

Visual Analysis of Topical Evolution in Unstructured Text: Design and Evaluation of TopicFlow

Alison Smith^{*†}, Sana Malik^{*}, Ben Shneiderman^{*}

^{*}University of Maryland, College Park, MD

[†]DECISIVE ANALYTICS Corporation, Arlington, VA

{amsmit, maliks, ben}@cs.umd.edu

Abstract. Topic models are regularly used to provide directed exploration and a high-level overview of a corpus of unstructured text. In many cases, it is important to analyze the evolution of topics over a time range. In this work, we present an application of statistical topic modeling and alignment (binned topic models) to group related documents into automatically generated topics and align the topics across a time range. Additionally, we present TopicFlow, an interactive tool to visualize the evolution of these topics. The tool was developed using an iterative design process based on feedback from expert reviewers. We demonstrate the utility of the tool with a detailed analysis of a corpus of data collected over the period of an academic conference, and demonstrate the effectiveness of this visualization for reasoning about large data by a usability study with 18 participants.

Keywords: Statistical Topic Modeling · Data Visualization · Natural Language Processing · Topic Evolution

1 Introduction

Statistical topic modeling is a well known technique for discovering the “topics” that occur in a collection of documents. Topic modeling has been used to provide a high-level overview of a corpus as well as directed exploration[14]. Typically, topic modeling is applied as a “batch” process and leads to topics that cover the entire corpus but don’t take into account the fact that topics may change over time. Although this is sufficient in some cases, the increasing availability of streaming data has given rise to a number of Use Cases that require an understanding of the evolution of topics. For example, a public relations team would like to monitor how discussion about the company they are representing changes over time; campaign managers may want to understand how their candidate’s public perception has changed over time with respect to how they have been represented in the media; a news team would want to identify emerging topics representative of “breaking news”; and researchers may want to track how published work has shifted over time within their respective fields.

Techniques exist for modeling the evolution of topics, but this work does not typically lend itself to identifying emerging topics or modeling the flow of topics as they converge, diverge, and end. Additionally, because these algorithms explicitly incorporate time into the underlying topic modeling algorithm, they are not generalizable to

alternative topic modeling implementations (entity topic modeling, hierarchical topic modeling, etc.) and incorporating additional features.

We present binned topic models, an application of statistical topic modeling that is well suited for studying topic evolution on streaming text data. This technique models complex trends over time, and is particularly suited for discovering emerging topics and following topic divergence and completion. Binned topic models are topic models generated independently for adjacent time slices of text data, so topics generated at one slice do not directly correspond to the topics of another. To align the topics we use the cosine similarity metric. Displaying the results of topic modeling, and in particular topic evolution, is a difficult problem. In this paper, we provide a solution to this problem with our visualization tool, TopicFlow, which visualizes the emergence, convergence, and divergence of complex topics in a data stream¹.

In this paper, we:

1. Describe an analysis technique for text data over adjacent time slices, *binned topic models and alignment*, which is an application of Latent Dirichlet Allocation (LDA) [2] to time-stamped documents at independent time intervals and alignment of the resulting topics,
2. Introduce TopicFlow, an *interactive visualization tool* that aligns similar topics between time slices and displays topics as they emerge, converge, and diverge over a given time period, thereby identifying and providing insights that would otherwise go unnoticed, and
3. Present a multi-part evaluation of TopicFlow that shows its usefulness for following the flow of topics in text.

2 Related Work

TopicFlow covers two main areas: automatic topic detection by generating topics from a high volume of unstructured text and trend visualization over time.

2.1 Topic Detection

Existing tools follow trends in user-generated web content, however, these either only deal with short phrases [10] or are primarily concerned with locating spikes in activity rather than analyzing the trend throughout the full time range [8].

Latent Dirichlet Allocation (LDA) is an unsupervised algorithm for performing statistical topic modeling that uses a “bag of words” approach, treating each document as a vector of words where order is ignored. Each document is represented as a probability distribution over some topics where each topic is a probability distribution of words. The traditional LDA model does not take into account how topics may change over time.

A few variants of statistical topic modeling exist for incorporating time into the topic model. Topics over Time [22] is a topic model that captures time jointly with

¹This work is an extension of our prior work [13], in which we originally introduced TopicFlow as a Twitter analysis tool.

word co-occurrence patterns, such that each topic is associated with a continuous distribution of timestamps. In this case, the meaning of a topic remains constant over the time range. Topics over Time performs batch processing, meaning that as new data comes in, the method must re-model the entire data set. [1] presents continuous time dynamic topic models, a dynamic topic model that uses Brownian motion to model latent topics through a sequential collection of documents. Topics are not held constant, and the words that make up the topic may change over time. This technique facilitates evolution analysis of a particular topic over the time range; however, the model fails to represent the emergence of a unique topic within the time range or the convergence or divergence of existing topics.

2.2 Trend Visualization

The primary motivation for TopicFlow is to analyze the evolution of discovered topics for any unstructured text source. In this initial work, we develop a visualization that is representative of topic evolution over a time range.

