Evaluating Similarity Metrics for Latent Twitter Topics

Xi Wang(✉), Anjie Fang, Iadh Ounis, and Craig Macdonald

University of Glasgow, Glasgow, UK
{x.wang.6,a.fang.1}@research.gla.ac.uk,
{iadh.ounis,craig.macdonald}@glasgow.gla.ac.uk

Abstract. Topic modelling approaches such as LDA, when applied on a tweet corpus, can often generate a topic model containing redundant topics. To evaluate the quality of a topic model in terms of redundancy, topic similarity metrics can be applied to estimate the similarity among topics in a topic model. There are various topic similarity metrics in the literature, e.g. the Jensen Shannon (JS) divergence-based metric. In this paper, we evaluate the performances of four distance/divergence-based topic similarity metrics and examine how they align with human judgements, including a newly proposed similarity metric that is based on computing word semantic similarity using word embeddings (WE). To obtain human judgements, we conduct a user study through crowdsourcing. Among various insights, our study shows that in general the cosine similarity (CS) and WE-based metrics perform better and appear to be complementary. However, we also find that the human assessors cannot easily distinguish between the distance/divergence-based and the semantic similarity-based metrics when identifying similar latent Twitter topics.

1 Introduction

Twitter has become a popular way for people to express their opinions and preferences. Researchers are often interested in examining the topics that are being discussed on such a platform [1–3]. To this end, topic modelling approaches, such as Latent Dirichlet Allocation (LDA), can be used to identify topics [2, 4]. However, redundant topics can cost researchers more time when examining their content. Therefore, it is necessary to identify the redundant topics before presenting them to the researchers. We assume that highly similar topics could be redundant and a topic similarity metric can be used to calculate the similarity among topics generated by a topic modelling approach.

We evaluate various topic similarity metrics in order to offer practical suggestions on how to effectively measure the similarities among latent topics generated from Twitter streams. A topic in a topic model is a distribution over words [4]. Commonly, the similarities of topics can be computed by using the distribution of topics over the vocabulary. Previous work has applied metrics such as the Hellinger distance (HD) [5], the Jensen Shannon (JS) divergence [6] or the

L. Azzopardi et al. (Eds.): ECIR 2019, LNCS 11437, pp. 787–794, 2019.
https://doi.org/10.1007/978-3-030-15712-8_54

cosine similarity (CS) [1,7] to measure the similarity between topics. These metrics compute the distance/divergence of topic distributions. We also propose and evaluate a new word embedding (WE)-based metric to measure the semantic similarity between topics, since word embedding has been reported to more effectively capture the semantic similarity [3,8].

We conduct a user study through crowdsourcing to examine the effectiveness of the four aforementioned similarity metrics (i.e. HD, JS, CS and WE). Our crowdsourced user study shows that the human assessors cannot easily distinguish between the distance/divergence-based and the semantic similarity-based metrics when identifying similar topics. However, we also find that, in general, the CS and WE-based metrics align the best with human judgements, as they outperform at least one other metric on our Twitter dataset. In particular, our results suggest that the CS and WE-based metrics appear to be complementary. While the CS-based metric can better assess the topic similarity when topics share the same frequent words, the WE-based metric can better capture the semantic similarity of topics. Overall, our paper contributes new insights about measuring topic similarity in Twitter, and how the topic similarity metrics perform compared to human judgements.

2 Related Work

Typically, three types of metrics can be used to capture the similarity between topics: **(1) Divergence-based metrics**. Gretarsson et al. [9] and Kim et al. [6] applied the Kullback Leibler (KL) and Jensen Shannon (JS) divergence metrics to measure the textual differences of latent topics. Kim et al. [6] concluded that the JS divergence gave the best performance when compared to the other approaches tested. **(2) Coefficient-based metrics**. The coefficient-based metrics, Jaccard's Coefficient, Kendall's τ coefficient, and discounted cumulative gain can all be used to compute the similarity between topics. However, Kim et al. [6] showed that the divergence-based metrics are better than these coefficient-based metrics, as the coefficient-based metrics require a corpus-dependent probability mass. **(3) Distance-based metrics**. Gretarsson et al. [9] estimated the similarity of latent topics by computing the L_1 distance. Later, Maiya et al. [5] adopted the Hellinger distance metric to calculate the similarity between topics. On the other hand, the most common distance metric used in the literature is the cosine similarity [6,10,11]. Indeed, the cosine similarity has been shown to provide superior performance compared to other divergence-based metrics [7].

