Improving LDA Topic Models for Microblogs via Automatic Tweet Labeling and Pooling

Author 1
Affiliation 1
Address 1
Country 1
email1

ABSTRACT

Twitter : the world of 140 characters poses serious challenges to the efficacy of topic models on short, messy text. While topic models such as Latent Dirichlet Allocation (LDA) have a long history of successful application to news articles and academic abstracts, they are often less coherent when applied to microblog content like Twitter. In this paper, we investigate methods to improve topics learned from Twitter content *without* modifying the basic machinery of LDA; we achieve this through various pooling schemes that aggregate tweets in a data preprocessing step for LDA. We empirically establish that coherence across three diverse Twitter datasets in comparison to an unmodified LDA baseline and a variety of pooling schemes.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous ; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Theory

Keywords

ACM proceedings, LaTeX, text tagging

1. INTRODUCTION

The "undirected informational" search task, where people seek to better understand the information available in document corpora, uses techniques such as multidocument summarisation and topic modeling. Topic models uncover the salient patterns of a collection under the mixed-membership assumption: each document can exhibit multiple patterns to different extents. When analysing text, these patterns are represented as distributions over words, called *topics*. Probabilistic topic models such as Latent Dirichlet Allocation (LDA) [1] are a class of Bayesian latent variable models that have been adapted to model a diverse range of document genres.

Table 1: Sample Topic Words

Poor Topics	Coherent Topics
barack cool apple health iphone	flu swine news pandemic health
los barackobama video uma gop	death flight h1n1 vaccine confirmed

To address the undirected informational task arising for the exploration of Twitter content, we propose the use of popular topic models like LDA. However, Twitter content poses unique challenges different to much of standard NLP content: (1) posts are short (140 characters or less), (2) mixed with contextual clues such as URLs, tags, and Twitter names, and (3) use informal language with misspelling, acronyms and nonstandard abbreviations (e.g. O o haha wow). Hence, effectively modeling content on Twitter requires techniques that can readily adapt to this unwieldy data while requiring little supervision.

Unfortunately, it has been found that topic modeling techniques like LDA do *not* work well with the messy form of Twitter content [16]. Topics learned from LDA are formally a multinomial distribution over words, and by convention the top-10 words are used to identify the subject area or give an interpretation of a topic. The naive application of LDA to Twitter content produces mostly incoherent topics — some are vaguely interpretable but contain unrelated words in the top-10 word set. For example, Table 1 demonstrates poor topic words as compared to topic words which are much more coherent and interpretable.

How can we extract better topics in microblogging environments with standard LDA without the need of any major modifications? While lingistic "cleaning" of text could help somewhat, for instance [3], a complementary approach using LDA is also needed because there are so few words in a tweet. An intuitive solution to this problem is tweet pooling [13, 4]: merging related tweets together and presenting them as a single document to the LDA model. In this paper we examine various tweet-pooling schemes the performance of these methods across different datasets; these are constructed so that they are representative of the diverse collections of content possible in the microblog environment.

Evaluation of the resultant topic model on this data is challenging because of the unsupervised nature of the problem. Hence, we look at a variety of topic coherence evaluation metrics including the ability of the topics obtained to reconstruct known clusters and the interpretability of topics via statistical information measures.

Given our diverse datasets and evaluation metrics, we wish to answer the following questions: (1) Do the different proposed pooling strategies perform better than unpooled tweets? (2) Which pooling scheme works best for which metric and why? In answering these questions, we make the following novel contributions. First, we show that different pooling schemes perform differently across different datasets and a few pooling schemes consistently outper-

form unpooled tweets. Second, we show that a novel *hashtag-based pooling* approach leads to best results across *all* evaluation metrics. The performance is a substantial improvement over unpooled use.

In the following sections we dicuss different tweet pooling schemes, our method for Twitter dataset construction, and evaluation metrics. We use the Mallet [6] implementation of LDA for performing our experiments. We then present our results and discuss related work.

