Visualizing Topic Models Generated Using Latent Dirichlet Allocation

Ashwinkumar Ganesan, Kiante Branley
Advisor: Dr. Jian Chen & Dr. Shimei Pan

Abstract—Topic Modeling is a set of statistical methods used to find "topics" in a given document corpus, where a *topic* is defined as a word distribution. Topic Modeling opens the doors to a set of interesting questions such as the various topics in documents & which documents are associated which kinds of topics. One of the problems with statistical methods is that the word distributions generated may not interpretable by the user. Our project, *LDAExplore*, discusses the visualization designed, to look at topic models and how the user can interact with these statistical methods and provide feedback to improve the underlying model. Latent Dirichlet Allocation (LDA) is one of the basic methods to find such hidden topics that has been used in the project. To validate our design, we have used the *abstract*'s of 322 *Information Visualization* papers, where every abstract is considered a *document* and generated topics which are then explored by users.

Index Terms—DataViz, InfoViz, LDA, Latent, Dirichlet Allocation, Parallel Coordinates, Treemap, Topic Modeling.

1 Introduction

A Large body of human knowledge has traditionally been stored in the form of documents. These documents maybe in the form of books, magazines, journals. With the advent of the digital age, more and more content is stored digitally and with the cloud computing becoming increasingly common, we stored a lot of information online. We have websites that present this knowledge online in the form of HTML pages and are looking at this information being presented on mobile platforms such as IOS and Android in the form of apps. This growing knowledge base brings with it a unique set of challenges such as searching through a large set of documents for a specific piece of information, grouping similar documents together, looking at how these documents and their underlying content change over time. These changes could be the common theme or the language that is utilized in these documents.

Topic Modelling tries to automate the process of extracting topics from documents and annotate them with semantic information [2]. It is a set of statistical algorithms that extract correlated words from documents. These extracted word sets are called *Topics*. They are later annotated with semantic information, so that they are easily understood by people. For e.g. consider a a word set extracted such as *visualization, sets, clusters, infoviz, interfaces* from a document then we have a general notion that this word set represents the topic *Information Visualization*. In this example, the "topic" generated using a topic modelling algorithm is the word set and *Information Visualization* is the semantic "topic name" annotated by the user. Latent Dirichlet Allocation (LDA) [1] is one of the common methods to perform topic modelling on a given corpus of documents.

LDA generates viz. two types of distributions i.e. the topic distribution for each document in the set and the word distribution for each topic. The model works by computing the joint probability distribution of the words in each document to group words which appear together and form a single topic. These distributions can be changed by tweaking the underlying parameters. Our project, *LDAExplore*, tries to give visual cues about how these distributions look, and how the topics and documents are interrelated at the corpus level and for groups or individual documents. Also, it gives the users the ability query documents for specific words based on topics and see how correlated a topic is to a document. Some of the main concerns while creating the design are that visual should be able to work with a large set of documents while providing the ability to see individual and group relations. The number of topics, though, is considered to be a smaller set as compared to the number of documents.

2 Related Work

There are quite a few ways of looking that the problem of how to represent distributions to show their interrelations. Some of the methods include looking at distributions graphs or sets [5]. In these visualizations, the distributions are converted to sets transforming them to a hierarchy of nodes. A standard technique to represent sets is to create an *Euler* diagram [5] or a variant of it. Region-based overlays can be used such as *Bubble Sets* and *Texture Splatting* or Line-based overlays like *LineSets* and *Kelp Diagrams* [5]. Some of the other methods include *Node-Link* diagrams and *Force-Directed* graphs. Standard Node-link diagrams include *Jigsaw*, Anchored maps and *PivotPaths*. These design patterns invariably limit the number of nodes that can be represented

• *Ashwinkumar Ganesan is with the Department of Computer Science and Electrical, UMBC.*
Kiante Branley is with the Department of Computer Science and Electrical, UMBC.
E-mail: {gashwin1, bran4}@umbc.edu

in a visual without applying some aggregation method which clusters nodes or edges together.

