulate the queries and perform re-scoring using our customized scoring function.

Our approach in FASILKOM02 which in particular uses re-tweet value in the scoring component and both internal and external information for query expansion yield the best result. However, for highly relevant tweets, FASILKOM04 which uses external information for query expansion outperforms the other runs. Based on our analysis, Google snippets can give richer terms than internal dataset (which usually contain nonstandard terms and abbreviations), so that it can help us to get more high relevant tweets. As for RT values, results show that the use of RT value is not enough to indicate that a tweet is highly relevant.

For future direction, we would like to explore the possibility to use other properties of Twitter users and also how to find a range of timestamp which is considered as the "prime time" for a given topic.

7. REFERENCES

[1] H Kwak, C Lee, H Park, and S Moon, What is Twitter, a social network or a news media. WWW 2010, pages 591-600, North Carolina, USA. ACM

[2] R Nagmoti, A Teredesai, M De Cock, Ranking Approaches for Microblog Search, WI-IAT '10 Proceedings of the 2010. ACM

[3] E Bakshy, J M.Hofman, A.Manson, and J.Watts, Everyone's an influencer: Quantifying Influence on Twitter, In Proceedings of the 4th ACM International Conference on Web Search and Web Data Mining, WSDM 2011. ACM

[4] Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In Proceedings of HLT-NAACL 2003, pp. 252-259.

[5] J Weng, E.Lim, J Jiang, Qi He, TwitterRank: Finding Topic-sensitive influential Twitterees. In Proceedings of the 3rd ACM International Conference on Web Search and Web Data Mining, WSDM 2010. ACM

[6] K. Massoudi, E. Tsagkias, M. de Rijke, and W. Weerkamp. Incorporating query expansion and quality indicators in searching microblog posts. In ECIR 2011: 33rd European Conference on Information Retrieval, pages 362–367, Dublin, 2011. Springer, Springer

[7] Pal, A., and Counts, S. 2011. Identifying topical authorities in microblogs. In Proceedings of the 4th ACM International Conference on Web Search and Web Data Mining, WSDM 2011. ACM

[8] J.Welch, U.Schonfeld, D.He, J.Cho, Topical semantics of twitter links. In Proceedings of the 4th ACM International Conference on Web Search and Web Data Mining, WSDM 2011. ACM

[9] D.Jurafsky, J.H.Martin. Speech and Language Processing. 2009.Pearson International Edition