CorrRank: Correlation Based Ranking Topic Model

Zhibo XIAO\textsuperscript{1}, Feng CHE\textsuperscript{1}, Enuo MIAO\textsuperscript{2}, Mingyu LU\textsuperscript{1,*}

\textsuperscript{1}Information Science Technology Department, Dalian Maritime University, Dalian 116026, China
\textsuperscript{2}Computer Teaching Department, Dalian Naval Academy, Dalian 116018, China

Abstract

Topic models are unsupervised mixture models that can summarize, organize digital content in a semantic-rich manner, they have received wide acceptance and been applied in various scenarios, hence become a very important research area. Unfortunately, learned topic distributions cannot distinguish from each other, which makes them difficult to be further utilized by other tasks. CorrRank is proposed to address this problem, which calculates topic correlations with the covariance matrix learned from correlated topic models, and evaluates topic quality with topic-document frequency and topic significance. Topic ranking is then carried out according to topic correlation and topic quality scores. CorrRank is applied to multi-document summarization to test its effectiveness. The experiments show that the proposed method can effectively solve the problem.

Keywords: Machine Learning; Topic Models; Ranking Topic Models; Topic Importance Measure; Multi-document Summarization

1 Introduction

Topic models have become fundamental research tools for machine learning, natural language processing, computer vision and information retrieval. As unsupervised learning methods, topic models can discover usually co-occurred terms and aggregate them in separate clusters according to their shared semantic, which usually called topics. Topic models can help to summarize, organize large volume digital contents in a semantic-rich manner. Proposed in 2003, Latent Dirichlet Allocation (LDA) \cite{1, 2, 3} based topic models have been heavily studied and some important cornerstones have been made.

The problem that hindering the further development of topic models is that learned topic distributions couldn't distinguish from each other by their own.

\textsuperscript{*}This work is supported by the National Natural Science Foundation of China under Grant No. 61370070, 61272369, 61073133, 61175053, 61105117, 61033012; Science and Technology Planning Project of Dalian City (2011A17GX073; 2010E15SF153; 2013J21DW006) and the Fundamental Research Funds for the Central Universities No. 3132013335.

\textsuperscript{*}Corresponding author.
Email address: lumingyu@dlmu.edu.cn (Mingyu LU).
In order to measure the importance of topics and obtain a ranked topic list, we propose a ranking topic model called CorrRank. CorrRank is based on the posterior result learned by Correlated Topic Models (CTM) [4]. CorrRank solely uses the learned model of CTM to get a ranked topic list. Specifically, CorrRank uses the learned covariance matrix of CTM to evaluate the relationships between the topics, and then outputs a ranked topic list according to a topic’s influence to other topics. The more connected one topic to other topics, the more important this topic is in the corpus. CorrRank not only maximally utilizes the result of CTM, but also can easily integrate other importance measure in machine learning, which exhibits great flexibility and extensibility. Unlike Dirichlet distribution used in LDA, CTM uses logistic normal distribution as prior distribution, which losses the computation efficiency brought by the conjugacy between Dirichlet distribution and the multinomial distribution. Computation methods proposed in [5] is then adopted to improve the efficiency of CTM. In this paper, we firstly established the concept of ranking topic models, which treats topic discovery and topic ranking as two equally important tasks. It is also worth mentioning that CorrRank solely use the information of the corpus and learns a ranked topic list. In the end, we test our proposed ranking topic model to multi-document summarization task and get good performance.

The following paper is organized as follows: in chapter 2 we review other works concerning ranking in topic models. In chapter 3, we briefly introduce CTM. In chapter 4, we formally present CorrRank algorithm. In chapter 5, we conduct various experiments to test our model. We conclude our work and point out the future direction of our work in the end.

2 Related Works

There aren’t a lot of work discussing ranking problems in topic models given it is still a relatively new area. Topic over time proposed [6] and dynamic topic models [7] both considered the time-stamp of documents by incorporating new random variables in the models, these two models can be seen as re-ranking topic distributions by time.