In [12], Mikolov et al. proposed a shallow learning technique called word2vec, which represents individual words as high dimensional word embedding vectors. These word representations can be used to capture the semantic similarity between words [13]. However, it is unclear which of the aforementioned types of metrics better reflects a user's view of topic similarity on Twitter. Based on these prior studies, we choose the JS divergence, the Hellinger distance, the cosine as well as a new word embedding-based similarity for evaluation in our user study.

3 Metrics and Methodology

We introduce the used topic similarity metrics and their differences. The cosine similarity-based metric (CSM) can be applied over two distributions (i.e. two vectors) for computing the similarity of two topics. This method has been previously used in [3,14]. The JS divergence-based metric (JSM) is a symmetric form of the KL divergence. It is often used as a topic similarity metric in prior work [6,9]. The Hellinger distance-based metric (HDM) is often used to quantify the similarity between a pair of probability distributions, as in [5].

In addition, we propose a word embedding-based metric (WEM), where each topic is represented by its top n words, ranked by its words' posterior topic probabilities. We then compute the similarity of two topics by the pairwise word semantic similarity shown in Eq. (1), where W_i denotes the set of top n words for topic i, and Vec_p indicates the vector of word p in a WE model.

$$WES(\theta_i, \theta_j) = \sum_{p \in W_i} \min_{\forall q \in W_j} cosine(Vec_p, Vec_q) \tag{1}$$

Differences Among Metrics. Each of the 4 aforementioned metrics focuses on different aspects when estimating the topic similarity, providing a good representative sample of similarity metrics to compare to human judgements. The CS-based metric tends to compute the similarity using words with high frequencies. Compared to CSM, JSM and HDM alleviate the effects of high-frequency words. Moreover, while JSM tends to normalise the word probability differences, the HD-based metric applies a square root to smooth the probability differences. Unlike the CS, JS and HD-based metrics, which compute the similarity of topics using the whole topic distributions, the WE-based metric exploits instead the semantic similarity between the top-ranked words in the generated topics.

Pairwise Comparison of Metrics. We evaluate the performances of the four metrics using a pairwise approach, i.e. assessing the performances of each pair of the four metrics. This pairwise comparison method has been previously applied in the literature to compare different systems [3]. Specifically, given a topic from a topic model (we call it the **base** topic) and two metrics A & B (a metric pair), we use metric A and B to choose two **candidate** topics that are the most similar to the base topic. Two candidate topics together with their base topic are called a *topic set*. For each metric pair, we sample a number of topic sets. We conduct a user study to obtain the ground-truth from human judgements. A metric in a metric pair obtains a score of "1" if it aligns with the human judgement on a topic set, otherwise, "0". Accordingly, we use a signed-rank test on a set of generated paired scores to identify the statistically significant between each metric pair.

Twitter Dataset. We use a Twitter dataset that is related to the US 2016 election and which contains tweets posted from 01/07/2016 to 31/10/2016. This dataset has 18k sample tweets[1] collected by searching a list of keywords related

[1] This sample of tweets is in English, does not contain retweets and each tweet has at least 5 words.

to the US 2016 election (e.g. “Trump”, “Hillary”, “debate”, “vote”, “election”, etc.) using the Twitter Streaming API[2]. Since the election contains numerous discussions across a range of topics, this election-related dataset allows us to obtain sufficient topics for applying a topic modelling approach such as LDA.

