Retrospective Tweet Summarization: Investigating Neural Approaches for Tweet Retrieval

Lila Boualili
lila.boualili@irit.fr
IRIT, University of Paul Sabatier
Toulouse, France

Lynda Said Lhadj
l_said_lhadj@esi.dz
ESI
Algiers, Algeria

Mohand Boughanem
mohand.boughanem@irit.fr
IRIT, University of Paul Sabatier
Toulouse, France

ABSTRACT

While being a valuable source of information, Twitter can be overwhelming given the volume and the velocity of the information being published. Thus, an automatically generated summary containing relevant tweets and covering the key aspects of the user query could be of great interest. However, dealing with tweets presents challenging issues such as the redundancy of information, their limited length and informal style. To address these issues, we follow a two-step approach. First, we retrieve the top-k relevant tweets with respect to the user topic. Deep neural (DN) models are mainly investigated for their ability to learn the relevance function from the raw text input. Meanwhile, the distributed representations of tweets can reduce the semantic gap between the tweets and the topic. Then, the relevant tweets are clustered according to their similarity and the representative tweet of each cluster is included in the summary. Experiments on the TREC Real-Time Summarization (RTS) task have shown that DN models are promising and can even surpass the performance of a traditional IR model (BM25).

KEYWORDS

Tweet summarization, tweet retrieval, deep neural models, relevance

1 INTRODUCTION

Twitter is becoming an undeniable source of real-time information providing in many cases the latest news, sometimes even before traditional media, especially when it comes to unpredictable events such as natural disasters. Following an ongoing event can be difficult due to the large amount of information produced with a high velocity on a wide range of topics. In 2019, 500M tweets were published every day, that is, on average, 6000 tweets every second. Providing users with automatically generated summaries about their topics of interest is an interesting solution to prevent them from being overloaded with irrelevant and redundant tweets. However, tweet summarization requires handling the particular nature of tweets that (1) does not necessarily use the same vocabulary as the user's interest topic expressed in a query, (2) tweets and queries are of limited length making the relevance estimation based on term frequency ineffective, and at last (3) are highly redundant. Several models have been proposed to tackle the tweet summarization problem [5, 13–15, 18]. Most of these models generates summaries by retrieving the most relevant tweets then discarding the redundant ones using classical IR models also called bag-of-words models. Accordingly, the retrieved tweets do not always match the search intent expressed in the query for two main reasons : The bag-of-words representation of the tweets is not sufficient to capture their semantics. Moreover, the term frequency is inefficient in the case of tweets because of their limited length (280 characters) where a term rarely appears more than ones.

To tackle these issues a two-stage approach is followed. First, we investigate Deep Neural (DN) models to retrieve relevant tweets. Their interest lies, on one hand, on their ability to learn a complex task such as ranking from raw text inputs. On the other hand, it is known that relevance in IR is vague and difficult to estimate since it is the result of a complex cognitive process. Second, as relevant tweets are generally redundant, we address this problem by clustering the retrieved tweets so that similar ones are grouped in the same cluster and only a representative tweet per cluster will be included in the summary. We focus in this paper on some existing DN models [4, 6, 9, 11] with different language representation models (Word2Vec [8], GloVe [12]) and empirically select the best performing one on the TREC Real Time Summarization (RTS) collections. The obtained results show that estimating relevance using a DN model is promising and can even surpass the performance of a classic IR model (BM25). In addition, our results on the scenario B of TREC-RTS compete with the results of the second-best run of the 2016 campaign. Our code is available for reproducing our experiments and future work .

The rest of the paper is organized as follows. In section 2, we review some related work. In section 3, we describe our two-step approach. Finally, in section 4 we present the experimental setup we use and discuss our results.

