TextViz

Text Visualization for Topic Modeling

Divya Karthikeyan
School of Information
University of California, Berkeley
divyak@berkeley.edu

Vaidyanath Venkitasubramanian
School of Information
University of California, Berkeley
tvv@berkeley.edu

Abstract—Managing large text collections is a huge challenge in almost every industry. In this paper we discuss a visualization-based approach that helps in understanding the underlying relationship between documents. TextViz is a text analysis and visualization platform that aids in the analysis of text corpora using topic-modeling methods. The results are displayed as a map that provides means to explore the relationship between the documents, topics and top words in the corpus.

Author Keywords—Text Visualization; Topic Models; Text Mining; Text Analysis;

INTRODUCTION

Topic models are statistical models for discovering the abstract topics that occur in a collection of documents. A document could concern multiple topics in different proportions and this could be determined by the presence of related words in them. The results of topic modeling require intensive expert verification and model refinement to analyze the quality of the analysis. Also every time the model parameter is changed it is essential to understand how the underlying structure and links between documents and topics change.

Manual analysis of topic modeling results are time consuming but provide the best understanding. A method that could strike a trade off between the two would be the most ideal to perform text analysis on large documents. We wanted to develop a tool that would provide a quick summary of text analysis in addition to allowing the user to explore through in detail. Also since text analysis requires the variation of model input parameters to determine an ideal condition the interface would accept parameter changes and produce results in one click.

RELATED WORK

There have been previous attempts to tackle the problem at hand and develop a system that helps in visualizing text analysis. Discussed below are two techniques that attempt at doing this.

1. Provide a visual approach for assessing the quality of topic models
2. Provide a navigator to browse through the documents and underlying structure
3. Provides a Metadata and Topic Model Analysis Toolkit

Termite is a visualization technique for assessing textual topic models developed at Stanford University. It uses a tabular layout to promote comparison of terms within and across topics [1]. The tool is designed to assess individual topics and all topics as a whole. The visualization is a term topic matrix that shows term distributions for all latent topics. Hall et al. [3] studied research trends in computational linguistics using latent Dirichlet allocation (LDA) over 14,000 publications. After expert analysis of the topic model results, only 36 out of the 100 topics were retained. Current evaluations are dependent on expert analysis and hence Termite came up with a visual means to help expedite the process. They filter the display to show up only the most probable or salient features. The users have an option to filter between 10 and 250 words to display [1], but it is recommended to use less than 250 words since it would deter the effectiveness of the visualization. The future work includes gathering model parameters from the user and reassessing the results with the term matrix for iterative topic modeling.

Princeton University has a visualization developed that summarizes the corpus, reveals relationships between and across the content of documents [2]. They make use of LDA to expose hidden random variables that encode its thematic structure. David Blei et al. were the first to provide a graphical model for topic discovery. They applied the method to 100,000 Wikipedia articles as a running example. The navigator has two main types pages: one for topics and the other for the documents. Every topic is tied to the documents that it contains. There is also a list of related topics in every corresponding link. Their user study indicated that users discovered relationships that would have remained obscure by using Wikipedia traditionally [2].
MetaToMATo (Metadata and Topic Model Analysis Toolkit) is an interface developed in John Hopkins University. Corpus visualization tools provide high-level view of the data. There are two types to such tools: [5] Tools that rely on metadata and tools that rely on information derived from the documents. This tool has a corpus, documents and topics view. Users are able to browse through the components and understand hidden structure.

**METHODS**

Our visualization interface encompasses the topic modeling algorithm and a coherent visualization technique to effectively display the results. We included in our system some features that were part of the future work in some of out references. Our system contains the following components.

*Figure 1: Overall system components.*

**Model Selection**

We provide two, topic model approach to use for clustering the document collection: LDA and Non-negative Matrix Factorization (NMF).

LDA is a generative model where each topic can be viewed as a mixture of various topics. LDA considers the documents as a bag of words and makes the assumption that the order of occurrence of words does not matter. While this assumption is unrealistic, this is reasonable, since the goal is only to uncover the hidden structure of documents [4].

The algorithm can provide a set of probabilities for the presence of a word in a topic and the proportions of a document in a topic.

NMF can be used in text mining applications. A document term matrix is constructed with term weights. Features are derived from the documents.

The interface provides a Model selection section where all model parameters can be user specified. If left empty the default parameters will be used. The file upload option will accept single/multiple documents. The topic modeling parameters that the user could specify are as follows:

1. Number of Topics: The number of topics that the model must generate
2. Number of top words to be retrieved from a topic: The tops words from a topic to be displayed in the map
3. Number of features: Features to be considered in modeling
4. Algorithm: Choice of the topic modeling algorithm LDA/NMF
5. Stem words: Perform additional processing in the text by stemming the words

**Map visualization**

The topic model results are displayed in concept map visualization. The Map holds the generated topics in the center and links to documents and words on either side. The graph is interactive and allows the user to click through components and browse the underlying structure.

Hover over a topic name, document or words highlights the corresponding links. Clicking on either of them show up a new graph that shows all relationships to the selected topic/document or word. There is an ability to go back to the overall topic structure.

*Figure 2: Interactive concept map highlighting the documents and words in Topic 2.*
The user has the option of varying the model parameters at any point during the interaction. The user can run over numerous iterations to find an optimal model and result. As shown in Figure 5, the model selection box takes model inputs form the user. If the parameters were left black, optimum default values would be used.

A text summary pane lists quick statistics about the document corpus uploaded. The details include:
1. Number of documents
2. Total number of words and unique words
3. Most frequent words throughout the corpus

**IMPLEMENTATION DETAILS**

The web-based interface allows users to upload documents and perform the text analysis. The application is built with a JavaScript, HTML and CSS front end and Python backend. The text processing and topic modeling is in python using nltk and scikit.

**Design Process**

We wanted to visualize text from the documents to understand how clusters are generated. The initial design was to implement using word clouds. But word clouds would become challenging to compare when there are more than few topics.

Since all prior work in this field provides matrix/tabular visuals of word counts and frequencies concerning a topic, we wanted to retain the word or text format of the data. To retain
all hidden relationships in one map we decided to use a circular force layout. The topics would appear in the center listed one below the other. The top words used to cluster appear in the left semi circle. The topics on hover link to words that belong to the corresponding topic.

The text processing is performed in Python. Data from the documents uploaded in input to the python script that converts the documents into bag of words. The script then performs topic modeling based on one of the two algorithms and returns an object. The object is used in JavaScript to generate a force layout.

RESULTS

The interface produces interactive visualization that allows users to browse through the documents and topic model results. The topics are presented vertically along the center of a circle. The left half of the circle is populated by ‘words’ and the right half by ‘documents’. The interface was tested with random text from Wikipedia and a sample of 10 resumes a sample of 10 resumes. The user would be able to understand the structure of the documents iteratively in a short span of time.

DISCUSSION

We had some comments from the poster session as follows:

1. Can be linked to Google scholar
2. Would be a very useful tool for lawyers as most of their work revolves around identifying topics associated with documents

FUTURE WORK

Our current system implements two topic-modeling approaches. It would be beneficial to implement other approaches like latent semantic analysis (LSA). The addition of probabilities corresponding to the topics generated in the concept map could help distinguish documents that fall under multiple topics with the probability score. Presently the system links to all the topics the document might belong to.

The interface accepts documents in text format and an extension would be to provide some data processing options if the documents are not in textual format.

REFERENCES