2. TWEET POOLING FOR TOPIC MODELS

The goal of this paper is to obtain better LDA topics from Twitter content without modifying the basic machinery of standard LDA. As noted in Section 1, microblog messages differ from conventional text: message quality varies greatly, from newswire-like utterances to babble. To address these challenges with topic modelling, we present various pooling schemes to aggregate tweets together for use as training data for building better LDA models. The motivation behind tweet pooling is that individual tweets are very short($\leq$ 140 characters) and hence treating each tweet as an individual document does not present adequate term co-occurence data within documents. Aggregating tweets which are similar in some sense (semantically, temporally, etc.) enriches the content present in a single document from which the LDA can learn a better topic model. We next describe various tweet pooling schemes.

Basic scheme – Unpooled Tweets.

The default way of training models involves treating each tweet as a single document and training LDA on all tweets. This serves as our baseline and we compare results of other pooling schemes against this unpooled scheme.

Author-wise Pooling.

Pooling tweets according to author is the standard away of aggregating Twitter data to improve LDA topic modeling and has been previously proposed in the literature [13, 4] and shown to be superior to unpooled Tweets. To use this method, we train LDA on aggregated author profiles, each of which combines all tweets posted by the same author.

Burst-score wise Pooling.

A *trend* on Twitter [7] (sometimes referred to as a trending topic) consists of one or more terms and a time period, such that the volume of messages posted for the terms in the time period exceeds some expected level of activity. In order to identify trends in Twitter posts, "bursts" of interest and attention can be detected in the data. We run a simple burst detection algorithm to detect such trending topics and aggregate tweets containing those terms having high burst scores. To identify terms that appear more frequently than expected, we will assign a score to terms according to their deviation from an expected frequency. Assume that M is the set of all messages in our Tweets dataset, R is a set of one or more terms (potential trending topic) to which we wish to assign a score, and y represents a day of the total z days. We then define $M(R, y)$ as the set of every Twitter message in M such that (1) the message contains all the terms in R and (2) the message was posted during day y. With this information, we can compare the volume in a specific day to the other days. Let $Mean(R) = \Sigma_y M(R, y)/z$. Correspondingly, $SD(R)$ is the standard deviation of the number of messages with the terms in R posted over all the days. The *burst-score* is defined as:

$$burst\text{-}score(R, y) = \frac{|M(R, y) - Mean(r)|}{SD(R)}$$

Let us denote the terms having burst-scores greater than 5 as *burst-term*. Then our first novel aggregation method of burst-wise polling aggregates tweets for each burst-term into a single document trains LDA on the set of documents for each burst-term. The tweets which do not contain any of the burst-term were ignored for the purpose of training the LDA model. We then use the trained model to infer a topic mixture for each of the individual tweets. This scheme is henceforth referred to as Burst Score-wise pooling.

Temporal Pooling.

The fourth scheme and our second novel pooling proposal is known as Temporal Pooling, which utilises the temporal information of the tweets. It is noted that whenever a major event occurs, a large number of users often start tweeting about the event. Temporal pooling of tweets might help us in extracting useful information from the LDA topics. We train the LDA on an aggregate document of tweets posted within the same hour.

Hashtag-based Pooling.

A Twitter *hashtag* is a string of characters preceded by the hash (#) character. In many cases hashtags can be viewed as topical markers, an indication to the context of the tweet or as the core idea expressed in the tweet, therefore hashtags are adopted by other users that contribute similar content or express a related idea. A few examples of the use of hashtags are: "ask GAGA anything using the tag #GoogleGoesGaga for her interview! RT so every monster learns about it!! " referring to an exclusive interview for Google by Lady Gaga (singer), "Whoever said 'youth is wasted on the young' must be eating his words right now. #March15 #Jan25 #Feb14 ", referring to the protest movements in the Arab world. For the hashtag-based pooling scheme, for each hashtag we aggregate tweets containing this hashtag and train LDA on this collection. The tweet pool for each hashtag thus represents a document. If any tweet has more than one hashtag, this tweet gets added to the tweet-pool of each of those hashtags. This results in tweets with multiple hashtags being repeated across documents.

Other Pooling.