2 RELATED WORK

The dominant approach for tweet summarization consists in two steps, by first selecting a list of the top-k relevant tweets, and then discarding redundant ones. The first step relies on query-tweet relevance weighting while the second uses tweet-tweet similarity measures. Sharifi et al.[13] have proposed the HybridTF-IDF approach where the overall set of tweets is considered as a document for Term Frequency (TF) calculation in order to overcome the tweet length problem. Top-weighted tweets are iteratively included in the summary if their cosine similarity with tweets already selected is under an empirically predefined threshold . *Sumbasic*, a term-frequency based method initially proposed for document summarization, has proven to be also efficient for tweet summarization. [18] have proposed one of the first summarization approaches for monitoring the live tweet stream for scheduled events. Term

https://github.com/BOUALILILila/NeuralTweetSummarization

frequency is used to measure the relevance of a tweet w.r.t to the event and Kullback-Leibler divergence [7] to reduce redundancy.

The introduction of TREC Real-Time Summarization (RTS) evaluation campaigns has led to the development of several models. In the best run of the scenario B[15], the tweet relevance estimation is based on the query term frequency in both tweet text and webpages linked to the tweet. The tweet similarity is measured by their common vocabulary. The second best performing run [17] used a combination of the social importance of the tweet with its relevance score. The social importance is obtained with a logistic regression model on social attributes such as the number of followers while the relevance is a combination of BM25, TF-IDF and cosine similarity. In TREC RTS 2017, the best run [5] evaluates tweets relevance using a language model combining the tweet and its linked web pages content. The second best performing model [16] proposed to lineally combine several relevance scores such as cosine similarity, IDF weights and negative KL-divergence language model. These models Aiming at reducing the gap between the user intent expressed in a query and tweets, our work focuses on the tweet relevance estimation where we investigate deep neural models.

3 TWEET SUMMARIZATION APPROACH

We present in this section the IR approach we follow in the context of tweet summarization. We assume that a summary is a set of the top non-redundant tweets that are relevant to the user's interest topic. First, we attempt to reduce the semantic gap between queries and tweets using distributed representations. Then, we investigate deep neuronal models – that are capable of learning a relevance function from the text inputs without further hand crafted features– to retrieve candidate relevant tweets. This overcomes the ineffectiveness of term-frequency based models for short text like tweets. As the resulting candidate list of tweets may contain redundant information, clusters of similar tweets are created using their representations in the latent space. The summary is then constituted from the representative tweet of each cluster. We detail each step of our approach in this section.

3.1 Text representation

The contextualization of words offered by the embeddings can enrich tweet representation and contribute to reduce the semantic gap between the query and the tweets. In this work, we investigate different distributed text representation models widely adopted in Natural Language Processing in general and IR particularly, because of their efficiency, for example: Word2Vec [8], GloVe [12] and Fasttext [1]. These models use shallow neural networks to learn word representations from their contexts. Specifically, Word2Vec has two configurations: SkipGram that takes a word as input and tries to predict its context, while CBOW tries to predict the word from its context. Fasttext is a framework that learns word representations like Word2Vec but at the n-gram level to overcome the words out-of-vocabulary problem. GloVe uses word co-occurrence statistics in the corpus in order to learn the word representation from both its local and global context.

3.2 Relevance estimation

The deep neural models we have investigated are: MatchPyramid [11], DRMM [4], ARC-II [6] and DUET [9]. We give in the following a global overview of each model. **MatchPyramid [11].** Models the query-tweet matching as an image recognition process. A matching matrix representing the similarities between the query-tweet words is constructed and viewed as an image. A neural network convolutional is then used to capture rich matching patterns layer by layer. MatchPyramid can thus identify salient hierarchical signals from n-gram to n-terms.

DRMM [4]. A relevance matching DN model for ad hoc retrieval integrating query term importance and local match signals between query-tweet words. The model uses matching histograms to capture exact match signals.

ARC-II [6]. Evaluates the query-tweet matching based on an interaction matrix between their words. A convolutional network with max-pooling, that is capable of capturing and preserving the order of local features in the interaction matrix, is then used to evaluate the matching score.

