Baseline Method of Topic Modeling – LDA [*]

Viswanath Kammula(vk865@nyu.edu), Ke Ye(ky822@nyu.edu)

1. Introduction.

As part of the baseline we will implementing the Latent Dirichlet Allocation(LDA) for the purpose of topic modeling.LDA is an unsupervised learning algorithm which will try to group all the similar documents under a single topic.LDA is a generative probabilistic model for collections of discrete data such as text corpora.The goal is to find short descriptions of the members of a collection that enable efficient processing of large collections while preserving the essential statistical relationships that are useful for basic tasks such as classification, novelty detection, summarization, and similarity and relevance judgments.

LDA assumes the following generative process for each document w in a corpus D:
1. $Choose\ N \sim Poisson(\xi)$
2. $Choose\ \theta \sim Dir(\alpha)$
3. For each of the N words we choose a topic $T_n \sim Multinomial(\theta)$.
4. Then choose a word from $p(\xi_n|T_n, \beta)$, a multinomial probability conditioned on topic T_n.

Figure 1 shows the probabilistic graphical model of LDA introduced by the paper of LDA(Blei et al., 2003), According to the paper of LDA (Blei et al., 2003), the probability of a sequence of words and topics can be expressed as

$$p(\xi, T) = \int p(\theta) \left(\prod_{n=1}^{N} p(T_n|\theta) p(\xi_n|T_n) \right) d\theta,$$

where θ is the parameter of a multinomial over topics. Thus, LDA will learn a conditional probability distribution $p(\xi|T)$ as the probability of the word ξ appears in the topic T for every latent topic T. We can then obtain the LDA distribution on documents by marginalizing out the topic variables with a Dirichlet distribution. We will get a k-dimensional vector by training k-topic model and then return the $p(\xi|T)$ values to the matrix. The output is a word-topic matrix which using the rows to represent word meanings. However, according to some previous research, we will see that LDA can't deliver robust word vectors. Therefore, we'd like to apply some new model to improve that as some later steps.

2. Data Information and Data Preprocessing .

As part of implementing the Baseline we are using the Reuters-21578 corpus. The whole dataset size in uncompressed sgml format is 28.0 MB.The Reuters-21578 collection is distributed in 22 files. Each of the first 21 files (reut2-000.sgm through reut2-020.sgm) contain 1000 documents, while the last (reut2-021.sgm) contains 578 documents.

We could directly read these documents from the reuters corpus in NLTK.The Reuters Corpus contains 10,788 news documents totaling 1.3 million words.The documents have been

[*] This work was supported by Prof. Zaid Harchaoui(zaid.harchaoui@nyu.edu)

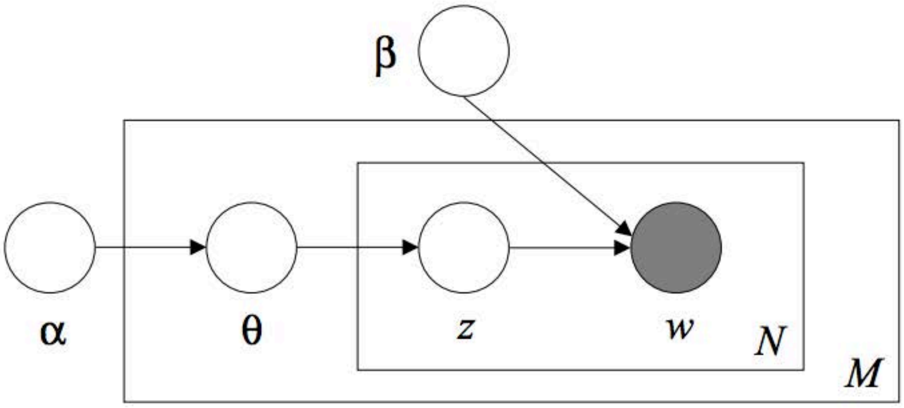

Figure 1. *The Probabilistic Graphical Model of LDA*

classified into 90 topics, and grouped into two sets, called training and test. As part of LDA we need to first performing tokenization of the Documents. Before performing tokenization we converted the words to lower cases and then removed all the stop words from the document. And then we performed stemming to remove the duplicate words.

