Generating User Review Scores from Twitter

Ken Mansfield
University of British Columbia
Department of Electrical and Computer Engineering
kmansfield@ece.ubc.ca

Abstract

The goal of this paper is to investigate the use of different Topic Modelling techniques to accurately label Twitter data about a specific item for the purpose of generating an aggregated user review. The latent topics will be discovered using two separate techniques, Latent Dirichlet Allocation and Dirichlet Multinomial Mixture. Once informative labels have been created, the documents belonging to those topics can be scored using Sentiment Analysis. Corpora will be generated using the Twitter Streaming API filtered by an item we wish to research. Short text produced by character limited tweets tend to be very noisy and data sparsity becomes a serious problem. Techniques will be discussed to overcome some of these issues.

1 Description

1.1 Topic Modelling Short Text

Topic Modelling is a very powerful way of summarizing large amounts of data and finding interesting patterns. One of the most powerful topic modelling algorithms is Latent Dirichlet Allocation. LDA uses statistical analysis to automatically discover previously unknown topics from input text.

The user first determines the number (N) of topics that they wish to discover. The algorithm starts by randomly assigning each word in the document to one of the N topics. It then learns the topics by iterating over each word in the document and assigning each document a topic distribution: $P(z_{di}|Z_{di})$ where z_{di} denotes the topic, and Z_{di} denotes the topic assignments of all the other words in the document collection.

There have been many papers written about applying LDA to microblogs such as Twitter with varying success. Most papers involve some sort of aggregation strategy to overcome the issue of sparsity and noise in the text. One of the most common methods is to apply the author-topic idiom and aggregate all tweets from an individual user. This approach overcomes the problem of sparsity, however, it would not be useful for this project since the goal is to place every tweet into an individual topic. Another option is to train the model on more coherent external data and then apply the individual tweets on the trained model. This approach may be suitable, but would require an external corpus of data to provide training, which may defeat the purpose of using real-time streaming Twitter data as a source.

Other models such as word co-occurrence network based models, Bi-Term topic models, and supervised/semi-supervised LDA have been proposed as effective alternatives. Another option, the one-topic-per-document Dirichlet Multinomial Mixture (DMM) model has been proposed as a good method for analyzing short text such as tweets. An individual tweet is unlikely to discuss more than a single topic so it might benefit from this model.

For this paper, two different topic modelling methods, Latent Dirichlet Allocation and the Dirichlet Multinomial Model, will be compared and contrasted for their abilities to model short text.

1.2 Corpus

Twitter is one of the most popular sources of data for language processing and machine learning because it is free, easy to obtain, and it is being generated enmasse in real-time. On average, 600 tweets are produced every second. Using the Twitter API, we can make REST API requests to retrieve recent tweets, or stream live tweets.

Processing Twitter data does have its own challenges due to the short length nature of tweets, and the slang and abbreviations that is present. Research has shown that LDA and similar techniques have worked well on short documents such as tweets.

1.3 Generating User Review Scores

The final goal is to produce a score for each topic produced by the topic model with the hope that it resembles a review scored by ratings. Rating categories/aspects will be labeled by the latent topics that are discovered. The ratings will come from scores generated by performing sentiment analysis on each individual tweet, aggregated by the main topic that they belong to. The top 3 words for each topic will then become the labels for each rating category/aspect.

2 Related Work

2.1 LDA on Normal Text

Probabilistic topic models are frequently used to analyze the content of documents and the meaning of words contained within them. Latent Dirichlet allocation (D. Blei, 2003, 2012) is an unsupervised machine learning technique that is used to identify latent topic information from large document collections. Using a bag of words assumption, it treats each document as a vector of word counts and represents each document as a probability distribution over some topics as well as representing each topic as a probability distribution over a number of words. LDA is a generative model that uses a dirichlet prior on the topic model that can be used on unseen data, and benefits from the fact that the parameters of the model do not grow with the size of training corpus.

LDA has typically been used on corpora containing documents which are generally at least a few hundred words. Micro-blogging has quickly become a dominant form of electronic communication popularized by services such as Twitter and SMS messaging. Short texts, such as the 140 character limit imposed by Twitter, have posed significant challenges to the LDA model. To work around this, many alternative approaches have been considered. TwitterRank (J. Weng et al, 2010) addressed the problem of identifying influential users on Twitter using a modified PageRank algorithm which uses LDA for inducing topics on user aggregated messages. It treats a document as a collection of tweets from a single user similar to the author-topic model. This method overcomes the issue of short text documents by creating longer documents aggregated by many smaller pieces of text.

