Latent Dirichlet Allocation on Small Documents

Daniel Lewitz and Ryan Saeta

Carleton College, 2016

Abstract. An algorithm for latent Dirichlet allocation using Gibbs sampling was used on a corpus comprised of tweets found from the Twitter API for particular hashtags. The topics were then subjectively assessed and topic assignments for words in the corpus were analyzed. We found that the topics that were generated contained words that could be judged to belong together, while the quality of topic assignments for documents of the same hashtag varied.

Key words: latent Dirichlet allocation, topic model, Bayesian inference, Gibbs sampling, Twitter

Introduction

Latent Dirichlet allocation (LDA) is a generative probabilistic model that assess a list of *topics* to a list of *documents*.[1] A *corpus* is a collection of documents, *documents* are collections of words, and *topics* are probability distributions over words. The input of LDA is merely a corpus and a desired number of topics. The algorithm then decides a probability distribution over words for each topic, and then assigns a distribution over topics for each document.

Under this model, we have an assumed method of creating documents. The process for writing any particular document (excluding grammatical rules and other semantics) is as follows:

1. Create topics as a distribution over a list of words
2. Choose what proportions of each topic will be covered in this document
3. According to the distributions from 1. and 2., probabilistically draw words from the vocabulary.
4. Repeat 3. until document is of appropriate length.

With this assumption, LDA is a way to work backwards from this generation of a document. We are given documents that we assume to have been made this way, and we wish to find the topics and distributions over those topics that constitute each document.

We wish to examine the practicality of using an LDA model on documents of shorter length. Specifically, we will use tweets as our documents, in contrast to the much longer documents normally used in LDA. We will use as a corpus a collection of several hundred tweets, coming from a small number of hashtags. We will then examine if there is a strong correlation between the origin (hashtag) of a tweet, and the assignment of topics given to it by our model. While one of

the challenges of implementing an LDA model is the lack of an obvious metric to evaluate its success, we propose several techniques of measuring how well our model creates topics that can be easily identified with one of the hashtags.

Background

LDA was first introduced in 2003 by David Blei, Andrew Ng, and Michael Jordan (see [1]) as a probabilistic topic model. They outline the aforementioned assumptions of document generation and explain the interaction between documents, topics, and words. Blei et. al compare their novel LDA model to unigrams, mixture of unigrams, and pLSI. Where pLSI suffered from a sever case of overfitting, LDA was introduced to mitigate such problems.

Historically, LDA has been used to model topics from large documents and even larger corpuses. Millions of Wikipedia articles have been run through the topic to assess its quality, using more than 500 topics.[7] Additionally, using a variation of LDA called Corr-LDA Liu et. al [6] describe how they found sequences of miRNA using this probabilistic topic model. Interestingly, there have been very few documented investigations of running LDA on small documents.

LDA operates on the Dirichlet distribution which is a distribution over distributions. The Dirichlet takes 2 parameters, α and β each between 0 and 1. α regulates the document distribution over topics while β regulates the topic distribution over words. This is to say that a high α value makes documents appear more similar while a high β makes topics appear more similar. The values of α and β that should be chosen are dependent on the purpose for which one is using LDA.

Our Data

As mentioned, historically LDA has been used almost exclusively for relatively long documents. Few have applied the model to shorter documents; those who have have often used pooling, where shorter documents are grouped together by some criteria to create longer documents on which to use the method [3]. For our project, we attempted to use individual tweets, online posts of size at most 140 characters.

For our model, we used as a corpus a collection of tweets obtained from the Twitter API. We downloaded 100 tweets from each of the following hashtags:

- #HB2
- #BlackLivesMatter
- #ss2016
- #science
- #gswvokc
- #indy500

These were chosen so as to have at least some level of variance, as the differences between words commonly used in #blacklivesmatter tweets and in #science tweets should mean we get topics easily associable with one or the other. Before using these as documents for our model, we removed all hashtags, punctuation, stop words, and any other content that would not be identified with a particular topic. All tweets were passed into the LDA model together.

Training the Model

We used a Gibbs sampling algorithm to train our model. As inputs, we gave a corpus of tweets, desired α and β values, desired number of topics, and desired number of training iterations. The number of topics, k, varied, usually around 10, while the number of training iterations varied between 25 and 125. Note that due to the terseness of our documents, we used a smaller k value than is typical of many LDA models.

After passing in our corpus to the model, we gave an initial topic assignment to each word in each document. For each training iteration, we went through each word in each document, and forgot the current assignment. Assuming the rest of the topic assignments in the model are correct, the assumptions of LDA imply that topic t would generate word w in document d with probability $p(t|d)p(w|t)$. In our context, $p(t|d)$ is simply the proportion of the words in the document that are assigned to topic t, while $p(w|t)$ is simply the proportion of words assigned to topic t that match word w. We can now randomly assign a new topic to w with probability proportional to $p(t|d)p(w|t)$.

Method

Using our tweets, we attempted to find optimal values for α and β that would maximize the success of our model. We measured success in a number of ways. First, we assessed the topic distribution for each document. This gave us a sense of the quality of how individual document were analyzed under the model. Then we examined the coherence of topic distributions within hashtag groups and then between hashtag groups. This multi-step analysis provides an overall assessment of how our model worked on small documents.

Document distributions over topics

Being limited to only 140 characters, tweets cannot be strictly tied to the assumptions made originally in [2]. Instead, we assumed that tweets pertained generally to one or two topics, given the expected overlap of subjects (i.e. #BlackLivesMatter and #HB2 having similar references to discrimination and Fox News). As such, we expected that topics ought to be fairly specific and documents to be comprised of relatively few numbers of topics. Overall, we can assess the topics themselves subjectively but then we can see how many documents are of more than 3 topics.