4 Crowdsourced User Study

We now describe how we perform the user study to obtain human judgements. The CrowdFlower[3] platform is used. Each worker is presented with multiple topic sets. Similar to [3,15], a topic is represented by the 15[4] most frequent words from its word distribution. A worker is asked to choose a topic out of the two candidate topics, which is the most similar to the base topic. If a worker cannot make a decision, they can select the option of “Either of them”. To help the workers undertake the task, we provide them with guidelines that explain how to identify the most similar topic. For example, they can check whether the base and candidate topics contain words that refer to the same topic. We also provide a list of election-related hashtags (e.g. #FeelTheBern, #Wikileaks) and some commonly mentioned key players in the election (e.g. Mike Pence, Tim Kaine) with their corresponding descriptions. After the workers choose a given candidate topic, they are asked to specify how *easy* they found the question. Next, we explain how we generate topic sets and our precise used experimental setup for the user study.

Generating Topics. We apply Gibbs sampling [16], an approximate inference technique for LDA[5], to generate topics from the election Twitter data. The number of topics K is set to 90[6] and we generate 10 repeated topic models[7]. For each of the chosen topic models, we use the topic coherence metric [3], which has been shown to be particularly effective on Twitter compared to other existing metrics, to rank the 90 topics by their coherence. Then, we select the top 30 topics out of 90 from each topic model. We obtain 300 topics as the pool of base topics. For each metric pair, we randomly select 50 base topics from the base topic pool. For a given metric pair, each metric selects the most similar topic to a base topic as a candidate topic. Accordingly, we obtain 50 topic sets[8] for each metric pair (300 in total).

[2] https://dev.twitter.com.

[3] http://crowdflower.com.

[4] In [3,15], the top 10 words are used to estimate a given topic’s coherence. However, Ramage et al. [1] argued that the top-ranked words might often be similar. Hence, we choose to use the top 15 words in this work.

[5] We use Gibbs sampling as it can still generate topics that connect well to the real topics (see [2]). We plan to study topic similarity using different LDA approaches in the future work.

[6] We found that topic models with $K = 90$ have a higher coherence according to the topic coherence metric [3] used in our experiments.

[7] Each topic model contains 90 topics.

[8] The order of topics in the topic sets is shuffled.

User Study Setup. We first limit the CrowdFlower workers to the US as the topics are related to the US election. In total, we had 60 workers who passed the test and entered the task. Among the 60 workers, 35 workers maintained the required accuracy of 70% and their judgements were retained. Each worker has to spend at least 10 s on each question and can only answer at most 20 judgements. Such a setup allows us to obtain judgements from many users. We pay a worker US$0.05 for each question. We obtain at least 3 judgements for each question. We require a minimum agreement of 60% among the 3 workers on any of their answers. Otherwise, additional workers are allocated the same question until such an agreement is reached. Among the 300 questions, 38.4% required additional workers.

Setup of the WE-Based Metric. We use tweets to train the word embedding model, since our topics are generated from tweets. First, we use the Twitter Streaming API (sample mode) to crawl a collection of random tweets posted from January to July in 2016. The size of this collection is about 200 million tweets. To obtain the embedding, we apply fastText[9] on the collected tweets. Our WE-based metric leverages this trained embedding to evaluate the similarity of the top 15 words in two topics.

Table 1. Comparison of the 6 metric pairs. Statistically significant differences are indicated by *.

	CSM vs. WEM	CSM vs. JSM	CSM vs. HDM	WEM vs. JSM	WEM vs. HDM	JSM vs. HDM
# of votes	25 vs. 25	31 vs. 19	23 vs. 27	23 vs. 24	30 vs. 19	23 vs. 23
p-Value	1.0	0.03*	0.49	0.86	0.05*	1.0

5 Results Analysis

We first report the metric preferences from our user study. Then we report a qualitative analysis of the results.