2.2 Dirichlet Multinomial Mixture

A larger document may consist of many dozens or even hundreds of topics, changing topics each paragraph, or even as frequent as every sentence. Short text, on the other hand, is unlikely to contain very many topics. Another approach for handling short text is to assume that there is only one topic per document. In the Dirichlet Multinomial Mixture Model (J. Yin et al. 2013, D. Nguyen et al. 2015), each document is treated as having only one topic. The process of generating a document in the collection is to first select a topic assignment for the document, and then the topic to word Dirichlet multinomial component generates all the words in the document from the same topic. More recently (J. Yin, 2014) introduced a collapsed Gibbs sampling algorithm for the DMM model in which a topic is sampled over all the documents in the collection.

2.3 Bi-Term Topic Model

Bi-Term Topic modelling is another method that has been proposed for improving models generated for short texts. Using BTM, the model learns the topics by directly modelling the generation of word co-occurrence patterns (bi-terms) in the entire corpus. Topics can be thought of as groups of correlated words and the correlation is revealed by word co-occurrence patterns in documents. BTM explicitly models the word co-occurrence patterns in the documents. BTM solves the problem of sparse patterns in individual short documents by using the richer global word co-occurrence patterns to discover the topics better. Experimental results have shown that BTM can learn better quality topics than other state-of-the-art models

2.4 Sentiment Analysis

Social media has provided researchers a wealth of information for many areas of research due to the availability of massive amounts of data written in natural language. Sentiment analysis can then be performed on that data to determine the opinions of various subjects contained within the text. This information has become an invaluable tool for researchers and businesses alike. For example mining the sentiment of a certain product can help businesses to determine a market strategy. Mining the sentiment of a company can help investors determine whether they should buy or sell that stock. Sentiment analysis has also famously been used for tracking and predicting

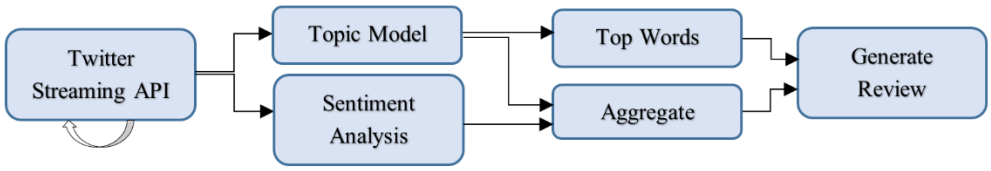

Figure 1: Data Pipeline

the success of election campaigns. Data scientists in particular have gained a particular interest in sentiment analysis using machine learning for predictive analysis as well as researching different ways of visualizing the data.

Twitter is a popular social media service where people write short tweets about all types of subjects including their opinions. There are on average 6000 tweets produced every second world-wide. With this amount of information being produced in real time, we can easily use this data to gauge sentiment on a massive variety of topics.

The simplest algorithm for generating a sentiment score involves using a dictionary of words which each have a score indicating their sentiment value. The word hate, for example might have a score of -5 and the word love might have a score of +5. The text is then searched for each one of the words in the dictionary and the accumulated score becomes the final sentiment score. This simple algorithm is a naïve first approach that can fail when it comes to certain phrases or slang or more complex ideas such as sarcasm.

There exists a great deal of research into sentiment analysis using Natural Language Processing and Machine Learning which can increase the success of sentiment classification. One such approach uses a *Sentiment Treebank* (R. Socher, 2013) to aid the process. It contains fine grained sentiment labels for 215,154 phrases in the parse trees of 11,855 sentences used for training their *Recursive Neural Tensor Network*. This RNTN takes phrases of any length as input and then represents a phrase through word vectors and a parse tree. It then computes vectors for higher nodes in the tree using the same tensor-based composition function. The combination of the Sentiment Treebank and the RNTN has propelled their results to show an improvement over the previous state of the art sentiment analysis techniques by up to 85.4%.