Topic-based, interactive visual analysis tool (TIARA) [3] shows topic distributions across documents across time. This helps users understand how content changes across time. *RoseRiver* is another analytics system to used to visualize how topics evolve [6]. The system uses a tree cut approach with a combination of a word cloud. The word cloud is a standard method to show a word set with word sizes varying according to their frequency (or probability). Higher frequency (or probability) words are given a larger size in the word cloud.

One of primary problems with topic modelling methods is that the "topic" generated by them, may not have clearly understood by the user. Sometimes, the words in the topic do not form a coherent group together, giving the user no idea of the larger (latent) topic. One of the solutions to this problem, is to introduce a human-in-the-loop paradigm where users can interact with the algorithm, providing feedback such that the underlying model can be modified to generate "better" or semantically correct topics. One example of such a system is *UTOPIAN* [7], which uses a force-directed graph to represent topics.

3 Design Considerations

3.1 Task Analysis

As described in the previous section, results after performing topic modelling, can be unintuitive. There are two basic requirements our project tries to achieve. The first is to provide the novice users ability the ability to understand the a method like LDA. The other is to provide advanced users with the options to explore the document set. This project provides the base design to achieve these goals while to keeping the design open, so that featured identified in the future can be added. *LDAExplore* is a visual tool that is designed to explore topics, their word distributions and the relations that exist between topics and documents. Following are the set of tasks which form the basis for our design.

3.1.1 Visualize Topics

LDA generates a set of topic, each having its own individual word list. Each word has a probability of being associated with the topic. When the user interacts with a specific topic the relative importance of words will be displayed. This is one way for the user toidentify an important topic.

3.1.2 Overview of Document - Topic Relations

Once the user has a generic idea of how the topic look, user can then focus on the documents and topics correlations. The main purpose is to be able to see which topics have a higher density of documents, while which topics have a lower density. The density of documents gives the user an idea which topics are more or less important.

3.1.3 Remove Topics from the Visual

When the user knows which topics are important, the rest of the visual can de-cluttered by removing excess topics from the visual.

3.1.4 Filtering Documents

When the size of the corpus is large, the user will likely try to understand only a subset of the documents. Hence the user will filter the documents based on various criteria:-

1) The *IDs* of the documents. This will be helpful, if the user is aware of what documents are there in the corpus.
2) By top words in each document that associated with the topics having the highest probability of being related to the document.
3) The user may not know the documents, but may be able to identify some based on the *name* or *title* of the document. Thus, the user can filter out individual the documents.

3.1.5 Perform set operations

While working with groups of documents, the user intuitively performs set operations such as *include* and *exclude*. *Include* gives the user, the ability to add a filtered set of documents to future filters. *Exclude* does the exact opposite, which is to remove the document set from any filter operations in the future. Once the document corpus has been explored, the user can export the filtered data for any post processing that is performed separately.

3.1.6 Show & Cluster Similar Topics

Once the word distributions for each topic are known, the user will look for topics that are similar. Topic similarity can measured using methods such KL-Divergence. Similar topics can be grouped together, so that the number of topics on the visual display can be reduced and hierarchical topic collections can be formed.

3.1.7 Perform Cluster Operations

Similar topics, will lead to a new group of documents which have these topics. The main task for the visual is to enable the user to define groups of documents and topics based on their own knowledge.

3.1.8 Annotating Topics

User should have the option to annotate of topics and documents or the respective clusters they lie in.

3.2 Prototype Design

Based on the set of tasks defined, we describe how *LDAExplore* works.

3.2.1 Visualizing Topics

We use a *treemap* to visualize the topics. The diagram below, shows how the topics are displayed. Each rectangle is a topic and is given an ID like $T_1 ... T_n$ where there n defined topics. In figure 2, the total number of topics are 20. In the treemap, the size of the rectangle defines the probability of the topic. In the current design, a likelihood is defined for individual topics with respect to each document, rather than the document collection as a whole. Hence the topics are represented by squares showing that all topics have an equal likelihood.

