Latent Dirichlet Allocation as a Twitter Hashtag Recommender System [*]

Brian Gillespie
Northeastern University
Seattle, WA
bng1290@gmail.com

Shailly Saxena
Northeastern University
Seattle, WA
saxena.sha@husky.neu.edu

ABSTRACT

In this paper, we investigate an unsupervised approach to hashtag recommendation, to aid in the classification of tweets and to improve the usability of Twitter. Two corpora were generated from a sample of tweets collected from September 2009 to January 2010. The first corpus was composed simply of individual tweets, and the second corpus was made by aggregating tweets by UserID into what is referred to as the USER PROFILE model. Latent Dirichlet Allocation(LDA) was used to generate a topic model for each corpus, and then this model was queried to to generate topic distributions for new test tweets. By sampling a topic from each test tweet, and then sampling one of the top terms from that topic, a set of words can be selected as recommended hashtags. This recommendation process was applied to several example tweets, and the relevancy of the suggested hashtags were evaluated by human observers. The results of this would be briefly mentioned here if there were any.

Keywords

Twitter, LDA, topic modeling, NLP, social media

1. INTRODUCTION

Twitter is an amazing source of text data, both in content and quantity. With over 305 million monthly active users across the globe, there are a wide variety of subjects being discussed. Cataloging this data, and finding ways to make it more search-able is very desirable, as this data can be an effective resource for business knowledge and studying social trends. Hashtags appear as the natural categorization index for tweets, however since only 8% of tweets contain hashtags, they cannot be used as a direct categorization for all tweets. Compounding this issue, there are no prescribed hashtags; any sequence of characters can be a hashtag as long as it has a # in front of it. A hashtag recommendation system can be implemented to encourage users to utilize more hashtags, and to provide a more consolidated hashtag base for tweet categorization. But in order to avoid prescribing hashtags, and thus limiting the free expression of Twitter users, it is more reasonable to generate the suggested hashtags from the content of Twitter itself. This has the added gain of providing an adaptable system that can develop alongside the vocabulary and interests of the Twitter user base.

2. RELATED WORK

Although there has been much research in the structure and dynamics of microblogging networks, and many insights have been gained by looking into how users interact with one another and with the world, the area of hashtag recommendations is still not explored enough. Providing hashtags is an important feature of Twitter for they aid in classification and easy retrieval of tweets and this problem has been addressed in a number of ways so far.

There have been recent developments in hashtag recommendation approaches to Twitter data such as Zangerle et al. [10] and Kywe, Hoang, et al. [6] where recommendations are made on the basis of similarity. These similarities can be of tweet content or users. [10] analyzed three different ways of tweet recommendations. They rank the hashtags based on the overall popularity of the tweet, the popularity within the most similar tweets, and the most similar tweets. Out of these, the third approach was found to out-perform the rest. On the other hand, Mazzia and Juett [8] use a naive Bayes model to determine the relevance of hashtags to an individual tweet. Since these methods do not take into consideration, the personal preferences of the user, Kywe et al. [6] suggest recommending hashtags incorporating similar users along with similar tweets and thus achieving more than 20% accuracy than the above stated methods.

One of the drawbacks of these approaches is that they rely on existing hashtags to recommend the new ones. Since, the current number of hashtags is already very low and very few of them are hardly ever repeated, using these hashtags will not improve classification of tweets. [2] Godin et. al took these facts into consideration and applied LDA model which was trained to cluster tweets in a number of topics from which keywords can be suggested for new tweets. In this paper, we extend the work done by Godin et al.[2] by incorporating *user profile* approach as suggested in Hong et

[*] There are no legal restrictions! Break all the rules! We don't own this, and wouldn't even dare claim it if our lives depended on it. Plagiarize to your heart's desire, but maybe hook us up with a job interview if you think we had some good ideas in here.

Table 1: Twitter Sample Sep 2009 to Jan 2010

UserID	Tweet
66	For all the everything you've brought to my internet life #mathowielove
53717680	What's good Tweeps this Friday
16729208	"I bet you all thought Bert & Ernie were just roommates!"
86114354	@miakhyrra take me with you! lol

al. [5]. In this approach, tweets were aggregated by the User IDs of their authors, and a new set of documents was created where each document is a *user profile* of a unique user id and his or her combined set of tweets.

Hence, we explore some of the more promising approaches to hashtags recommendations in tweets, and investigate their implications to further research into developing a more successful Twitter hashtags recommendations using topic modeling. Due to the success of the *user profile* approach, we will employ and evaluate this method, to further investigate its effectiveness.

3. DATASET AND PREPROCESSING

The datasets used were obtained from Cheng, Caverlee, and Lee [9]. In order to reduce noise in the text data, a set of stop words were removed from the tweet bodies, the remaining words were then stemmed, and any hyperlinks were also removed. A bag of words model was created, and TF-IDF vectors were generated. For the *user profile* LDA approach [5] and the Twitter-LDA model [11], the vectors were aggregated based on user IDs. We also processed each tweet for the documents "as-is" for the Twitter-LDA model. The training and tests datasets were merged, and then repartitioned to prepare for 10-fold cross-validation.