DUET [9]. Relies on both lexical and semantic matching signals in order to evaluate the relevance of a tweet w.r.t to a query. It uses two DN networks, one for each signal type. The final relevance score is the sum of the of the two network scores.

At the end of this step, a list of candidate relevant tweets are ranked according to their relevance score w.r.t to the user topic (query). As a relevant information is widely shared, the top selected tweets can contain redundant information. Thus, a further step reducing the redundancy is required.

3.3 Redundancy reduction

Two tweets are considered redundant if they carry the same information. In order to discard relevant tweets, we use a clustering method to group similar tweets. To measure the similarity between tweets, we use their distributed representations since similar words have close representations in the latent space. We obtain clusters of equivalent tweets in terms of information and we select the most relevant one (having the highest relevance score) per cluster to build the summary.

4 EXPERIMENTS

In order to assess our assumptions, we conduct a set of experiments aiming at the following objectives:

- Studying the impact of language representation models on DN models for tweet retrieval.
- Comparing the effectiveness of these DN models with a traditional IR baseline.
- Investigating clustering as a redundancy reduction method.

Finally, we compare our global tweet summarization method with prior work [10, 13] and official runs of the TREC RTS 2016 and 2017 [5, 15–17].

4.1 Experimental Setup

Dataset. We used the replay mechanism of the scenario B over the tweets collected during the evaluation period of the TREC RTS 2016 and 2017 campaigns. This scenario consists in identifying up to 100

ranked tweets per day and per topic. These tweets are sent to the user daily. The TREC RTS 2016 collection consists of 203 topics, but only 56 were assessed along with 44, 566 relevance judgements. For the 2017 dataset, 97 topics were assessed with only 39, 106 relevance judgements. Each topic includes a title, a description and a narrative. In these experiments, we have used only the title having the form closely related to real user queries, ie with only the important key words.

Evaluation metrics. We use standard MAP and precision at 10 (P@10) for evaluating the tweet retrieval component. Then we use the official evaluation metrics of TREC RTS campaigns that are variants of the NDCG metric namely: nDCG0, nDCG1 and nDCGp to evaluate the overall summarization system. The difference between these metrics resides in the penalty the system receives when it sends tweets in a "silent day" for a given topic, where there is actually no relevant tweets for the topic. During a silent day, nDCG0 gives a gain of 0 to the system that sends tweets, nDCG1 rewards the system that did not push any tweet with a perfect gain of 1 else 0, while nDCGp –introduced in the 2017 campaign– gives a penalty that goes from 0 to 1 according to the number of tweets pushed by the system.

Tweet Processing. Before relevance estimation, potential irrelevant tweets are filtered. Each tweet with less than 5 tokens is considered too short to carry any information, thus it is automatically discarded. This yields to reduce the number of candidates tweets and to decrease the computational complexity. We also filter tweets that have more than one URL and more than three hashtags. Theoretically, such tweets are supposed to highlight the tweet subject. However, considering the tweet length, more than three subjects indicates low quality rather highlighting the key topic of the tweet.

Deep Neural models. In these experiments, we rely on the open source implementation made available in MatchZoo. The aim of this platform is facilitating the design, comparing and sharing deep text matching models. In addition, a five-fold cross validation is used to evaluate the DN models in all experiments in order to use all the data.

4.2 Results and discussion

4.2.1 Impact of language representation models on DN models. To evaluate the effectiveness of the continuous-word representation for our task, we used locally pre-trained embeddings and global pre-trained embeddings as input for the DN models : MatchPyramid [11], DRMM [4], ARC-II [6] and DUET [9].

For local embeddings, we have pre-trained the two configurations of Word2Vec: (a) Skip-Gram (SG) and (b) CBOW with Fasttext using 3-grams that is : (a) FastText-SG and (b) FastText-CBOW on tweets collected before the evaluation period of TREC RTS campaigns by [3]. Table 1 shows the statistics of the local collection.