Two trend visualizations that are closely related to TopicFlow are [6] and [4]. ThemeRiver uses a stream graph to visualize thematic variations over time from a large collection of documents. ThemeRiver defines themes as single words, and the strength of a theme is determined by the number of documents containing the word. This definition does not support complex themes that must be defined by more than a single word. TextFlow, shows the evolution of topics over time as well as merging and splitting. TextFlow uses a semi-supervised clustering technique for topic creation and represents topic convergence and divergence using a flowing “river” metaphor. The river metaphor is visually appealing for a small number of topics, however it quickly becomes cluttered; even at 15. Also TextFlow inhibits access to the underlying data, which limits analysis. The TopicFlow approach involves a general solution which is not limited to analysis of trends over time; unlike these existing trend visualizations, the TopicFlow approach can be adapted to any grouping, for example geographic location, publication, or author.

The TopicFlow visualization was directly inspired by a Sankey diagram [16], which is a type of network graph typically used to represent directional flows of data through a system where the width of the paths are proportional to the flow quantity. TopicFlow uses a generalized version of a Sankey diagram implemented in the Data-Driven Documents library [3], which is a library specifically designed for creating visualizations for large datasets.

3 Binned Topic Models

In this section, we present the application of LDA to a corpus of unstructured text documents *binned* into time slices followed by the alignment of the topics produced for each bin. To begin, the corpus is divided into bins; the number of bins to be used is specified as an input parameter. Each bin represents a time slice of equal length with no restriction on the number of documents it may contain. In future versions a non-parametric modeling approach or an approach based on expected document rate may

be more appropriate to determine the bin size and number of appropriate bins. For the underlying Statistical Topic Modeling algorithm, TopicFlow uses an open-source LDA implementation[18]. Standard LDA requires as input the documents and the number of topics² to discover, although algorithms exist to automatically determine an appropriate number of topics based on the data [21]. To produce a binned topic model, LDA is applied independently for the documents of each bin followed by an alignment step, which aligns similar topics between bins.³ The algorithm employs a stop words list to remove common words that do not contribute significant meaning to topic modeling. The TopicFlow stop words list contains standard English, and, additionally, is modified for a given dataset by including query terms used in data collection and stop words specific to the domain⁴. In later work, we intend to support a dynamic stop words list which will incorporate stop words specified by users, as well as, stop words discovered from the domain [23].

The granularity of this modeling approach can be adjusted by varying both the number of topics modeled as well as the size of the bins. Bin size selection depends on the event timescale a user is interested in (e.g. for breaking news, bins on the order of minutes or hours would be preferred; for consumer trends, bins on the order of days or weeks may be more applicable). The number of topics depends both on bin size—larger bins will typically contain more topics—and the level of topical detail users or their analysis requires.

The result of topic modeling is a distribution of words for each topic in the topic model, $P(word|topic)$, and a distribution of topics for each input document, $P(topic|doc)$. For our Use Cases, we provide users with the ability to select a topic of interest and see all corresponding documents. To enable this, each document was assigned to the topic resulting in the highest $P(topic|doc)$. Additionally, in presenting this information to users, we rank the documents by probability, such that documents with higher $P(topic|doc)$ for the topic are ranked above those with a lower probability. We chose this method because it is a simple and effective way to distribute documents across topics.

3.1 Topic Alignment

The binned topic modeling approach generates an independent topic model for the documents in each bin. Because the topics generated at individual bins do not directly correspond to each other, an alignment step is necessary to associate similar topics in adjacent bins and visualize the topic evolution. Binned topic models result from using cosine similarity to compare each pair of topics from adjacent time slices. Cosine similarity measures the cosine of the angle between two vectors⁵. This metric is regularly used for the comparison of documents or the cohesion of clusters in text analytics and

²For TopicFlow, the number of topics is adjustable with a default of 15 to balance granularity and comprehensibility of the resulting topics

³For this implementation the LDA algorithm runs for 100 iterations with $\alpha = 0.5$ and $\beta = 0.5$.

⁴For example, Twitter-specific stop words include {rt, retweet, etc.} and Spanish stop words include {el, la, tu, etc.}

⁵ $cos(A, B) = \frac{A \cdot B}{\|A\| \|B\|}$

data mining, respectively [20], and has also been previously used for the comparison of topics produced by LDA [12][17]. While many metrics exist specifically for measuring the similarity or divergence between two probability distributions [11][15] [9], small differences in low-probability outcomes may have a relatively large impact on the overall metric. For binned topic models, this is undesirable because the topics produced by LDA are primarily characterized by high probability words and variations in low-probability words may be noisy. By using cosine similarity, the influence of any two corresponding probabilities on the similarity calculation is proportional to the product of those probabilities relative to the products of other pairs, limiting the impact of lower probabilities compared to higher probabilities.