[8] proposed the problem of re-ranking topic distributions according to their importance. They defined important topics and irrelevant topics in three different manners, and then used weighted scores derived from three manners to rank topic distributions accordingly. They first raised the problem that an ordered topic list with ranking is necessary for the model. [9] proposed methods to select the appropriate words to represent topics, this can be seen as re-ranking terms in each topic. They proposed a series of features to depict the importance of terms in topic, and selecting important terms in topic via features is more suitable than just using probability information. Though from different perspectives, [8] and [9] revealed the fact that a ranked topic list is necessary. [10] proposed four topic ranking methods and did a thorough job to evaluate the importance of topics. [11] is an improvement of [12], both works investigated how link structures can influence topic discovery. The drawbacks of their works are clear that they need link information between documents to work.

3 Correlated Topic Models

CTM are extension of LDA. Topic models are probabilistic models that can learn hidden semantic topics of the corpus. Each document is treated as a combination of terms drawn from a mixture
model. These mixture components are called “topic”, each topic is a probabilistic distribution of the whole vocabulary. Each document is a combination of different topics with different proportions. In LDA and CTM, each document’s topic proportion is expressed by a latent variable. LDA uses Dirichlet distribution, which is conjugate with multinomial distribution, CTM uses logistic normal distribution, which can express the correlations between topics (for example, articles talking about economic usually talk about politics as well). Although CTM is more expressive, but due to the non-conjugacy between logistic normal distribution and multinomial distribution, the computation efficiency is lost.

Suppose there are \( K \) topics \( \beta_{1:K} \), each topic is a distribution of all the vocabulary in the corpus. Let \( \pi(\theta) \) represent logistic normal function, which maps a real value vector to the same dimension simplex, \( \pi(\theta) \propto \exp\{\theta\} \). CTM assumes a document is generated in the following method:

1. Draw topic proportion variable from a logistic normal distribution \( \theta \sim N(\mu_0, \Sigma_0) \)
2. For each term \( w_n \):
   
   (a) Draw topic assignment variable from multinomial distribution \( z_n \mid \theta \sim Mult (\pi(\theta)) \)
   
   (b) Draw term from multinomial distribution \( w_n \mid z_n, \beta \sim Mult (\beta_{z_n}) \)

\( \Sigma_0 \) is the topic correlation matrix, topic assignment variable \( z_n \) shows which topic term \( x_n \) belongs.

Given topic variable \( \beta_{1:K} \), to solve CTM model is to calculate the document level variable’s conditional distribution \( p(\theta, z_{1:N} \mid w_{1:N}, \beta_{1:K}) \). The goal to solve this probability distribution is twofold: one, we can use the result to predict new data’s topic; two, this process is the sub-process of variational EM, which can estimate the parameter of logistic normal distribution (namely, mean \( \mu_0 \) and covariance matrix \( \Sigma_0 \) using maximum likelihood estimation. Due to the lack of conjugacy, each document’s estimation process is more complex than LDA’s estimation process. Blei and Lafferty used a specified Taylor approximation to calculate this posteriori in [4]. Wang and Blei adopted mean-field variational inference method to calculate this non-conjugate problem in [5], they used to variational distribution \( q(z) \) and \( q(\theta) \) to approximation he true distribution, and proposed two computation methods, Laplace approximation and Delta approximation to solve the problem. Both two methods use gradient ascent method to optimize variational parameters, the differences lies in how to update \( q(z) \) and \( q(\theta) \). Since this aspect is not the main concern of this paper, please read [5] for detailed methods.

### 4 CorrRank

We formally present our CorrRank model. Previous works mainly focus on how to access the quality of the topics, we do believe the quality of the topics is a very important contributing factor when ranking topics, but we also believe the correlations between topics should be put more emphasis on. When ranking topics, we propose that our ranking measure consists two parts: topic correlation and topic quality. Most topic models learn topic from corpus level, document level and topic level as is shown in the plate notation of the models. We believe that ranking topics should consider these factors as well. In our model, topic correlation assesses the relationship between topics on the corpus level. Topic quality includes two parts, one is the number of documents who has this topic and the other part is how similar one topic is to other
topics. The first part access the topic importance on document level, and the second part weights the topic importance on topic level.

4.1 Topic correlation

In CTM, the logistic normal distribution is used to represent the prior knowledge. Logistic normal distribution has two parameters, one is the mean $\mu$, and the other is the covariance $\Sigma$. In $\Sigma$, each column represent a topic, since $\Sigma$ is a symmetric matrix, its row also represent topics. For each topic, we calculate its connection degree with other topic with Eq. (1):

$$t_{ck} = \sum_{i,j=1,i\neq j}^{K-1} \sigma_{ij} \quad k \in \{1, 2, \cdots, K-1\}$$

$\sigma_{ij}$ is the element of matrix $\Sigma$, $k$ is the topic index. The connection degree between topics reflects the popularity between topics, topic with high connection degree means that the semantic of this topic is closer to other topics, topic with low connection degree means that the topic is more isolated.