For the 300 topic sets, we obtain 900 judgements from 21 different workers. In terms of task difficulty, among the collected judgements, 22% (196) of them are labelled as "easy" and 75.6% (628) are "reasonable". Only 2.4% (66) of these judgements are "hard" for humans to make. This suggests that the task of our user study is reasonably easy for the workers. We use the method explained in Sect. 3 to calculate the p-value, which indicates whether two metrics perform significantly differently. The number of votes and p-values of the 6 metric pairs are listed in Table 1. For example, "31 vs. 19" in the CSM vs. JSM column indicates that the CSM metric (with 31 votes) significantly outperforms the JSM metric with (19 votes). Similarly, we also observe that the WEM metric is significantly better than HDM.

[9] http://fasttext.cc. The context window size is 5 and the dimension of the vector is 100.

Overall, we do not observe significant differences among the rest of 4 metric pairs. As mentioned in Sect. 3, the CS, JS and HD-based metrics consider the probabilities of all the topics' words while the WE-based metric focuses on the semantic similarity of top-ranked words. Since neither the WE-based metric nor the other 3 metrics are consistently better than the rest of metrics, this suggests that the two types of metrics align equally well with the human judgements when assessing topics similarity. On the other hand, while no metric in this study consistently beats all the others, we do observe that, in general, the CS and WE-based metrics perform the best and outperform the other 2 metrics. In addition, according to the signed-rank significance test, only CSM outperforms JSM and WEM outperforms HDM significantly. Hence, later, we further analyse the CS and WE-based metrics, their differences and why they were the preferred metrics according to human judgements.

Table 2. Topic sets of WEM vs. CSM by columns

Base topic: people #trump talking @realdonaldtrump #hrc making guy believe abt hey actually fake democratic supporter trying	**Base topic:** #wikileaks #draintheswamp #podestaemails #votetrump #voterfraud #neverhillary #trump yeah dems #alsmithdinner #corruption politics @realdonaldtrump dump readin
Candidate topic 1 (selected by CSM): #trump #putin russia putin #rednationrising talking #billclinton #tgdn morning tomorrow want iran pennsylvania gold standard	**Candidate topic 1 (selected by WEM):** #neverhillary #trumppence @realdonaldtrump polls #makeamericagreatagain watching @hillaryclinton comes watch way right nov crap #corrupthillary strong
Candidate topic 2 (selected by WEM): @realdonaldtrump @gop #hrc #america @thedemocrats say course point truth campaign support telling @reince isn moment	**Candidate topic 2 (selected by CSM):** #wikileaks emails #podestaemails @wikileaks fuck hacked according #octobersurprise trending report #assange funded @hillaryclinton staff coverage

Overall, the CS and WE-based metrics performed better than the other two metrics. However, from the signed-rank test, there is no evidence indicating that one metric is significantly better than the other. By examining the topic sets, we find that these two metrics perform differently in different scenarios. CSM is good at matching the most similar topic set when their informative words have high frequencies. The candidate topic selected by CSM is intuitively more similar to the base topic. However, CSM might fail to select the most similar one if the base topic does not share high-frequency words with any candidate topic. On the other hand, WEM can work better in this instance, as it puts more emphasis on the semantic similarity among top words. For example, "#vote" is related to "vote", "votes", "winning", etc. WEM allows to capture the semantic relationships between two topics. However, if two topics share several words with high frequencies, WEM does not outperform CSM, since CSM effectively captures the similarity. For instance, in the first column of Table 2, CSM fails to match the top words in the base topic, which results in the choice of a non-relevant candidate topic 1 (more about "putin" and "russia"), while candidate topic 2 chosen by WEM is better. On the contrary, in the second column of Table 2, when topics share the top words (e.g. "#wikileaks" and "email"), the CSM performs better than WEM. In general, we see a complementary relationship between WEM and CSM.