Coherence of topic distributions in hashtag groups

Here we group tweets by the hashtags (origins) under which they were found. We expect that tweets from the same hashtag will generally consist of different topics than those from other hashtags. We can go through each tweet that was drawn from a particular hashtag and add up the topics to which each word was assigned. We can use the Pearson χ^2 test of independence to see if the overall origins have distinctive topics assigned to them. This is similar to burst score-wise pooling described in [3], but is just used for analysis rather than training or evaluation.

Results

In analyzing the results, we will first evaluate the topics themselves by looking at words that are highly associated with each topic. Then we will move on to evaluating the topic assignments given for particular documents and groups of documents.

Evaluation of Topic Generation

First we subjectively looked at the top 10 words associated with the first 10 topics generated. We noticed that there were some topics that contained very similar words like "blackisbeautiful", "blackandeducated", and "hbcugrad". This showed us that this topic was specific for #BlackLivesMatter. This we could see with a number of different topics as well, like "space" and "tech" were for #science and "kevin", "durant", "game", and "warriors" pertained to #OKCvGSW.

```
Topic 0: durant, kevin, going, controversy, antitransgender, exploiting, holybullies, foxnews, via, okcvsgsw
Topic 1: snapchat, sleep, space, tells, decisions, climatechange, making, privilege, blacklivesmatter, science
Topic 2: hbcugrad, equal, treatment, performance, gawd, thanks, good, great, blacklivesmatter, science
Topic 3: okcvsgsw, news, tech, science, virginia, threatening, looks, businesses, north, carolina
Topic 4: pastors, backing, event, black, warriors, video, nba, blacklivesmatter, game, okcvsgsw
Topic 5: away, new, gender, want, nero, lady, gaga, people, science, blacklivesmatter
```

Fig. 1. Sample topic preview after 50 training iterations on corpus

Evaluation of Topic Distribution in Documents

Here we will evaluate how documents have been assigned to topics. We will first look at documents from the same origin, and then we will look at overall trends in documents between origins.

Documents from Same Origin Because of the nature of tweets, we had to make different assumptions about the contents of documents. For example, tweets tend to be very short and thus precise in content. Therefore we expected that most tweets only referenced 1 to 3 topics each. Moreover we expected that all tweets from the same origin would reference the same select few topics over others.

To evaluate whether this expectation held, we went through each origin and for each document in that origin, we added up the number of times a word was assigned to each topic. With that, we did a Pearson χ^2 test of independence to see if these topic references could have been made randomly.

Interestingly, if we had relatively high α and β values, say around 0.5, we found that over each document, the p-value for the χ^2 tests were more in the middle, ranging from 0.05 to 0.70. However, if we had low values for α and β (around 0.05), these p-values ranged from practically 0 to practically 1 on the same run. This means that the lower our values for α and β, the more likely our origins cover a narrow *and* wide array of topics.

Documents Between Origins Similarly, we ran a Pearson χ^2 test of independence for the matrix comprised of the aforementioned sums of topic references for each origin. Therefore we had a matrix of size topics $\times$ origins.

This χ^2 test always resulted in a p-value less than 10^{-200} which shows that between origins, the topic assignments are actually quite different.

Discussion

Limitations

In this study, we had to make a lot of simplifications and use a small corpus. Due to the 100 tweet limit on the Twitter API, we had to query the API on different days using `result_type=recent`. As always, results could be improved with a larger training data set to account for a wider range of vocabulary.

Additionally, there are generally few numeric, objective evaluations of the quality of topic models. As such subjective opinion had to be relied on in order to assess our model.

Conclusion

Creating a topic model for Twitter is made difficult by the small document size, few number of words that are spelled correctly (and uniformly), words that are not stop-words, and the general character limit of tweets. This poses potential problems to our initial assignments of words to topics in tweets that contain a small number of words. If we initially assign 2 out of the 4 words of a tweet to a particular topic, then over iterations of learning, it is more likely that the other 2 words get assigned to the same topic, even if it is not logical given the rest of the corpus.

A larger document size would cause the initial topic assignments to have a mitigated influence on subsequent topic assignments because there would be a wider range of topics already in the document.

The topics that were generated had some subjective coherence among words. Words that humans could identify as being related showed up in the top 10 words for some topics, indicating that a particular origin may have more to do with one topic over another.

However, topic assignments for particular words in origins did not perform as well as hoped. Because each topic has the same words in it, it is possible for any word in any document to be assigned to any topic. There merely exists a probability distribution over words that make it more likely for say "science" to pertain to topic 1 than topic 4. The small number of words caused a skewed word assignment to propagate with higher influence than if the corpus was larger.

References

1. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. the Journal of machine Learning research, 3, 993-1022.
2. Blei, D: Probabilistic topic models. Proceedings of the 17th ACM SIGKDD International Conference Tutorials (2011)
3. Buntine, W., Mehrotra, P., Sanner, S, Xie, L. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, 889-892.
4. Teh, Yee W., David Newman, and Max Welling. A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation. Advances in neural information processing systems. (2006)
5. Chang, Jonathan, et al. "Reading tea leaves: How humans interpret topic models." Advances in neural information processing systems. (2009)
6. Liu, B., Liu, L., Tsykin, A., Goodall, G. J., Green, J. E., Zhu, M., ... & Li, J. (2010). Identifying functional miRNAmRNA regulatory modules with correspondence latent dirichlet allocation. Bioinformatics, 26(24), 3105-3111.
7. Hoffman, Matthew, Francis R. Bach, and David M. Blei. "Online learning for latent dirichlet allocation." advances in neural information processing systems. 2010.