While a few other combinations of pooling schemes (eg.author-time, hashtag-time, *etc*) are possible, the initial results obtained were not as good as those presented for the currently outlined pooling schemes. This may be due to the lack of data in each finer-grained pool. Despite the initial negative results, these combinations of pooling schemes might be further explored in future work and may help unveil even finer-grained topics (i.e., short-term events centred on an author group or set of hashtags).

3. TWITTER DATASET CONSTRUCTION

The different pooling schemes and their proposed modifications result in different topic models, the evaluation of which is a major concern. We wish to answer questions like: Which scheme performs better on which aspects and on what kinds of data? Due to the large number of tweets ($\sim$360K) in any of the twitter specific datasets, manual labeling of topics is not feasible. To circumvent this problem of unsupervised evaluation we carefully construct our datasets keeping the following point in mind: The datasets should cover diverse collections of content, but also the known source of the content should help in evaluation of the different schemes.

We construct three datasets which we believe are representative of the diverse collections of content found on Twitter. We chose one or two term queries (often with similar pairs of queries to encourage a non-strongly diagonal confusion matrix) to search a tweet

Table 2: Datasets

Dataset	Term/%
Generic	music/17.9 business/15.8 movie/14.5 design/10.8 food/9.6 fun/9.1 health/6.9 family/6.4 sport/4.9 space/3.2
Specific	Obama/23.2 Sarkozy/0.4 baseball/3.5 cricket/1.8 Mcdonalds/1.5 Burgerking/0.5 Apple/16.3 Microsoft/6.8 United-states/40.7 France/4.9
Events	Flight-447/0.9 Jackson/13.9 Lakers/13.8 attack/13.8 scandal/4.1 swine-flu/13.8 recession/12.3 conference/14.1 T20/4.4 Iran-election/8.6

collection and each resulting set of tweets was labeled by the query that retrieved it. Since the number of queries (equivalently the number of clusters) is known beforehand, we could use this knowledge to evaluate how well the topics output by LDA match with known clusters. A brief description of the three datasets is as follows:

Generic Dataset: 359478 tweets from 11 Jan'09 to 30 Jan'09. A general dataset with tweets containing generic terms which represent a broader sense.

Specific Dataset: 214580 tweets from 11 Jan'09 to 30 Jan'09. A dataset composed of tweets which have specific terms which refer to named entity topics.

Event Dataset: 207128 tweets from 1 Jun'09 to 30 Jun'09. An event related dataset which contains tweets which were posted about some particular events. The query terms are terms which represent events.

Each of these datasets was created by querying a collection of 100 million tweets spanning two months (Jan'09 & Jun'09) with terms that relate to generic queries (broad topic words like music, business, *etc.*), specific queries (named entity topics like Obama, McDonalds, *etc.*) and event related queries (actual events in that timeframe like recession, Flight 447, Iran elections, *etc.*). Table 2 give the terms and the percentage tweets in the datasets which contain that term.

Note that the three datasets span three different scenarios in which tweets would be posted. Evaluating our methods on each of these would give us useful insights as to which methods work well for which type of data.

4. EVALUATION METRICS USED

Evaluation of the different topic models based on the features of coherence: topical consistency of documents assigned to a topic with high probability, or human interpretability of the most probable words for a topic are both important issues, but the unsupervised nature of topic models makes this difficult. For some applications there may be extrinsic tasks, such as information retrieval or document classification, for which performance can be evaluated. However, such tasks are not applicable for evaluating topics models in the *undirected informational task*.

We evaluate our topic models based on the following two general approaches to measuring topical coherence, as well as a pure probability approach [12].

Clustering-based metrics: We would like to measure the quality of topics found by the models, that the models can recall known existing topics in the data, and can consistently assign the right tweets to the right topics. Fortunately, each dataset we constructed is a class-labeled dataset containing ten categories. Since we know the ground truth label of all the tweets in the dataset (their categories), we can measure the quality by how likely the topics agree with the true category labels. To measure how well the topics produced by LDA reconstruct known clusters and how consistent they are, we use clustering-based measures of purity and normalized mutual information (NMI), both defined below.