For Global embeddings, we used Word2Vec pre-trained on Google News Corpus, GloVe pre-trained on tweets and GloVe pre-trained on Wikipedia. Table 2 reports the best results in term of MAP we obtained on the TREC RTS 2016 set. For our task, local embeddings seem to yield better performance overall the models, the best result has been achieved by MatchPyramid on local Fastext-CBOW embeddings. DRMM performs poorly because it uses frequency histograms that are more efficient for long documents rather than short documents like tweets.

https://github.com/NTMC-Community/MatchZoo

Table 1: Local embeddings pre-training collection.

Tweet count	word count	vocabulary size (unique words)
45,798,044	321,658,075	7,707,693

Table 2: BEST MAP percentage results on the TREC RTS 2016 dataset for each model ranked from best to worst.

Model	Embeddings	Local	MAP(%)
MatchPyramid	Fasttext-CBOW	✓	36.13
DUET	Fasttext-CBOW	✓	29.32
ARC-II	CBOW	✓	28.61
DRMM	GloVe-tweets	-	08.06

Table 3: MatchPyramid vs. BM25

Model	TREC RTS 2016		TREC RTS 2017	
	MAP	P@10	MAP	P@10
BM25	13.44	23.75	26.83	34.69
MatchPyramid	36.13	28.93	32.20	24.00

4.2.2 Contribution of DN models over a traditional IR baseline. In order to evaluate the effectiveness of DN models, we compare the MatchPyramid model with the traditional BM25 baseline to assess its effectiveness for tweet retrieval. Table 3 shows the results obtained on the TREC RTS 2016 and 2017 datasets. For the 2016 collection, MatchPyramid clearly outperforms BM25 in terms of MAP and P@10. For the 2017 collection, we notice that MatchPyramid is able to retrieve more relevant tweets than BM25, however, it has a lower P@10 indicating that it has less capacity to rank these tweets in the top-10.

4.2.3 Evaluation of the proposed tweet summarization method. In this experimentation, we compare our overall approach with a prior work on Microblog summarization and the two best official runs of TREC RTS 2016 and 2017. The main results are reported in Table 4.

TREC RTS 2016. The results show that our approach outperforms the second best TREC RTS-2016 submission with an improvement of more than 34.78% in terms of nDCG0@10. We also notice an improvement of 4.5% in terms of nDCG0@10 over the best TREC RTS-2016 submission. We recall that the nDCG1@10 measures rewards, with a gain of 1, any system that does not push tweets in a silent day. The inconvenient of this measure is that a system can have a non-zero score for an empty submission (no results returned on any given day). Indeed, it is difficult for a system to obtain a high gain (close to 1) while an empty submission allows the system to easily have a gain of 1 on silent days, knowing that 30.89% of the days of the RTS-2016 campaign are silent.

TREC RTS 2017. We notice that our approach does not perform well on this collection. This can be explained by two reasons: The

Table 4: Evaluation on the TREC RTS replay mechanism of scenario B. Metrics consider only the top-10 results (@10).

Model	TREC RTS 2016		TREC RTS 2017	
	nDCG0	nDCG1	nDCG1	nDCGp
Ours	07.13	27.13	13.18	24.26
TF-IDF [2]	08.34	17.45	25.70	31.66
HybridTF-IDF [13]	07.67	16.78	24.97	30.95
Sumbasic [10]	05.36	16.55	24.24	30.22
PolyURunB3 [15]	06.84	28.98	-	-
nudtsna [17]	05.29	27.08	-	-
HLJIT_qFB_url [5]	-	-	29.10	36.56
PKUICSTRunB1 [16]	-	-	30.03	34.83

first one concerns the candidate relevant tweets retrieval. We think that the relevance estimation using MatchPyramid was not able to outperform BM25 in terms of P@10. Considering a poor candidate list of tweets, it is difficult to construct a good summary. The second reason may be related to the silent days issue that we did not handle in this work.