2.5 Multi-Aspect Sentiment Analysis with Topic Models

Work has been done to investigate the ability of topic models and sentiment analysis to perform multi-aspect rating prediction. (B. Lu et al, 2014) investigated the use of weakly supervised LDA to seed *aspects* as topics from Yelp and TripAdvisor reviews. Sentiment analysis was then performed on each of the documents in which the topics were contained. Their results have shown that weak supervision performs well and effectively aids multi-aspect rating prediction.

3 Implementation

3.1 Twitter Streaming API

Twitter provides developers an API with various methods to interact with its data. The three main API's to interact with Twitter data are:

- Twitter Search API: Used for conducting singular searches, reading user profile information and posting tweets.
- Twitter Streaming API: For streaming of a rate-limited sample of real-time tweets.
- Twitter Firehose API: For streaming all tweets in real-time.

The Firehose API is the ideal tool for businesses and researchers to analyze twitter data but its high cost and huge bandwidth requirements make it difficult for everyone to afford. For this paper the Twitter Streaming API will be used. The Twitter API is known to return less than one percent of all tweets. In my tests the Twitter API is able to return around 100 tweets per second using a bandwidth of around 400 kilobytes per second as each Tweet is around 4kB with all its metadata. The Twitter4j Java package is used as a wrapper to interact with the streaming API.

To generate a review based on a specific item or subject, a filter is used to ensure that we only receive tweets containing that word or words. The filter is also set to specify tweets in English only. The streaming API listener is set to close after the specified number of tweets are received to build the corpus, which is set to 4000 by default.

3.2 Text Pre-Processing

Since LDA and DMM both use a bag of words approach that does not need to look at complete grammar, unnecessary words can be removed. Stop words are words that can occur with a very high frequency within the text, yet they add little insight into the corpus if they are selected as a topic so it is necessary to remove them. To avoid being selected as topics, all punctuation is removed as well. URL's occur very frequently in tweets as users often reference web sites. These are unlikely to provide any insightful information about topics so these are removed.

Despite all the pre-processing done above, tweets still remain extremely noisy. They are full of emoticons, numbers and other symbols. To address this, all non-alphabetic characters are removed, and all words less than 3 characters are removed. The resulting text is often very short as some tweets may only contain URLs or irrelevant symbols, so text less than 4 words are removed. All retweets (RT) are removed as well because they are a huge source of duplicates.

Due to the 140 character limit in tweets, abbreviations (or common misspellings) are used with great frequency. These words carry the same meaning as their expanded versions, however, they will get treated as different words by the topic model. To accommodate this, an option would be to translate abbreviations and slang, as well as correct misspelling. This would increase the frequency of words that have different representations. Another frequently used technique in NLP, stemming and lemmatization, can be used to standardize words. This would also increase the frequency of words by replacing words with their common endings. The methods described above, replacing slang and abbreviations, stemming and lemmatization, would have a strong effect on a small corpus where each word may have a stronger effect on the overall outcome of the topic distributions, however, on a large enough corpus it is hoped that similar words would converge into the same topics. These techniques applied on a large corpus would also greatly increase the processing time. For these reasons, they will not be performed, but should be investigated in the future as a comparison.

3.3 LDA

Once the corpus has been assembled from collecting tweets from the streaming API, topic modelling can begin. To do this, LDA is run in order to discover the topics and the top words associated with each topic. Determining the ideal number of topics to choose for the LDA model is one of the most contentious issues in topic modelling. 1 topic per every 10 documents is a good estimate for regular sized documents, however, given that short texts discuss fewer topics, a ratio of 1:80 will be used, i.e. 50 topics for a corpus of 4,000 tweets. A large number of iterations, 2000-3000, are chosen to ensure that the algorithm has sufficient chance to converge.

Many software packages are available for performing LDA. MALLET (A. McCallum, 2002), the Machine Learning for Language Toolkit developed at the University of Massachusetts, will be used. The resulting model contains several important pieces of information including the distribution of topics over documents, and the distribution of words over topics. With this information, we can determine the top topic that each document belongs to, as well as the top words for each topic.

3.4 DMM

DMM provides an alternative approach to perform topic modelling that has been met with great success on short texts. The input parameters to DMM remain roughly the same as LDA: the corpus, # of topics, and # of iterations. To perform DMM, the jLDADMM (D. Nguyen, 2015) package developed at Macquarie University will be used.