Semantic coherence and interpretability: Learnt topics should be coherent and interpretable. Topic coherence meaning semantic coherence - is a human judged quality that depends on the semantics of the words, and cannot be measured by model-based statistical measures that treat the words as exchangeable tokens. It is possible to automatically measure topic coherence with near-human accuracy [10] using a score based on pointwise mutual information (PMI). We use this to measure coherence of the topics from different tweet-pooling schemes.

The scores used are as follows:

Purity.

To compute purity [5], each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and dividing by N. For each tweet d, we use the maximum value in topic mixture θ_d to determine its class/topic. As the number of correctly assigned tweets increases for each cluster, the overall purity score increases. hence high purity scores reflect better cluster reconstruction.

Normalized Mutual Information.

Since we know the ground truth label of all the tweets in the dataset, i.e., their categories, we can measure the quality by how likely the topics agree with the true category labels. But high agreement is easy to achieve when the number of clusters is large, thus one needs a divisor to discount for a large number of clusters. The resulting two-part score is:

$$NMI(T, G) = \frac{2I(T; G)}{H(T) + H(G)}$$

where $I(T, G)$ is Mutual Information and $H(T)$ gives the entropy. NMI [5] is always a number between 0 and 1. NMI score will be 1 if the clustering results exactly match the category labels while 0 if the two sets are independent. For each tweet d, we use the maximum value in topic mixture θ_d to determine its cluster.

Pointwise Mutual Information.

One of the goals of our work is to get topics that are more coherent. PMI is one measure of the statistical independence of observing two words in close proximity. We treat two words as co-occurring if both the words occur in the same tweet. For a topic t_k, we measure topic coherence as the average of PMI for the pairs of its top ten words $\{w_1, ..., w_{10}\}$.

$$PMI\ \ Score(t_k) = \frac{1}{100}\Sigma_{i=1}^{10}\Sigma_{j=1}^{10} PMI(w_i, w_j)\ ,$$

where the PMI of a given pair of words (w_i, w_j) is $PMI(w_i, w_j) = log\frac{p(w_i,w_j)}{p(w_i)p(w_j)}$. The average of the PMI score over all the 10 topics is used as the final measure of the PMI score. We use our own dataset to calculate the word probabilities used in this score (rather than using the Wikipedia corpus [10]).

Held Out Probability.

Another way of evaluating topic models is to compare predictive performance by estimating the probability of a subset of held-out documents. We used the Left to Right evaluation algorithm as described in [12] to calculate these values, which is an unbiased method. Another approach is the so-called document completion method [12], however with so few words we felt holding out a subset of a (small) document was ill-advised.

5. RESULTS FOR POOLING SCHEMES

In this section we discuss the results of the experimental evaluation of the tweet pooling schemes introduced in Section **??**. The datasets used were described in Section 3 while the evaluation metrics used were described in Section 4.

5.1 Document Characteristics

We first have a look at the document characteristics of the documents in the different pooling schemes for the three datasets. Characteristics like the number of documents affect LDA directly and hence it will be interesting to look at what the training data consists of. Table 3 presents the required statistics.

The statistics presented above highlight the differences in the characteristics of the documents on which LDA models have been trained. The number of documents decreases as we move from Unpooled scheme to Authorwise and Hashtagwise pooling scheme, while the corresponding size of the documents in each case increases. On an average the document size increases by a factor of seven in hashtag-based pooling when compared against unpooled or authorwise pooling schemes. Thus each document in hashtag-based pooling contains more content from which LDA could possibly extract latent semantics. On the other extreme lies the temporal pooling with very less number of documents and hence each document of a much larger size. Such large documents might impact the topic model in an unpleasant manner. These statistics highlights that hashtag-based pooling scheme lies mid-way between both the extremes (small documents in unpooled tweets vs large documents in temporal pooling) and hence suggests that hashtag-based pooling should perform optimally in comparison to other schemes.

5.2 Comparison of Pooling Schemes

For the three datasets (viz. Generic, Specific and Events) and pooling schemes, we next evaluate the Purity scores, NMI scores, PMI scores and the Held-Out probabilities in Table 4 on the topic model obtained by training LDA using each scheme.