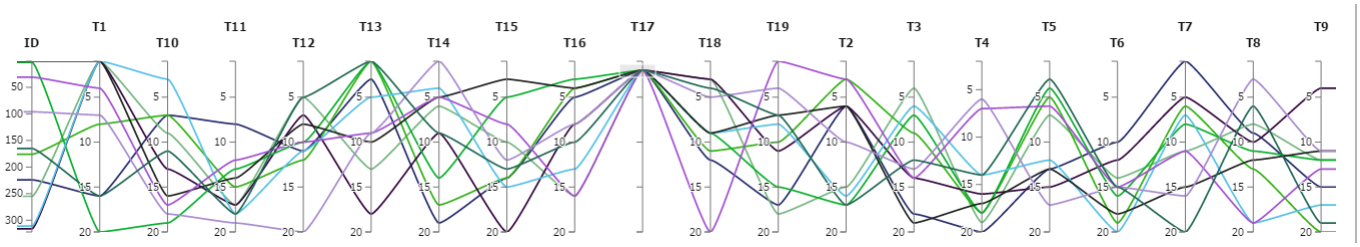

Fig. 1: Displaying the Topic Distribution

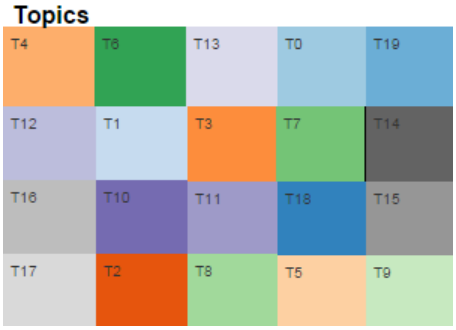

Fig. 2: Displaying the Topic - Document Relations

When the user clicks on the a specific topic, the word associated with the topic are revealed. Figure 3, shows the an example word distribution. The word with the highest probability has the largest area. In this example, the word *data* has the largest probability for topic *T4*. In the current design, the number of words displayed per topic is maintained constant at 10. The user can traverse back to the *Topics*, clicking on the *Topics* tab at the top of the treemap. Whenever the user drills down, into the word distribution, the parent is always known.

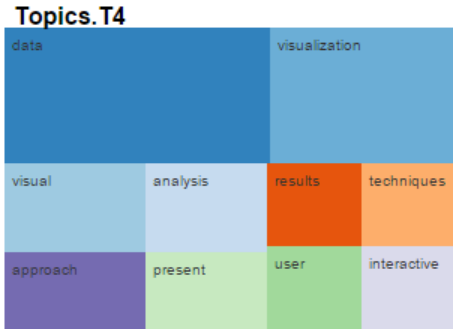

Fig. 3: Displaying the Topic Distribution

3.2.2 Relating Topics & Documents

We use parallel coordinates to show the correlation between topics and documents. The parallel coordinates have two types of axes. The first is the one that represents the document. The second type of axis is for topics. For e.g. consider Figure 2, which shows a total of 21 axes. The first axis *ID* shows the all the documents in the corpus and ranges from document id D_0 to $D_3$22. The other 20 axes each represents a topic. Once LDA generates the probabilities for each topic in a document, we rank the topics. The topic axes shows the rank of the topic with respect to the document. So in figure 2, document D_0 has Topic T_1 at rank 1, while others have it at rank $4, 7, 16\&20$ respectively. Each document has a different color, so that the user can distinguish between the graphs of various documents. The ranks are ordered in ascending order with 1 showing the highest degree of correlation and n (where n is the number of topics), showing 0 or the least correlation. Topic ranks give the user which topic is the most correlated and which is the least.

3.2.3 Filtering

LDAExplore provides a range of filtering options. Figure 4 provides a detailed overview of the filtering features. There are three main types of filtering:

1) Filtering by Range - Each axis can be used to filter the documents by selecting a specific range on the axis. The parallel coordinates displays the curves, only for the selected documents. Each axis can filtered simultaneously, thus creating a *filter-chain*.
2) Filter by Searching - A simple search bar is provided to the user, to search for documents using words. The words are generated by calculating the probability of a word from a each topic, being associated to the document. 4 explains this mechanism in more detail.
3) Filter by Selecting Individual Documents - The documents column, lists each document in the corpus by its title. The user can click on a specific document to filter it out. Whenever a document is filtered out, its color is changed to downplay the document.

Once the user has filtered the documents, the user can highlight a specific document within the filtered document set by using the mouse to hover over the document - word set (column 3 in figure 4). The total number of document in the filtered set is given at the top on the navigation bar. Also, the user can export the filtered data to CSV format.