3.5 Sentiment Analysis

To be able to aggregate sentiments based on topics, sentiment analysis can be applied to each of the documents. The Sentiment Analysis module contained within the Stanford Core NLP package is used. The benefit of using this package is that the model is already pre-trained and can be applied directly on the data.

3.6 Generating Reviews

Once the scores have been applied to each document, for each topic the scores of all documents who belong to that topic can be averaged to produce an overall score for that topic. The top words of each topic can be used as the category for each review.

4 Results

4.1 Topic Discovery

Topic modelling techniques such as LDA and DMM are powerful tools for discovering latent topics from a large collection of documents. However, in the case of Twitter text, the topics do not always conform

to the typical evaluation categories one might expect in a review. For example, with "iPhone" selected as the item to generate the review for, one of the latent topics discovered was "giveaway." There were multiple tweets that were received that discussed a contest in which iPhones were being given away as prizes. This is a coherent topic choice, however, it does not resemble a category that would more likely be encountered in a cellular phone product review, such as "battery life," or "performance." Other work (S. Rogers et al, 2013, B. Lu et al, 2011) on extracting topics for rating predictions generate their data from corpora derived from review sites such as Yelp or TripAdvisor. Since the text from these websites are already in the form of a review, it is much more likely that the topics discussed within such text are far more likely to be relevant choices as rating categories.

Research such as TwitterRank (J. Weng, 2010) focused their efforts on aggregating tweets based on individual users. This approach has the benefit of being able to focus on users who are more likely to produce coherently written text, such as academic users. As a comparative test, a review was generated based on search item "Brexit", a current trending political topic. The tweets about this subject (figure 2) tended to be much more coherent than text about "iPhone" (figure 3), which was full of advertising, much of which is likely generated by bots.

Subjectively, it appears that the DMM models produce more coherent topics. Based on results obtained by (J. Yin et al, 2014), quantitative results such as document cluster evaluation would likely have validated these results, however, due to time constraints, quantitative evaluations were not performed.

4.2 Sentiment Analysis

The sentiment analysis package contained within the Stanford Core NLP package is designed to work at the sentence level, which would seem to correlate well with tweets, which are around the length of a single sentence. However, in practice, it was found that the resulting sentiment scores leaned towards neutral or negative values. The model comes pre-trained on 12,000 English language sentences on a recursive neural network. Re-training the model with Twitter specific data may be necessary for more accurate results, however, a labelled training set is not available.

4.3 Strengths and Weaknesses

Utilizing a real-time stream of Twitter data can certainly have its benefits, for one, we have instant access to trending information. It may take time for data to be posted to traditional review sites that can be harvested for topic modelling. Another benefit is the seemingly endless amount of information available. By listening to the real-time data stream, it is possible to gather very large corpora on nearly any topic we wish. This is also one of the main strengths of this project, the automatic construction of the corpus based on any desired search term. This allows for an automated pipeline that can generate the corpus, the topic models, perform the sentiment analysis, and generate the scored reviews with a single push of a button.

The main weakness of this project also has to do with the corpora being used. The data retrieved from Twitter is exceedingly noisy and incoherent. Great effort was put towards cleaning the data, however, upon analyzing the data, it was clear that the majority of the tweets do not conform to topics that would be relevant to a review. In particular, by examining text streams related to a product such as a cellular phone, it was found that the majority of tweets were advertisement for products or services with little more than a price and an http link. Another weakness lies in the processing time. Generating a corpus of 4000 tweets, and performing LDA/DMM and Sentiment Analysis on the data takes over an hour on a desktop computer.

5 Discussion and Future Work

The main problem observed with using Twitter to generate latent topics for reviews was the fact that

Topic 2, Score: 1.0 (strongly negative)	Topic 7, Score 2.1 (mildly positive)	Topic 42, Score: 1.33 (mildly neutral)
obama	love	america
coming	vote	trump
backlash	referendum	trumptrain

Figure 2: Selected results for the search item "brexit". Political tweets tend to produce more coherent topics.

