Community pooling: LDA topic modeling in Twitter

Federico Albanese [1 2] **Esteban Feuerstein** [1 3]

Abstract

Social networks play a fundamental role in propagation of information and news. Characterizing the content of the messages becomes vital for tasks like fake news detection or personalized message recommendation. However, Twitter posts are short and often less coherent than other text documents, which makes it challenging to apply text mining algorithms efficiently. We propose a new pooling scheme for topic modeling in Twitter, which groups tweets whose authors belong to the same community on the retweet network into a single document. Our findings contribute to an improved methodology for identifying the latent topics in a Twitter dataset, without modifying the basic machinery of a topic decomposition model. In particular, we used Latent Dirichlet Allocation (LDA) and empirically showed that this novel method achieves better results than previous pooling methods in terms of cluster quality, document retrieval tasks, supervised machine learning classification and overall run time.

1. Introduction

Characterizing texts based on their content is an important task in machine learning and natural language processing. Latent Dirichlet Allocation (LDA) is a generative model for unsupervised topic decomposition (Blei et al., 2003). Documents are represented as random mixtures over topics, and each topic is characterized by a distribution over words.

In practice, content analysis on microblogging services can be particularly challenging due to short and often vaguely coherent text (Mehrotra et al., 2013; Ma et al., 2019). Given the fact that Twitter has become a platform where a tremendous amount of content is generated, shared and consumed, this problem become of interest for the scientific community Hong and Davison presented an intuitive solution: tweet pooling (making a longer document by aggregating multiple tweets following different schemes) (Hong & Davison, 2010). Tweet-pooling has been shown to improve topic decomposition, but the performance varies depending on the pooling method (Hong & Davison, 2010; Mehrotra et al., 2013; Ma et al., 2019; Alvarez-Melis & Saveski, 2016; Ollagnier & Williams, 2019). For example, Mehrotra et al. (Mehrotra et al., 2013) extended this idea by pooling all tweets that mention a given hashtag. More pooling techniques are described in detail in section 2.

In this paper, we proposed a novel pooling techniques based on community detection on graphs. Previous works stated that LDA has problems with sparse word co-occurrence matrix (Ma et al., 2019) and showed that users in a community tweet mostly about one or two particular topics (Albanese et al., 2020). Based on these issues, we proposed a community pooling method which groups tweets whose authors belong to the same community on the retweet network, increasing the length of each document and reducing the total number of documents. We empirically showed that this scheme improves the performance over previous pooling methods in a generic twitter dataset in terms of clustering quality, document retrieval, supervised machine learning classification tasks and run time.

This work is organized as follows: In Section 2 we described the different pooling schemes for topic models and propose a novel method. In Section 3 we described the dataset that we used to test our method. In section 4, we defined the experiments and evaluation metrics that we used to measure the performance of all pooling schemes. In section 5, we showed the results of the experiments. Finally, we interpreted the results in the conclusions section.

2. Tweet pooling for topic models

Microblog messages are very short texts. In particular, Twitter posts are only 280 characters or shorter. Consequently, using each tweet as an individual document does not present adequate term co-occurence data within documents (Mehrotra et al., 2013). This induced the idea that aggregating similar tweets gives place to larger documents and allowed LDA to learn better topic decompositions. In this section,

[1]Instituto en Ciencias de la Computación, CONICET - Universidad de Buenos Aires, Argentina. [2]Instituto de Cálculo, CONICET-Universidad de Buenos Aires, Argentina. [3]Departamento de Computación, Universidad de Buenos Aires, Argentina.. Correspondence to: Federico Albanese <falbanese@dc.uba.ar>.

Proceedings of the LatinX in AI (LXAI) Research workshop at ICML 2021.

we described five tweet pooling methods for topic modeling proposed in the literature and presented a new pooling method based on community detection.

Tweet-pooling (Unpooled): The default approach which treated each tweet as a single document. This served as our baseline for comparison to pooled schemes.