3.1 The Dataset

The training dataset from Cheng, Caverlee, and Lee [9], contains 3,844,612 tweets from 115,886 Twitter users over the time period of September 2009 to January 2010. The test set, also from [9], contains 5,156,047 tweets from 5,136 Twitter users over the same time period. In general, tweets contain between 1 and 20 words, with an average of 7 words per tweet. A smaller proportion of the tweets contained more than 20 words as seen in Figure 1. Each line of the dataset contains a unique user ID and tweet ID, text content, and a timestamp. The text content of each tweet is limited to 140 characters, and can contain references to other users of the form @username, as well as popular #hashtags. Many tweets also include hyperlinks which are often passed through URL shorteners (e.g. http://goo.gl/uLLAe). The implications of these more anomalous text instances are considered in the next section.

An additional document corpus was generated from each dataset by aggregating tweets by user ID. This reduction is referred to as the *user profile* model [5]. After aggregation, the corpus contained a total of 115,886 documents. Data preprocessing was performed on both of these corpora, and is described in the next section.

3.2 Data Preprocessing and Reduction

Due to the unconventional vocabulary in tweets and the large amount of noise inherent in such a diverse set of text, we decided to perform several data preprocessing steps. A regular expression was used to remove any non-latin characters and any URL links, as URL links are generated randomly and cannot be easily related to a topic. We also remove all numbers, punctuations, retweets and words starting from character @. After pruning the tweet contents, the text was tokenized, stop words were removed, and the remaining tokens were stemmed using the Porter Stemming library from the Natural Language Toolkit (NLTK). The prepared corpus of tweets was then converted to a set of Term Frequency (TF) vectors, using the corpora library from gensim. We noticed that many tweets contained excessive repetition of words (for instance one tweet read: "@jester eh eh eh eh eh eh eh eh eh..."), so in order to reduce any bias towards overused words in the data we removed words appearing over 5 times in a tweet, as well as using the TF-IDF model to reduce emphasis on such words. To reduce bias to common words across the corpus, we removed terms appearing in over 70% of the documents as per [11]. As seen in Figure 2, before preprocessing the data was dominated by common words that do not confer much meaning (e.g get, u, just, one). After preprocessing, however, we begin to see more meaningful words (e.g people, twitter, home, blog in Figure 3). After preprocessing, there were 26,542,006 total words and 184,002 unique words in the vocabulary.

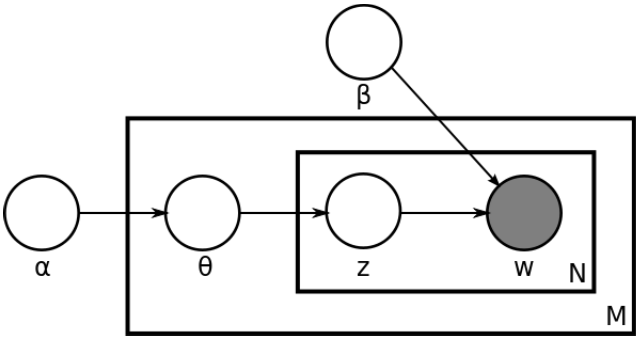

Figure 1: Plate Diagram of LDA.

4. MODEL AND METHODOLOGY

Latent Dirichlet Allocation, first proposed by Blei, Ng, and Jordan [1] is a generative probabilistic model, that supposes each document in a corpus is generated from a smaller, hidden set of topics. Each word in a document is drawn from a topic, making every document a mixture of samples from a variety of different topics. The plate diagram in Figure 1 demonstrates the generative story for a given word in this model. Here, the outer plate represents the corpus level where M is the number of documents, and the inner plate represents the document level where N is the number of words in a given document. So working from the outside-in, α is a parameter to the Dirichlet prior which defines the topic distribution for a set of documents. This informs θ, the topic distribution for a given document, and this supplies us with a topic, z. Ultimately, each word in each document will be drawn from one such topic, which is itself a distribution of words with a Dirichlet prior parameterized by β.

There are multiple ways of estimating these hidden topics, including Hidden Markov Monte Carlo techniques such as Collapsed Gibbs Sampling [3], or with a stochastic gradient descent method such as in [4]. In this paper we utilized the latter, and the details of our topic model generation are discussed in the next section.

For the purpose of hashtag recommendation, we first need to see which topics are most represented by a new tweet. We then propose that the most significant words from within those strongly related topics will provide a more general representation of the message contained in a given tweet. So essentially for each new test tweet, we sample from the top topics in that tweet, and we recommend a hashtag from the top most representative words in that topic. In this way, we are recommending topics that also reflect the social component of Twitter. Since the topics we find are driven by the tweets themselves, our recommendations serve as an approximation of the connections between what users are talking about.

