Analyzing and Navigating ETDs Using Topic Models

Abstract. Electronic theses and dissertations (ETDs) contain valuable knowledge that can be useful in a wide range of research areas. Accordingly, we are building electronic infrastructure leveraging advanced work on digital libraries, for discovering and accessing the knowledge buried in ETDs. In this paper we focus on our work to incorporate topic modeling into digital libraries for ETDs. We present ETD-Topics, a framework that extracts topics from a large text corpus in an unsupervised way. The representations learnt from topic models can be useful for downstream tasks such as searching and/or browsing documents by topic, document recommendation, topic recommendation, and describing temporal topic trends (e.g., from the perspective of disciplines or universities).

1 Introduction

Scholarly documents like ETDs contain important research findings, which are of value to diverse digital library users. Examples of such users include students and researchers who want to review work related to their research area, as well as librarians and university administration who want an overview of recent research in their institutions.

With the vast amount of research being conducted across a variety of domains, millions of ETDs are now publicly available online. However, digital library services for ETDs have not evolved past simple search and browse at the metadata level, thus rendering the vast amount of information from these documents underutilized.

In recent years, there have been many advances in the fields of text mining and natural language processing (NLP). One line of work of value in the analysis of ETDs is topic modeling, which aims to extract thematic collections of words that could represent topics, from a large corpus of text documents. Several topic modeling algorithms have been proposed over the years. Probabilistic Latent Semantic Indexing (PLSI) [1] and Latent Dirichlet Allocation (LDA) [2] were among the earliest works in this domain. Recent works in the domain of topic modeling include neural topic models such as NeuralLDA, ProdLDA, [3], and CTM [4]. The representations learned from topic models can be used for downstream tasks that rely on document representations, such as finding similar documents (document recommendation), finding similar topics, analyzing the variation of topics over time, etc.

In this work, we propose ETD-Topics, a topic modeling based framework for analyzing and discovering information contained in ETDs using several state-of-the-art topic models. Our framework allows users to extract topics present in an ETD collection using one of the several topic models provided. Users can then select a topic of interest, and do further analysis of the topic using multiple end-user services supported in our framework. Supported services include searching documents associated with a particular topic, calculating the distribution of the documents w.r.t. topics, document recommendation, topic recommendation, and topic trend analysis based on time range and/or university. Moreover, since topic models
are fully unsupervised in nature, our framework does not require any handcrafted labels such as categories, thereby making it easily deployable and scalable for new document collections.

Figure 1: An overview of ETD-Topics.

2 ETD-Topics: System Description

Fig. 1 shows the architecture of our framework. The system pipeline components are described below:

2.1 Data Source

Since our framework aims to assist in analysis of massive amounts of ETD data, we require a large collection of text ETDs. For each ETD, we use its title and abstract as the corresponding text. This text is then tokenized and goes through a series of preprocessing steps, such as stop word and punctuation removal, removing terms with low document frequency (infrequent words), and lemmatization. We also drop documents whose token count is less than a certain threshold number, as these are likely to be documents with limited or missing text. Finally, we obtain a list of tokens for each document that can be sent to the topic modeling module.

2.2 Topic Modeling

This module forms the main backbone of our system. It takes the preprocessed data as input and uses topic modeling algorithms to extract the topics from the document corpus. The topic modeling algorithms currently supported are:

- **LDA [2]**: LDA is one of the earliest topic models, that uses Bayesian priors as the initial document-topic and topic-word assignments, and then updates these distributions based on the probability with which a document or a word is associated with a certain topic.
- **NeuralLDA [3]**: This is the neural network based version of LDA, that utilizes a variational inference [5] method for learning document-topic representations.
- **ProdLDA [3]**: This is an improved version of NeuralLDA, that is designed to give more coherent and interpretable topics.
- **CTM [4]**: In contrast to other topic models that use bag-of-words representations for text and hence ignore the order of words, this model combines representations from language models like BERT [6] in the topic modeling process, thus incorporating word context.

Since topic models require several iterations over the dataset for training, we train all the models offline, using different numbers of topics for each model. We set the number of topics (denoted as $K$) to $\{10, 25, 50, 100\}$ while training the models, thus resulting in 16 pre-trained models (4 models, each with a different value of $K$, for each of the 4 algorithms listed above).
All topic models typically give two types of outputs. The first is a $K \times V$ topic-word distribution matrix, where $V$ is the vocabulary size. Each row of this matrix represents the importance of each of the words in the vocabulary for the respective topic. The second is an $M \times K$ document-topic distribution matrix, where $M$ is the number of documents in the corpus, where each row represents the proportion of each of the topics in the respective document.

2.3 User Services

The front end user interface (UI) encapsulates multiple downstream tasks and services for users of a digital library. Below are descriptions of services illustrated in Fig. 2.

2.3.1 Topic Browser

Our framework allows users to select one of the topic modeling techniques, and the number of topics, and to get the resultant topics. This service includes three major components:

- **Documents per Topic Distribution**: This module helps users find the most popular topics in the document collection. Given a threshold value (on a scale of $[0, 1]$, default = 0.3) and a topic, this component calculates the number of documents in the entire database for which the given topic constituted more than the threshold. The overall results are displayed as a histogram, where each bar shows the number of documents for that respective topic.