Topic 32, Score: 1.325 (mildly negative)	Topic 0, Score 1.06 (strongly negative)	Topic 33, Score: 1.03 (strongly negative)
battery	screen	games
life	glass	free
dont	protector	arcade

Figure 3: Selected results for the search item "iPhone". Topic 32 produces topics that are suitable for a review, Topic 0 and 33 are most likely advertising.

many of the topics were irrelevant for the use in a review. There are a variety of ways that this could possibly be resolved. If a set of desired topic categories are known a priori, we could filter the different latent topics to only include topics whose top words contain a word from the set of desired categories. This would be a simple method of reducing the dataset to only include information that may be relevant to us. A more advanced method could be to bootstrap the topic model with seed words as is done in the model that uses *Weak Supervision with Minimal Prior Knowledge* in (B. Lu, 2011). This would allow us to guide the latent topic learning towards more coherent topic review specific topics while also allowing use to use our large-scale unlabeled corpus.

These techniques would require us to have prior knowledge of the topics. If these topics are not known beforehand, it is possible that they could be generated from a different source, such as a review website. However, this may defeat the purpose of using Twitter as a source to generate our data, unless the goal is to use the vast amount of Twitter data to augment an already pre-trained topic model. A trained LDA model has the benefit of being able to be used to infer topics on unseen data.

Sentiment analysis is an inherently difficult problem in the NLP domain. Early approaches aimed at solving sentiment analysis used a bag of words approach that simply looked at the sentiment value of individual words. To achieve better results, the current state-of-the-art Sentiment Analysis techniques, such as the RNTN model used by (R. Socher et al, 2013), look for the meaning in word phrases. They provide a pre-trained English language model that can be applied directly to documents. The problem with using this model is that the text in Twitter data tends to be full of abbreviations, slang, and improper grammar that most likely will confuse the RNTN model. It might be worthwhile to train the sentiment analysis model on Twitter text to see if it produces better results. However, this may be difficult, because the jargon contained within texts can vary wildly depending on the topic, as well the users who are producing the tweets.

Possible future work would involve trying to solve some of the issues described above. Twitter data does not appear to be very suitable for generating reviews about items which are subjected to a large amount of irrelevant text such as advertisements. However, the topics become much more coherent when the discourse becomes more professional, such as the example involving a political term shown in figure 2. It would be worthwhile to compare LDA, DMM, and Bi-Term Topic Models quantitatively to determine which model is best able to discover coherent topics from the sparse text. With regards to sentiment analysis, it would be worthwhile to test a model specifically trained against short text for scoring the sentiment of tweets.

6 References

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. *Latent Dirichlet Allocation*. Journal of Machine Learning Research, 3:993–1022.

D. M. Blei. 2012. *Probabilistic Topic Models*. Communications of the ACM, 55(4):77–84.

K. Nigam, AK. McCallum, S. Thrun, and T Mitchell. 2000. *Text Classification from Labeled and Unlabeled Documents Using EM*. Machine learning, 39:103– 134.

J. Yin and J. Wang. 2014. *A Dirichlet Multinomial Mixture Model-based Approach for Short Text Clustering*. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 233–242.

D. Quoc Nguyen, R. Billingsley, L. Du and M. Johnson. 2015. *Improving Topic Models with Latent Feature Word Representations*. Transactions of the Association for Computational Linguistics, vol. 3, pp. 299-313.

D. Q. Nguyen. 2015. jLDADMM: A Java package for the LDA and DMM topic models. http://jldadmm.sourceforge.net/.

A. McCallum. *MALLET: A Machine Learning for Language Toolkit.* http://mallet.cs.umass.edu. 2002.

J. Weng, E.-P. Lim, J. Jiang, and Q. He. *TwitterRank: Finding topic-sensitive influential twitterers*, 2010. In Proc. Int. Conf. on Web Search and Data Mining, pages 261–270

D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. 2014. *The Stanford CoreNLP Natural Language Processing Toolkit*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.

X. Yan, J. Guo, Y. Lan, and X. Cheng. 2013. *A biterm topic model for short texts*. In WWW, 1445–1456.

S. Rogers, J. Huang, E. Joo, 2013. *Latent Subtopics in Yelp Restaurant Reviews*. Yelp Dataset Challenge winner

B. Lu, M. Ott, C. Cardie, and B. Tsou, 2011. *Multi-Aspect Sentiment Analysis with Topic Models*, Proc. Sentiment Elicitation from Natural Text for Information Retrieval and Extraction, IEEE CS, pp. 81–88.

R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng and C. Potts, 2013. *Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank Conference on Empirical Methods in Natural Language Processing*. EMNLP 2013