Although connection degree reflects the relationships between topics, but this degree is highly biased. It only takes the corpus level into consideration, but neglect the actual meaning of the terms. For example, if there exists one topic that contains a lot of popular but meaningless terms, it will be quite popular but be of less use to users. For this reason, we have to take document level and topic level information into consideration.

4.2 Topic quality

We use document frequency and topic significance to assess the quality of the topic.

4.2.1 Topic-document frequency

Topic-document frequency is similar to topic correlation, but it has its distinct character. Topic-document frequency measure the topic popularity on document level, it didn’t reflect any correlation between topics. The higher this value is, the more a topic prevails in the corpus. This reflects the proportion of one topic in a corpus. Topic-document frequency is calculated in Eq. (2):

$$df_k = \frac{d_k}{D} \quad k \in \{1, 2, \cdots, K-1\}$$

$d_k$ is the number of document that contain the $k$th topic, $D$ is the total number of the documents in corpus.

4.2.2 Topic significance

Topic significance is used to balance the topic correlation. As aforementioned, popular topics may contain popular but meaningless terms, [8] called them “junk” words. On the other hand, a small set of words that have genuine meaning are called “salient” words. In one document, there are always large number of junk words and small set of salient words. A topic is expected
to have a unique character, thus if a topic is closer to the empirical distribution of words in a document, the less uniqueness this topic possesses, hence less significance. Combined with document frequency, the significance of a topic is defined by the distance between the topic and the empirical distribution of the document which contains this topic. The empirical distribution of each document is defined as the probability of each term contains in the document. The topic significance score is defined in Eq. (3):

$$ts_k = - \sum_{i=1}^{D_k} KL(\phi_k \parallel p_i) \quad k \in \{1, 2, \cdots, K - 1\}$$  

(3)

$ts_k$ denotes the topic significance score, $K$ is the topic number, $D_k$ is the total number of document which contains topic $k$, $\phi_k$ is the $k$th topic distribution. KL divergence is used to calculate the difference between the topic distribution and the empirical distribution.

In all, document frequency and topic significance is combined to assess the topic quality.

$$tq_k = df_k \times ts_k$$  

(4)

Here, document frequency functions like a normalizer to prevent topics that prevail the corpus having too big weight.

4.3 CorrRank score

The final score is then combined to rank the topic distribution, we use a parameter to control the balance between topic correlation and topic quality. The topic ranking score is then defined in Eq. (5):

$$\text{RankScore} = \alpha \cdot tc_k + (1 - \alpha) \cdot tq_k$$  

(5)

We set $\alpha$ value to 0.6, which leans a little heavy over topic correlation part. The parameter is meant to be set manually by user to emphasis on either topic significance or document frequency.

After the score calculation, user can choose the actual top-N topics to perform other tasks. In this paper, we use multi-document automatic summarization to test the performance of the proposed model.

5 Experiments and Discussions

Since ranking topic model is a relatively new model and there is no acknowledged measure to value the performances of the algorithm. So we decide to adopt external measure to evaluate our proposed ranking topic model. Multi-document summarization has clearly a widely adopted measure- ROUGE, we can use ROUGE to evaluate multi-document summarization algorithms. Using CorrRank as base algorithm, we can compare CorrRank with other LDA based algorithms on multi-document summarization, if CorrRank based algorithm performs better, then this proves the effectiveness of our ranking scheme. We denote CorrRank based multi-document summarization as CorrSum.

For the purpose of evaluation of our multi-document summarization algorithms we used the DUC 2002 Corpus dataset. The data was made up of 59 sets of Documents each containing on
an average 10 documents and the candidate algorithms were required to make multi-document summary for each document set. The length of the summary was limited to 200 and 400 words. The candidate algorithms were supposed to be extraction based i.e. the sentences in the summary were supposed to be as they were in the documents, without any sort of modification.