Author-Pooling: All tweets authored by a single user were aggregated in a single document. The number of documents was equal to the number of users. This pooling method outperformed the Unpooled scheme (Hong & Davison, 2010).

Hashtag pooling: In this scheme, a document consisted of all tweets that mention a given hashtag. A tweet that contains multiple hashtags appeared in several documents. Tweets without hashtags were considered as individual documents. It has been shown that this method outperforms the baseline scheme and user-polling (Mehrotra et al., 2013).

Conversation pooling:

A document consisted of all tweets in a conversation tree (a tweet, all the tweets written in reply to it, and the replies of the replies, and so on). These schemes aggregated tweets from different authors and with multiple hashtags that belong to one conversation and has been shown to outperform other pooling schemes (Alvarez-Melis & Saveski, 2016).

Network-based pooling:

Twitter users were grouped together if they replied or were mentioned in a tweet or a replies to a tweet. Each document consisted of all tweets of a group of users. In contrast to Conversation pooling, only direct replies to an original tweet were considered since a conversation could shift its topic in time. This pooling scheme showed better results than the previous methods (Ollagnier & Williams, 2019).

Community pooling: [1]

A retweet graph was defined in terms of $G = (N, E)$, where users were nodes N and retweets between them were edges E (Albanese et al., 2020). Since a user could retweet multiple times other user's tweets, the edges were weighted. We found the communities of users using the Louvain method for community detection (Blondel et al., 2008), which seeks to maximize modularity by using a greedy optimization algorithm. Therefore, each community clustered users by their interactions. We proposed a novel method where a single document consist of all tweets authored by all users in each community. There were as many documents as communities in the retweet network. In contrast to previous schemes, the number of words in a document was bigger and the number of documents was smaller, making the word co-occurrence matrix more dense. Considering that LDA allows multiple topics in one single document, having longer documents with denser co-occurence matrix has been shown to be beneficial (Alvarez-Melis & Saveski, 2016). Also, other works had shown that tweets belonging to users of one community in the retweet network were mostly of one or two topics (Albanese et al., 2020).

[1]The source code of the Community pooling is available at https://github.com/fedealbanese/Community_pooling

3. Twitter dataset construction

Our experiments used data from Twitter Streaming API [2]. Similar to previous works, we constructed a dataset collecting tweets containing generic terms and each tweet was labeled by the query that retrieved it (Hong & Davison, 2010; Al-Sultany & Aleqabie, 2019; Mehrotra et al., 2013; Ollagnier & Williams, 2019). We removed all tweets that were retrieved by more than one query, so as to preserve uniqueness of the tweet labels, which was important for our analysis. All tweets had to be in English and where collected from December 15^{th} to December 16^{th}, 2020. The dataset consisted of a total of 115359 tweets. The generic terms and their percentage of tweets retrieved by each query were the following: "music" (36.78%), "family" (23.94%), "health" (17.21%), "business" (14.90%), "movies" (4.70%) and "sports" (2.44%). We prepossessed the tweets by lowercasing and removing stop-words.

4. Evaluation

Because there is no single method for evaluating topic models, previous work evaluated their proposed pooling method using different metrics or tasks. Each evaluation measured the performance of the pooling method differently. In order to present a complete and exhaustive analysis of previous schemes and the proposed pooling method, in this work we evaluated the proposed methodology using topic clustering metrics (Purity and Normalized Mutual Information) (Alvarez-Melis & Saveski, 2016; Hajjem & Latiri, 2017; Mehrotra et al., 2013; Quezada & Poblete, 2019; Ollagnier & Williams, 2019; Akhtar & Beg, 2019), a supervised machine learning classification task (Giorgi et al., 2018; Hong & Davison, 2010) and a document retrieval task (Alvarez-Melis & Saveski, 2016; Al-Sultany & Aleqabie, 2019).

Purity:

We defined each cluster as a topic and assigned the tweet to its corresponding mixture topic of the highest probability (a quantity estimated with LDA).