There are several reasons why our user study did not distinguish between 4 out of 6 metric pairs: CSM vs. WEM, CSM vs. HDM, WEM vs. JSM and JSM vs. HDM. First, two metrics can perform very similarly and thus humans cannot effectively distinguish between their chosen candidate topics. For example, for JSM vs. HDM, given a base topic, we find that 75% of the top 10 most similar topics ranked by the JSM and HDM metrics in a topic model are the same on average. Second, the number of topic set samples might not be large enough, and thus the statistical test cannot find a statistical difference between the two metrics. To conclude, our study shows that using our Twitter dataset, the CSM and WEM metrics align best with human judgements, and markedly outperform the HDM and JSM metrics in estimating the similarity of latent Twitter topics.

6 Conclusions

We studied the effectiveness of 4 commonly used similarity metrics when examining the similarity of latent topics on Twitter. We conducted a user study to ascertain which of the metrics align best with human judgements. Our study showed that, on our used Twitter dataset, the human assessors cannot distinguish between the distance/divergence-based metrics and the semantic similarity-based metric when identifying similar latent Twitter topics. However, the CS and WE-based metrics markedly outperformed the HDM and JSM metrics. In particular, we found that the CS and WE-based metrics appear to be complementary. While the CS-based metric better estimates similarity when the topics share the same high-frequency words, the WE-based metric better captures the semantic relationships among topics. Such complementarity might help to construct topic models with different requirements. As future work, we aim to conduct the same analysis on different datasets, and investigate how to seamlessly combine the CS and WE-based metrics to effectively estimate the similarity of latent topics on Twitter to further reduce redundancy in the generated topic models.

References

1. Ramage, D., Dumais, S.T., Liebling, D.J.: Characterizing microblogs with topic models. In: Proceedings of ICWSM (2010)
2. Zhao, W.X., et al.: Comparing Twitter and traditional media using topic models. In: Clough, P., et al. (eds.) ECIR 2011. LNCS, vol. 6611, pp. 338–349. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20161-5_34
3. Fang, A., Macdonald, C., Ounis, I., Habel, P.: Using word embedding to evaluate the coherence of topics from Twitter data. In: Proceedings of SIGIR (2016)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. Mach. Learn. Res. **3**, 993–1022 (2003)
5. Maiya, A.S., Rolfe, R.M.: Topic similarity networks: visual analytics for large document sets. In: Proceedings of IEEE Big Data (2014)

6. Kim, D., Oh, A.: Topic chains for understanding a news corpus. In: Gelbukh, A. (ed.) CICLing 2011. LNCS, vol. 6609, pp. 163–176. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19437-5_13
7. Aletras, N., Stevenson, M.: Measuring the similarity between automatically generated topics. In: Proceedings of EACL (2014)
8. Nikolenko, S.I.: Topic quality metrics based on distributed word representations. In: Proceedings of SIGIR (2016)
9. Gretarsson, B., et al.: TopicNets: visual analysis of large text corpora with topic modeling. ACM Trans. Intell. Syst. Technol. **3**(2.23), 1–26 (2012)
10. Ramage, D., Hall, D., Nallapati, R., Manning, C.D.: Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of EMNLP (2009)
11. Fang, A., Macdonald, C., Ounis, I., Habel, P., Yang, X.: Exploring time-sensitive variational Bayesian inference LDA for social media data. In: Jose, J.M., et al. (eds.) ECIR 2017. LNCS, vol. 10193, pp. 252–265. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56608-5_20
12. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of NIPS (2013)
13. Levy, O., Goldberg, Y.: Linguistic regularities in sparse and explicit word representations. In: Proceedings of CoNLL (2014)
14. Huang, A.: Similarity measures for text document clustering. In: Proceedings of NZCSRSC (2008)
15. Fang, A., Macdonald, C., Ounis, I., Habel, P.: Topics in tweets: a user study of topic coherence metrics for Twitter data. In: Ferro, N., et al. (eds.) ECIR 2016. LNCS, vol. 9626, pp. 492–504. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30671-1_36
16. Darling, W.M.: A theoretical and practical implementation tutorial on topic modeling and Gibbs sampling. In: Proceedings of ACL HLT (2011)