We used the ROUGE evaluator [13] to evaluate the summary. We removes the stop words and calculated the ROUGE scores separately for 200 and 400 length summary as we want to even see the effect of the length of the summary on the quality of the summary. We are mainly interested in the ROUGE-1 Recall score, which uses unigram statistics, since the precision scores can be manipulated by adjusting the length of the candidate summary. We compare the results of our algorithms against the top two algorithms of the DUC2002 Multi-Document Summarization task, GISTEXTER [14] and WSRSE [15] and two LDA based algorithms proposed in [16] in terms of ROUGE-1 Recall Measures. We also take a look at the 95% Confidence Interval. In our model, we choose half of the topics to be used to select the proper sentences in DUC2002 dataset. The experimental results are shown in Table 1.

<table>
<thead>
<tr>
<th>Table 1: Comparison results of different summarization algorithms on ROUGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROUGE Setting</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td><strong>200 words</strong></td>
</tr>
<tr>
<td>Recall</td>
</tr>
<tr>
<td>95% interval</td>
</tr>
<tr>
<td><strong>400 words</strong></td>
</tr>
<tr>
<td>Recall</td>
</tr>
<tr>
<td>95% interval</td>
</tr>
</tbody>
</table>

We can clearly see that CorrSum scored higher than its counterparts. Together with other LDA based method, CorrSum outperformed GISTEXTER and WSRSE, which proves that topic indeed plays an important role in summarization. CorrSum also outperform other three LDA based methods, which shows that the topic correlation also plays an important part selecting the important or influential topics in corpus.

To further illustrate the results of our results, we perform another comparison experiments. We compared with SumBasic, Doc-LDA, KL-LDA, and use GISTEXTER, WSRSE as baseline. SumBasic is a multi-document extractive summarization algorithm based on term frequency feature, Doc-LDA and KL-LDA are all multi-document summarization algorithms based on LDA, Doc-LDA chooses topics according to topic probabilities, and then select sentences according to selected topics. KL-LDA builds LDA models on corpus and single sentences, and use the KL divergences between them to decide which sentence is selected. We still perform our experiments on DUC 2002 corpus and evaluate them using ROUGE.

The experiment results are shown in Fig. 1.

From Fig. 1(a), we can clearly see that our proposed CorrSum algorithm always performs best in all five ROUGE measures. What is astonishing to discover is that all other LDA based algorithms perform poorer than baseline algorithms. It is quite noticeable the margin between these three algorithms with the baseline two algorithms. LDA based algorithms can indeed extract the latent
topics in the corpus, but the topics are not stable to be immediately adopted by summarization algorithms. On the other hand, the term frequency is a relatively safer feature, since it only sums up the occurrence number of each word. CorrSum performs better than all other algorithms, this results proves two points. First, LDA based feature can indeed bring up the performance of the summarization. Second, ranking scheme helps to reorganize the topic distributions. Both topic quality and topic significance contribute to the overall ranking scheme.

From Fig. 1(b), we can see the similar results on 200 words summary. CorrSum continues to outperform other comparison algorithms. Though the overall margin is not as large as 200 words summary. But still, CorrSum performs better on all ROUGE measures. Different from 200 words summary that LDA based summarization algorithms (other than CorrSum) are all beaten by baseline algorithms. In 400 words summary, in some tests, LDA based summarization algorithms start to outperform baseline algorithms. From this point, we can see that LDA based algorithms are born to analyze large corpus, it is quite obvious CorrSum and other LDA based algorithms can discover latent semantics of the corpus, while term frequency can only discover shallow meaning of the words and the corpus.

6 Conclusion

In this paper, we proposed a ranking topic model called CorrRank. CorrRank ranks the learned topic distributions of CTM. CorrRank considers two factors when ranking topics: topic correlation and topic quality. Topic correlation is calculated using the learned posteriori of CTM, topic quality is assessed by considering topic-document frequency and topic significance. We apply our proposed model on multi-document summarization tasks on DUC2002 corpus to empirically test CorrRank. The experiment results show that CorrRank outperforms the state-of-the-art algorithms and further shows its effectiveness. We also have to point out that, CorrRank has its limitation, it heavily rely on CTM, especially the topic correlation part. In the future, we intend to expand our work by devising more general topic correlation measures, so other topic model result can be used to rank topic distributions. Also we intend to apply to CorrRank to larger

![Fig. 1: Rouge scores of different summary algorithms](image-url)
corpus on summarization tasks, due to limit resources, it’s quite hard to collect human generated summarization.

References