The purity of a cluster measured the fraction of tweets in a cluster having the assigned cluster query label (Schütze et al., 2008). Formally, let T_i be the set of tweets in LDA topic cluster i and Q_j be the set of tweets with query label j. Let $T = \{T_1, T_2, ..., T_{|T|}\}$ be the set of size $|T|$ of all T_i

[2]https://developer.twitter.com/en

and let $Q = \{Q_1, Q_2, ..., Q_{|Q|}\}$ be the set of size $|Q|$ of all Q_j. Then, the purity is defined as follows:

$$Purity(T, Q) = \frac{1}{|T|} \sum_{i \in \{1...|T|\}} max_{j \in \{1...|Q|\}} |T_i \cap Q_j| \quad (1)$$

A higher purity score reflect a better cluster representation and a better LDA decomposition.

Normalized Mutual Information (NMI):

NMI measures the cluster quality using information theory and it is formally defined as follows:

$$NMI(T, Q) = \frac{2I(T, Q)}{H(T) + H(Q)} \quad (2)$$

where $I(\cdot, \cdot)$ is the mutual information and $H(\cdot)$ is the entropy, as defined in (Schütze et al., 2008). NMI minimum value is 0 (labels and cluster are independent sets) and maximum value is 1 (cluster results exactly matches all labels).

Supervised machine learning classifying task:

For the supervised machine learning task, we followed a basic machine learning classifying evaluation scheme (Hong & Davison, 2010). We separated the dataset in two parts (train and test), trained a classifier with the first one and performed a simple cross-validation on the second one. The first 80% of tweets (according to the time their were posted) were assigned to the train set and the other 20% to the test set. For this task, we trained two classic Machine learning models: a naive Bayes classifier and random forest classifier (Müller & Guido, 2016). We reported F-Measure (F1 score) of the different machine learning models on the test set.

Document retrieval task:

We also evaluated the topic decomposition of the different pooling methods on a document retrieval task. Using the same train-test split as the supervised classifier task, we used each tweet in the test set as a query and return tweets form the train set with the most topic similarity. If the retrieved tweets have the same query label, we consider it relevant.

Accordingly, we applied LDA using the different pooling techniques on the train set, for each tweet in the test set calculate its topic decomposition, computed the cosine similarity between its topic decomposition and the topic decomposition of all tweets in the train set and retrieved the top 10 most similar train tweets. Then, we calculated the F1 score in order to know if the categories of the retrieved tweets match the category of the test tweet.

Table 1. Purity and NMI clustering results.

SCHEME	PURITY	NMI
TWEET (UNPOOLED)	0.664	0.436
AUTHOR	0.696	0.374
HASHTAG	0.724	0.383
CONVERSATION	0.658	0.436
NETWORK-BASED	0.695	0.372
COMMUNITY	**0.780**	**0.439**

Table 2. Supervised classification F1 scores for naive Bayes and random forest.

SCHEME	RANDOM FOREST	NAIVE BAYES
TWEET (UNPOOLED)	0.833	0.814
AUTHOR	0.818	0.798
HASHTAG	0.813	0.779
CONVERSATION	0.829	0.814
NETWORK-BASED	0.821	0.798
COMMUNITY	**0.839**	**0.827**

5. Results

In this section we showed and discussed the results of the evaluation tasks described in the previous section. For each pooling scheme, we replicated the training workflow used in the literature and used an LDA model with 10 topics (Mehrotra et al., 2013; Ollagnier & Williams, 2019).

The results of the Purity and the NMI can be seen in table 1. The best performance is marked in bold. The table shows that the proposed Community Pooling has the best cluster in terms of the highest Purity and NMI scores of all methods.

Table 2 portrays the results of the machine learning classification task, which indicates if the topic decomposition is a good descriptor of the query label. Again, our experiments show that the best performance was achieved by the proposed Community Pooling method.

The document retrieval task considers small changes in the topic decomposition of a tweet, since it uses the cosine similarity between this decomposition instead of only taking into account the most likely topic as we did with the clustering metrics. The results can be seen in table 3 and shows that the proposed Community Pooling has the best performance.