3.2.4 Set Operations

There are 2 main set operations that are supported in *LDAExplore* i.e. inclusion and exclusion. The *include* operation is performed by the *Keep* and exclusion using *exclude* as seen in figure 4.

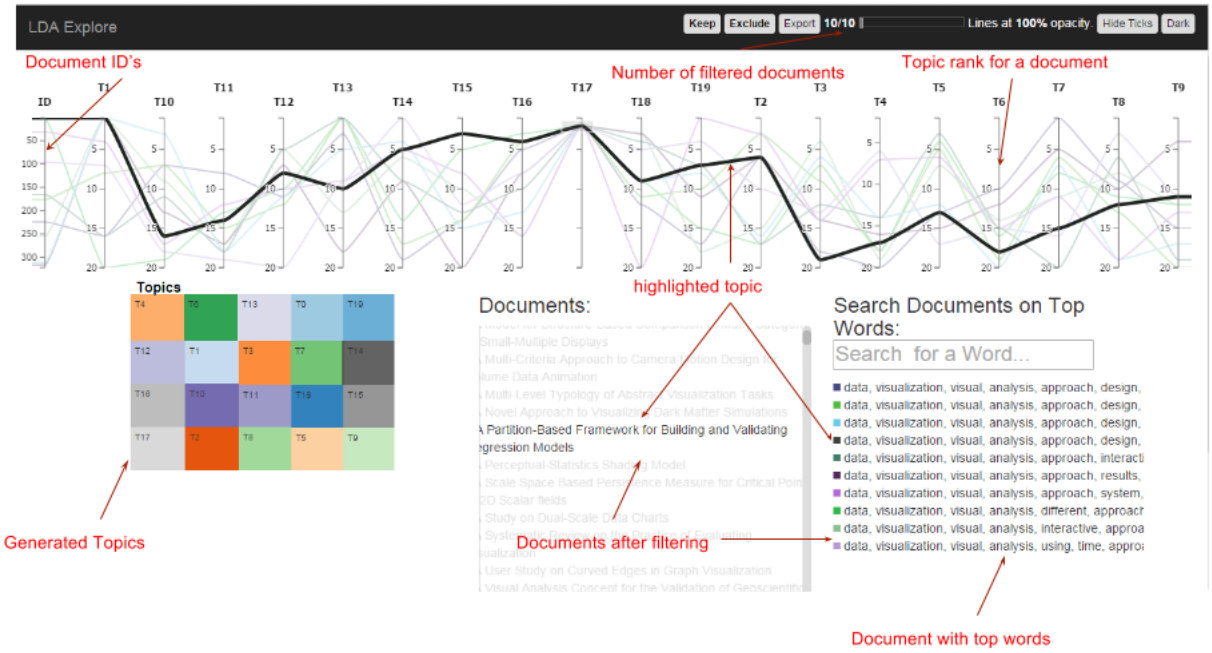

Fig. 4: Displaying the Topic Distribution

4 System Overview

5 Case Study

6 Conclusion

Acknowledgments

The authors would like to thank Dr. Jian Chen & Dr. Shimei Pan. Without their active guidance and support in design the visuals. We would like to thank all the participants of the study, who spent a portion of their valuable time evaluating our project.

References

[1] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.

[2] Blei, David M. "Probabilistic topic models." Communications of the ACM 55.4 (2012): 77-84.

[3] Pan, Shimei, et al. "Optimizing temporal topic segmentation for intelligent text visualization." Proceedings of the 2013 international conference on Intelligent user interfaces. ACM, 2013.

[4] Yang, Yi, et al. "Active Learning with Constrained Topic Model."

[5] Alsallakh, Bilal, et al. "Visualizing Sets and Set-typed Data: State-of-the-Art and Future Challenges (Supplementary Material)."

[6] Cui, W., Liu, S., Wu, Z., & Wei, H. (2014). How hierarchical topics evolve in large text corpora.

[7] Choo, J., Lee, C., Reddy, C. K., & Park, H. (2013). UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization. Visualization and Computer Graphics, IEEE Transactions on, 19(12), 1992-2001. Chicago