3. Model Setup and Evaluation. For running LDA we made use of gensim package in python and developed our learning model,we used the Training data under Reuters dataset as our corpus and set the total number of topics to build this model as 90. After running the model,we suprisingly found that the model was looking better without the stemming implementation in the pre-processing stage. Therefore, we decided to do LDA on the corpus with out performing stemming.

For evaluation of the model and also to find out how many topics exists in this corpus implicitly,We ran the model LDA model by increasing the number of topics for each run from 1 to 150 topics. And we also zoomed in and ran another series of models with number of topics from 1 to 30 to see if there is any subtle change inside. We calculated the coherence value for each run, which shows the performance of the model. When we plotted the Coherence values with the number of topics, we could observe that the coherence values getting decreased as the topics increase,and it started to converge when the number of topics was around 100.Coherence metric has been shown to correlate well with human judgments of topic quality. If we perfectly reconstruct topics, all the high-probability words in a topic should co-occur frequently, otherwise, the model may be mixing unrelated concepts. Given a set of words W, coherence is

$$Coherence(W) = \sum_{w_1,w_2 \in W} log \frac{D(w_1,w_2)+\epsilon}{D(w_2}$$

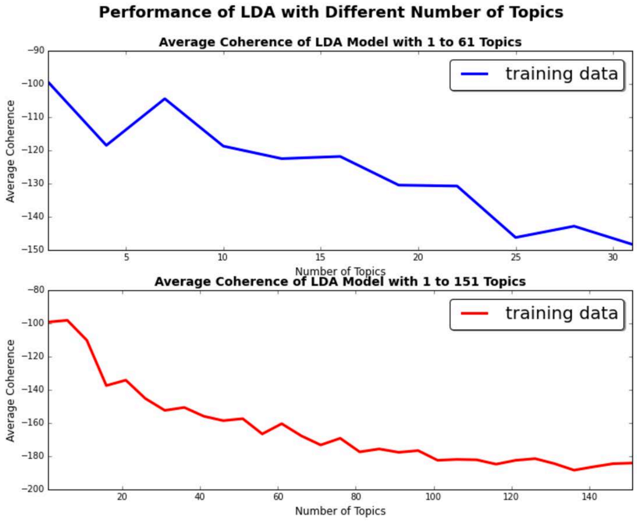

Figure 2. *The Coherence of the LDA models*

Besides, according to the paper of LDA(Blei et al., 2003), we also used per-word perplexity to evaluate the performance of our LDA model. The goal is to achieve a high probability on a held-out testing set. Thus, we computed the per-word perplexity of our held-out testing set so as to do the evaluation on our models. According to the paper, the perplexity is monotonically decreasing in the probability of the testing data. So the lower the perplexity, the better the generalization performance of the models. Here is the fomulation of perplexity, where M is the number of documents in the test set:

$$perplexity(D_{test}) = exp\left(-\frac{\sum_{d=1}^{M} log_p(w_d)}{\sum_{d=1} MN_d}\right)$$

We used both the per-word perplexity and the Coherence to evaluate our model and plotted the trends of the performance with changing number of topics in several LDA models. Figure 2 is the trend of the Coherence while Figure 3 is the trend of the perplexity. We also collected all the generated topics in a single txt file, which was easy to follow the track of training. The Coherence and perplexity keep decreasing as the number of topics increased,As mentioned above, we've known that the less the perplexity, the better the performance of the model, so we can see the performance of the model increased as the number of topics increased, which did make sense. But when the number of topics reached 90, the result began to converge and change little afterwards. And the interesting is, the original assigned number of topics for this dataset is 90, we will dive deep into that using other improved models other than LDA and make some comparisons in our following work.

REFERENCES

[1] Learning word vectors for sentiment analysis.,
`http://ai.stanford.edu/~amaas/papers/wvSent_acl2011.pdf`

[2] A practical algorithm for topic modeling with proveable guarentees.,
`http://cs.nyu.edu/~dsontag/papers/AroraEtAl_icml13.pdf`

[3] Supplement material for A practical algorithm for topic modeling.,
`http://cs.nyu.edu/~dsontag/papers/AroraEtAl_icml13_supp.pdf`

[4] Latent Dirichlet Allocation.,
`https://www.cs.princeton.edu/~blei/papers/BleiNgJordan2003.pdf`