Based on these results we conclude that hashtag-based pooling scheme *clearly* performs better than unpooled scheme as well as other pooling schemes.

6. RELATED WORK

Topic modeling is widely used in text mining communities with LDA being the benchmark. LDA has been extended in a variety of ways, and in particular for social networks and social media, a number of extensions to LDA have been proposed. For example, [9] proposed two methods to regularize the learning of topic models aimed at short text snippets. While the focus of this work was on blogs and search result snippets, it would be interesting to see how well they work on Twitter data. Also, the combination of the work proposed in [9] with the tweet pooling schemes we describe before could produce interesting results. For automatic hashtag labeling that proved crucial to improving topics in our hastag-based pooling model, [15] also uses tweet similarity as a criteria, but does not explore metrics based on inverse author frequency [2] that we found to offer the most robust performance across datasets and evaluation metrics. Additional features for hashtag assignment can be found in the comprehensive study [14] which can be leveraged in future extensions.

Our work is quite different from many pioneering studies on Twitter and topic modeling because we focus on how we could get better topic coherence over tweets with minimal modification to existing models. Prior work on topic modeling for tweets includes the work of [11] which presents a scalable implementation of a partially supervised learning model (Labeled LDA). [16] empirically compare the content of Twitter with a traditional news medium, New York Times, using unsupervised topic modeling. [4] use the topic modeling approach for predicting popular Twitter messages and classifying Twitter users and corresponding messages into topical categories. The TwitterRank system [13] and [4] uses author-based pooling to apply LDA to tweets. [16] compared topic characteristics between twitter and traditional news media; they propose to use one topic per tweet (similar to PLSA), and argues that this is better than no pooling, or the author-topic model. [8] used LDA for tweet retrieval. In addition, they used retweet as an indicator of "interestingness" to improve retrieval quality, which suggests additional features we could incorporate in future extensions to our pooling framework.

Our work is different from these in the sense that we provide a simple yet effective way which greatly improves the quality of topics obtained without making any major complicated modifications to standard LDA. The detailed experiments on a variety of datasets highlight our novel contribution of hashtag-based pooling

7. SUMMARY AND CONCLUSION

This paper presents a way of aggregating tweets in order to improve performance of topic models in terms of quality of topics obtained measures by the ability to reconstruct clusters and topic coherence. The results presented in Table 4 suggest that hashtag-based pooling outperforms all other pooling strategies including the default way of training topic models on Twitter data (unpooled).

Across diverse datasets and various topic coherence metrics, pooling schemes. This indicates the promise of this novel automatic hashtag

8. REFERENCES

[1] D. Blei, A. Ng, and M. Jordon. Latent Dirichlet allocation. volume 3, pages 993–1022, 2003.

[2] S. E. Chan, R. K. Pon, and A. F. Cárdenas. Visualization and clustering of author social networks. In *International Conference on Distributed Multimedia Systems Workshop on Visual Languages and Computing in Grand Canyon*, pages 30–31, Arizona, USA, 2006.

[3] B. Han, P. Cook, and T. Baldwin. Automatically constructing a normalisation dictionary for microblogs. In *Proc. of EMNLP-CoNLL 2012*, pages 421–432, Korea, 2012.

[4] L. Hong and B. Davison. Empirical study of topic modeling in Twitter. In proceedings 1st ACM Workshop on Social Media Analytics, 2010.

[5] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[6] A. McCallum. Mallet: A machine learning for language toolkit. 2002.

[7] M. Naaman, H. Becker, and L. Gravano. Hip and trendy: Characterizing emerging trends on Twitter. In proceedings Journal of the American Society for Information Science and Technology (JASIST), 62(5). doi: 10.1002/asi.21489, 2011.

[8] N. Naveed, T. Gottron, J. Kunegis, and A. C. Alhadi. Searching microblogs: coping with sparsity and document quality. In *Proceedings of CIKM '11*, pages 183–188, New York, NY, USA, 2011. ACM.