Finally, we reported the run time of all pooling techniques. The measured time includes tweet pooling (aggregating the tweets in different documents) and the LDA topic modeling, which varies depending on the total number of documents of each pooling methods. The results can be seen in table 4. The unpooled method has the best performance, followed by the proposed community pooling. Considering the fact that community pooling considerably reduces the number of

Table 3. Document Retrieval Task F1 scores.

SCHEME	F1
TWEET (UNPOOLED)	0.837
AUTHOR	0.839
HASHTAG	0.839
CONVERSATION	0.835
NETWORK-BASED	0.840
COMMUNITY	**0.843**

Table 4. Time performances in seconds.

SCHEME	RUN TIME
TWEET (UNPOOLED)	137.5
AUTHOR	429.7
HASHTAG	1737.2
CONVERSATION	738.1
NETWORK-BASED	1131.9
COMMUNITY	141.2

documents (and therefore the LDA topic modeling time), it follows that all other techniques have more than the double of the run time. All experiments were run using the same hardware on a GTX 1080 NVIDIA graphic card.

6. Conclusions

This paper presented a new way of pooling tweets in order to improve the quality of LDA topic modeling on Twitter. The methodology used here addresses the challenge of improving topic modeling without requiring any modification of the underlying LDA algorithm. Multiple pooling techniques were evaluated on different task including clustering metrics, supervised classification problems and document retrieval task. Our results indicate that the novel community based pooling outperforms all other pooling strategies in all tasks and metrics in a generic twitter dataset. Also, the run time analysis shows a significant improvement in time performance in comparison with the other pooling methods. In conclusion, we showed that building the retweet network, finding the communities and aggregate tweets based on the users cluster is an improvement over previous methods. Future work includes using multiple datasets.

References

Akhtar, N. and Beg, M. User graph topic model. *Journal of Intelligent & Fuzzy Systems*, 36(3):2229–2240, 2019.

Al-Sultany, G. A. and Aleqabie, H. J. Events tagging in twitter using twitter latent dirichlet allocation. *International Journal of Engineering & Technology*, 8(1.5):503–508, 2019.

Albanese, F., Lombardi, L., Feuerstein, E., and Balenzuela, P. Predicting shifting individuals using text mining and graph machine learning on twitter. *arXiv preprint arXiv:2008.10749*, 2020.

Alvarez-Melis, D. and Saveski, M. Topic modeling in twitter: Aggregating tweets by conversations. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 10, 2016.

Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *the Journal of machine Learning research*, 3: 993–1022, 2003.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

Giorgi, S., Preotiuc-Pietro, D., Buffone, A., Rieman, D., Ungar, L. H., and Schwartz, H. A. The remarkable benefit of user-level aggregation for lexical-based population-level predictions. *arXiv preprint arXiv:1808.09600*, 2018.

Hajjem, M. and Latiri, C. Combining ir and lda topic modeling for filtering microblogs. *Procedia Computer Science*, 112:761–770, 2017.

Hong, L. and Davison, B. D. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pp. 80–88, 2010.

Ma, T., Li, J., Liang, X., Tian, Y., Al-Dhelaan, A., and Al-Dhelaan, M. A time-series based aggregation scheme for topic detection in weibo short texts. *Physica A: Statistical Mechanics and its Applications*, 536:120972, 2019.

Mehrotra, R., Sanner, S., Buntine, W., and Xie, L. Improving lda topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pp. 889–892, 2013.

Müller, A. C. and Guido, S. *Introduction to machine learning with Python: a guide for data scientists*. " O'Reilly Media, Inc.", 2016.

Ollagnier, A. and Williams, H. Network-based pooling for topic modeling on microblog content. In *International Symposium on String Processing and Information Retrieval*, pp. 80–87. Springer, 2019.

Quezada, M. and Poblete, B. A lightweight representation of news events on social media. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1049–1052, 2019.

Schütze, H., Manning, C. D., and Raghavan, P. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.