4.1 Getting a Topic Model

In order to generate our topic model, we utilized the LDAModel package from gensim. This package uses the *onlineldavb.py* algorithm that was proposed in [4]. This is an online, constant memory implementation of the LDA algorithm, which allowed us to process a rather large dataset with limited resources. This package processes the documents in mini batches, and we chose our batch size to be the default, 2,000. The key parameters κ and τ as mentioned in [4] were kept at 0.5 and 1.0, respectively, as these values performed best in their performance analyses. Symmetric priors were also kept for the topic-to-word and document-to-topic distributions; these priors can be tweaked in order to boost certain topics or terms. Finally, our choices for number of topics were chosen based on the more successful results seen in [5]. For the per tweet based corpus we chose topic sizes of 50, 100, and 200 in keeping with [2]. However for the *user profile* model we turned to the experiments by Hong, et. al., who determined that this approach tends to favor lower topic sizes, and in their case a size of 40 performed best. For our own *user profile* implementation we used topic sizes of 20, 40, 70, and 100.

Since several models were created, it is informative to see if our various approaches have had much of an effect on the final model. In order to compare the similarity of our topic distributions, we calculated the Hellinger Distance between the topic distributions generated by our LDA models for each topic size and training corpus. For two distributions, P and Q, each of size k, the Hellinger Distance is given by

$$H(P,Q) = \sqrt{\frac{1}{2}\sum_{i=1}^{k}(\sqrt{P}-\sqrt{Q})^2}$$

. The Hellinger Distance is a distance metric that can be used to evaluate the similarity of two probability distributions, and in this case we are comparing the word distributions of the topics generated by our various LDA models. Of particular interest were the Hellinger Distance between the various *user profile* corpora and the single-tweet corpora. In Figure <FIG HERE> we show how the average, minimum, and maximum Hellinger Distances change between our two models change with increasing topic sizes. It is interesting to note that... Additionally, in Table <TABLE> we can see two of the more similar topics generated from each model. Talk about these a bit...

4.2 Hashtag Recommendation

In order to provide recommendations for a new tweet, we first decided to do some quick preliminary preprocessing by removing any user-added hashtags, references to other users, URL links, and non-latin characters. From here, we can convert this new document into its TF-IDF representation and merge this into our existing dictionary. We used the LDA model generated previously to infer the topics from which each word in the tweet is drawn. This can also be done using gensim; by querying the generated model, LDAModel will return the distribution of topics and their respective posterior probabilities for a given document. Using the belief that each document represents some mixture of topics, we can sample the most strongly correlated topics from a new tweet. Next, we select one of these strong topics, and recommend a hashtag from the set of words that make up that topic. In this experiment, we decided to select only from the words that most strongly represent their topics, in the hopes that this will strengthen the relevance of our suggestions.

5. EVALUATION

In order to evaluate the quality of recommendations from our model, we decided to make a simple web app, where a user can write out a tweet and the number of hashtags they'd like, and see our recommendations. We intend to provide this setup to several human testers, who will evaluate whether a given hashtag is relevant to their submission or not.

6. RESULTS AND DISCUSSION

For the evaluation experiments, we performed ten trials, each trial attempting to recommend a number of hashtags ranging from one to ten. After testing with <Y> human subjects, we tallied up the frequency of positive cases for each trial. The resuslts will be available in a figure.

7. REFERENCES

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[2] Fréderic Godin, Viktor Slavkovikj, Wesley De Neve, Benjamin Schrauwen, and Rik Van de Walle. Using topic models for twitter hashtag recommendation. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 593–596. International World Wide Web Conferences Steering Committee, 2013.

[3] Tom Griffiths. Gibbs sampling in the generative model of latent dirichlet allocation. 2002.

[4] Matthew Hoffman, Francis R Bach, and David M Blei. Online learning for latent dirichlet allocation. In *advances in neural information processing systems*, pages 856–864, 2010.

[5] Liangjie Hong and Brian D Davison. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88. ACM, 2010.

[6] Su Mon Kywe, Tuan-Anh Hoang, Ee-Peng Lim, and Feida Zhu. On recommending hashtags in twitter networks. In *Social Informatics*, pages 337–350. Springer, 2012.

[7] Tianxi Li, Yu Wu, and Yu Zhang. Twitter hash tag prediction algorithm. In *ICOMPâĂŹ11-The 2011 International Conference on Internet Computing*, 2011.

[8] Allie Mazzia and James Juett. Suggesting hashtags on twitter. *EECS 545m, Machine Learning, Computer Science and Engineering, University of Michigan*, 2009.

[9] J. Caverlee Z. Cheng and K. Lee. You are where you tweet: A content-based approach to geo-locating twitter users. In *Proceeding of the 19th ACM Conference on Information and Knowledge Management (CIKM)*, October 2010.

[10] Eva Zangerle, Wolfgang Gassler, and Gunther Specht. Recommending#-tags in twitter. In *Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011). CEUR Workshop Proceedings*, volume 730, pages 67–78, 2011.

[11] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval*, pages 338–349. Springer, 2011.