[9] D. Newman, E. Bonilla, and W. Buntine. Improving topic coherence with regularized topic models. In proceedings of NIPS, 2011.

[10] D. Newman, J. Lau, K. Grieser, and T. Baldwin. Automatic evaluation of topic coherence. In proceedings NAACL, 2010.

[11] D. Ramage, S. Dumais, and D. Liebling. Characterizing

Table 3: Document Characteristics for different schemes

Pooling Scheme	#of docs			Avg # of words/doc			Max # of words/doc		
	Generic	Specific	Events	Generic	Specific	Events	Generic	Specific	Events
Authorwise	208300	118133	67387	17.6	20.4	15.4	4893	3586	2775
Unpooled	359478	214580	207128	10.2	10.9	9.7	35	49	32
Burst Score	7658	7436	5434	76.5	154.2	71.6	61918	420249	57794
Hourly	465	464	463	8493.4	5387.5	2422	20144	18869	38893
Hashtag	8535	7029	4099	70.4	187.2	78.4	61918	420249	57794

Table 4: Results of different pooling schemes

Scheme	Purity			NMI Score			PMI score			Held Out Probability		
	Generic	Specific	Events	Generic	Specific	Events	Generic	Specific	Events	Generic	Specific	Events
Unpooled	0.49 ± 0.08	0.64 ± 0.07	0.69 ± 0.09	0.28 ± 0.04	0.22 ± 0.05	0.39 ± 0.07	-1.27 ± 0.11	0.47 ± 0.12	0.47 ± 0.13	-82.2 ± 6.3	-89.3 ± 7.2	-86.3 ± 7.4
Author	0.54 ± 0.04	0.62 ± 0.05	0.60 ± 0.06	0.24 ± 0.04	0.17 ± 0.04	0.41 ± 0.06	0.21 ± 0.09	0.79 ± 0.15	0.51 ± 0.13	-63.0 ± 4.3	-68.6 ± 4.7	-66.4 ± 5.2
Hourly	0.45 ± 0.05	0.61 ± 0.06	0.61 ± 0.07	0.07 ± 0.04	0.09 ± 0.04	0.32 ± 0.05	-1.31 ± 0.12	0.87 ± 0.16	0.22 ± 0.14	-64.8 ± 6.2	-69.4 ± 5.8	-67.9 ± 7.1
Burstwise	0.42 ± 0.07	0.60 ± 0.04	0.64 ± 0.06	0.18 ± 0.05	0.16 ± 0.04	0.33 ± 0.04	0.48 ± 0.16	0.74 ± 0.14	0.58 ± 0.16	-56.7 ± 5.5	-59.0 ± 4.5	-57.8 ± 6.1
Hashtag	$\mathbf{0.54 \pm 0.04}$	$\mathbf{0.68 \pm 0.03}$	$\mathbf{0.71 \pm 0.04}$	$\mathbf{0.28 \pm 0.04}$	$\mathbf{0.23 \pm 0.03}$	$\mathbf{0.42 \pm 0.05}$	$\mathbf{0.78 \pm 0.15}$	$\mathbf{1.43 \pm 0.14}$	$\mathbf{1.07 \pm 0.17}$	**-55.9 ± 4.3**	**-58.9 ± 4.1**	**-55.4 ± 4.3**

microblogs with topic models. In proceedings AAAI Conference on Weblogs and Social Media, 2010.

[12] H. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In proceedings ICML, 2009.

[13] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings WSDM '10*, pages 261–270, New York, NY, USA, 2010. ACM.

[14] L. Yang, T. Sun, M. Zhang, and Q. Mei. We know what @you #tag: does the dual role affect hashtag adoption? WWW '12, pages 261–270, 2012.

[15] E. Zangerle, W. Gassler, and G. Specht. Recommending#-tags in twitter. In *Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011).*, volume 730, pages 67–78, 2011.

[16] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li. Comparing twitter and traditional media using topic models. In *Proceedings of ECIR'11*, pages 338–349, Berlin, Heidelberg, 2011. Springer-Verlag.