5 CONCLUSION

In this work, we have presented an experimental study where we consider tweet summarization as a tweet retrieval task. To overcome the issue of the semantic matching between user query and tweets, we have investigated the impact of distributed representations and DN models on the retrieval task. The analysis of the obtained results have shown that no representation model was best for all the evaluated DN models for the TREC RTS task. However, MatchPyramid model yields the best results with an optimum on locally pre-trained embeddings with Fasttext model. In addition, we have shown that this configuration outperforms BM25. We concluded that neural models can effectively learn to retrieve relevant tweets. Hence, they are less performing for ranking which is challenging. Since the performance of DN models depends highly on the amount of training data, in future work, we plan on investigating weak supervision in order to generate training data at low cost.

REFERENCES

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. *CoRR* abs/1607.04606 (2016). arXiv:1607.04606 http://arxiv.org/abs/1607.04606

[2] Deepayan Chakrabarti and Kunal Punera. 2011. Event Summarization Using Tweets. *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media.*

[3] Abdelhamid Chellal. 2018. *Event Summarization on Social Media Stream: Retrospective and Prospective Tweet Summarization.* Thèse de doctorat. Université Paul Sabatier, Toulouse, France. https://www.irit.fr/publis/IRIS/2018_These-CHELLAL.pdf

[4] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management.* 55–64.

[5] Zhongyuan Han, Song Li, Leilei Kong, Liuyang Tian, and Haoliang Qi. 2017. HLJIT at TREC 2017 Real-Time Summarization. In *Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017, Gaithersburg, Maryland, USA, November 15-17, 2017*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. Special Publication 500-324. National Institute of Standards and Technology (NIST).

[6] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems.* 2042–2050.

[7] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[9] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web.* 1291–1299.

[10] Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005* 101 (2005).

[11] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Thirtieth AAAI Conference on Artificial Intelligence.*

[12] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP).* 1532–1543. http://www.aclweb.org/anthology/D14-1162

[13] Beaux Sharifi, Mark-Anthony Hutton, and Jugal K Kalita. 2010. Experiments in microblog summarization. In *2010 IEEE Second International Conference on Social Computing.* IEEE, 49–56.

[14] Lidan Shou, Zhenhua Wang, Ke Chen, and Gang Chen. 2013. Sumblr: continuous summarization of evolving tweet streams. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval.* 533–542.

[15] Haihui Tan, Dajun Luo, and Wenjie Li. 2016. PolyU at TREC 2016 Real-Time Summarization. In *Proceedings of The Twenty-Fifth Text REtrieval Conference, TREC 2016, Gaithersburg, Maryland, USA, November 15-18, 2016*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. Special Publication 500-321. National Institute of Standards and Technology (NIST).

[16] Jizhi Tang, Chao Lv, Lili Yao, and Dongyan Zhao. 2017. PKUICST at TREC 2017 Real-Time Summarization Track: Push Notifications and Email Digest. In *Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017, Gaithersburg, Maryland, USA, November 15-17, 2017*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. Special Publication 500-324. National Institute of Standards and Technology (NIST).

[17] Xiang Zhu, Jiuming Huang, Sheng Zhu, Ming Chen, Chenlu Zhang, Zhenzhen Li, Huang Dongchuan, Zhao Chengliang, Aiping Li, and Yan Jia. 2015. NUDTSNA at TREC 2015 Microblog Track: A Live Retrieval System Framework for Social Network based on Semantic Expansion and Quality Model. In *Proceedings of The Twenty-Fourth Text REtrieval Conference, TREC 2015, Gaithersburg, Maryland, USA, November 17-20, 2015*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. Special Publication 500-319. National Institute of Standards and Technology (NIST).

[18] Arkaitz Zubiaga, Damiano Spina, Enrique Amigó, and Julio Gonzalo. 2012. Towards real-time summarization of scheduled events from twitter streams. In *Proceedings of the 23rd ACM conference on Hypertext and social media.* 319–320.