- **Topic List**: For every topic, this module shows the top 10 words that are representative of that topic; the set thus serves as a type of label.

- **Similar Topics**: Some users work in interdisciplinary fields. In such instances, it is often desirable to show a list of related topics to the user. This is done based on similarities between different rows of the topic-word matrix.

2.3.2 Document Browser

The document browser allows users to get specific documents based on their interests. It mainly consists of two modules:
Table 1: Quantitative Comparison of Different Models, with underlined values indicating best performing models (*NeuLDA denotes NeuralLDA).

<table>
<thead>
<tr>
<th></th>
<th>LDA</th>
<th>NeuLDA</th>
<th>ProdLDA</th>
<th>CTM</th>
<th>LDA</th>
<th>NeuLDA</th>
<th>ProdLDA</th>
<th>CTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.75</td>
<td>1</td>
<td>0.96</td>
<td>1</td>
<td>0.044</td>
<td>-0.057</td>
<td>0.037</td>
<td>0.104</td>
</tr>
<tr>
<td>25</td>
<td>0.752</td>
<td>1</td>
<td>0.94</td>
<td>0.94</td>
<td>0.080</td>
<td>-0.038</td>
<td>0.077</td>
<td>0.114</td>
</tr>
<tr>
<td>50</td>
<td>0.792</td>
<td>0.988</td>
<td>0.92</td>
<td>0.948</td>
<td>0.076</td>
<td>-0.037</td>
<td>0.116</td>
<td>0.136</td>
</tr>
<tr>
<td>100</td>
<td>0.831</td>
<td>0.937</td>
<td>0.858</td>
<td>0.879</td>
<td>0.076</td>
<td>-0.039</td>
<td>0.117</td>
<td>0.130</td>
</tr>
</tbody>
</table>

- **Topic Specific Documents**: This module allows users to get relevant documents for one of the many topics shown in the Topic Browser. It selects the documents based on the presence of the selected topic in the document using the corresponding values of the document-topic vectors. It then displays the title and abstract of the selected document. It also allows users to get more details of a specific document by clicking on it.

- **Related Documents**: This module assists users in finding documents that are similar to a selected document. This is especially useful in the case of scholarly documents, since users are typically interested in finding multiple related works.

### 2.3.3 Topic Trend Analysis

- **Temporal Analysis**: Many users of a digital library, such as university administrators and faculty members, are interested in analyzing how different research areas trend over time. This module allows users to filter documents associated with a topic in a given time range.

- **University-Specific Analysis**: In some instances, users are interested in analyzing research trends in their institution, or in peer institutions. This module shows users research trends by university. Additionally, users can combine this feature with temporal analysis to get institution-specific research trends over time.

### 3 System Setup and Analysis

#### 3.1 Dataset and System Details

Our dataset has ~320K ETDs from over 42 universities. They come from 1845 – 2020, with most published after 1945. Our topic models are from open source implementations included in OCTIS [7]. The UI was developed using Flask with a Python backend.

#### 3.2 Comparison of Different Topic Models

Table 1 shows the performance of different topic models on our collection, for two commonly used metrics. **Diversity** is a measure of how diverse the top words of a topic are w.r.t. top words in other topics. A score of 0 indicates redundancy, while 1 indicates very diverse topics. **Coherence** measures how likely words from the same topic tend to occur in the same document. Models with high coherence tend to give more interpretable topics.

We observe that NeuralLDA produces more diverse topics than other models, indicated by its high diversity score, with CTM being the second best performing model in terms of

[https://flask.palletsprojects.com/]
Table 2: Corresponding Words for a Topic from Different Models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>network communication user channel mobile security node wireless protocol</td>
</tr>
<tr>
<td>NeuralLDA</td>
<td>thesis network perform introduce efficient end describe integrate linear</td>
</tr>
<tr>
<td>ProdLDA</td>
<td>network problem challenge approach base provide system design framework</td>
</tr>
<tr>
<td>CTM</td>
<td>network protocol ad mobile node attack internet secure request</td>
</tr>
</tbody>
</table>

diversity. However, the coherence scores for CTM are much better than other models, indicating more interpretable topics. A good topic model should ideally have high coherence and diversity scores, since high diversity and low coherence could also mean that the topics are composed of unique, yet unrelated words which do not indicate any themes. In Table 2 we also show the corresponding words for one topic obtained from all the models. The topic produced by NeuralLDA is less coherent, indicated by words like thesis and introduce, in line with its low coherence scores. In contrast, the topics produced by LDA and ProdLDA are cleaner, though they do have some open-ended words like user and provide. CTM produces the most coherent topic, which is also reflected by its high coherence scores. It appears that CTM is the best overall performing model on our ETD corpus.

4 Conclusion and Future Work

We presented a framework for using topic models for extracting topics from a large ETD corpus in an unsupervised way. The framework also supports several downstream tasks and a rich user interface to assist users of a digital library. In the future, we plan to improve the framework by adding support for more topic models, more services such as search, mapping topics to existing ontologies like from ProQuest, and improvements to the user interface.

References