Cosine similarity returns a value between -1 and 1, where 1 would represent the exact same word distribution for each topic. Although cosine similarity ranges between -1 and 1, when dealing with probability distributions it must be between 0 and 1, because there are no negative probabilities. Instead of assigning the one most similar topic at time $n + 1$ for each topic at time n , we present links for any topic pairs with similarity above a certain threshold to enable the visualization of topic convergence and divergence. The threshold varies with the data set and should be set to balance the discovery of useful topic links with the total number of links displayed.⁶

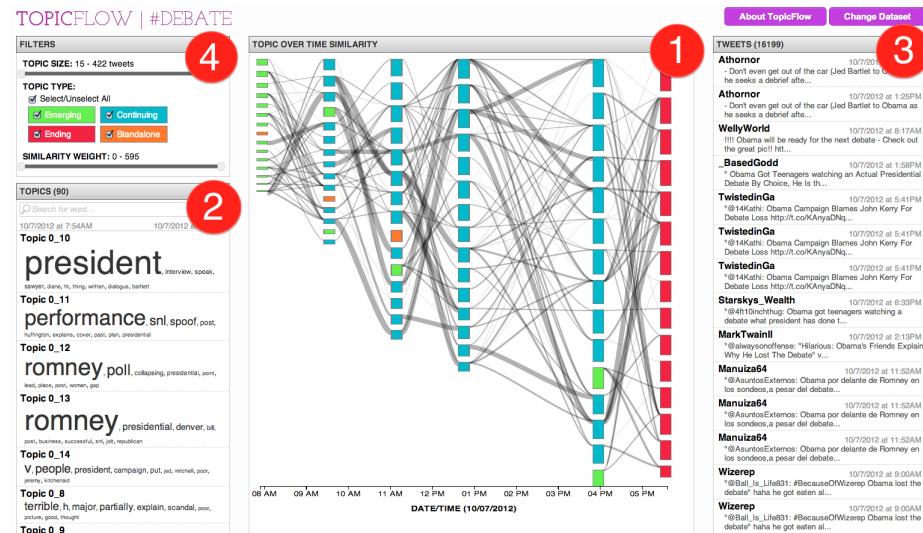


Fig. 1. TopicFlow consists of four coordinated windows: (1) the TopicFlow diagram, (2) a list of topics with their word summaries, (3) a list of the documents (in this case, tweets) in the dataset, and (4) a filter pane.

⁶For prototyping and evaluation purposes, the threshold was set between 0.15 and 0.25 depending on the dataset.

4 TopicFlow

The purpose of TopicFlow is to allow interactive exploration and analysis of the evolution of topics generated from a corpus of text documents. Figure 2 provides an overview of the TopicFlow system. The system begins by ingesting a corpus for a given time range and splitting the documents into “bins” based on time slices within the range. LDA is then applied independently at each of the bins, producing the corresponding topics. These topics are aligned at neighboring time slices using the cosine similarity metric. The resulting binned topic model is presented to users through an interactive visualization that provides methods for filtering, searching, and performing detailed exploration of the underlying data.

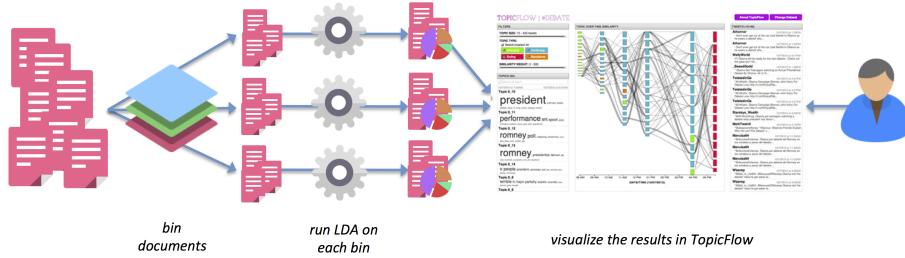


Fig. 2. System overview of the TopicFlow System. The system ingests a corpus for a given time range and splits the documents into time slices, applies LDA at each time slice, and aligns the resulting topics from neighboring time slices. The results are then presented to users through an interactive visualization that includes tools for filtering, searching, and performing detailed exploration of the underlying data through coordinated views.

4.1 Design Methodology

TopicFlow⁷ visualizes the evolution of topics of discussion within text corpora, and was designed to support six primary Use Cases:

1. Easily identify the most prominent topics within each time slice. A topic is considered more prominent if there are more documents associated with it.
2. Easily identify which topics are emerging, ending, continuing, or standalone. Here we introduce four new terms:
 - *emerging*: A topic that was not discussed in the previous time slice. (i.e., there is not topic similar to it in the previous time slice).
 - *ending*: A topic whose discussion does not continue into the next time slice (i.e., there is no topic similar to it in the next time slice).
 - *continuing*: A topic that has been discussed before and after its time slice.

⁷A prototype of the TopicFlow tool is available for demo here: <http://www.cs.umd.edu/~malkis/topicflow/TopicFlow.html>

- *standalone*: A topic which is not related to any topics in either the previous or next time slice.
3. Explore details about a selected topic. These details include its most probable words, assigned documents, and the flow of a topic over time. The *flow* of a topic is defined as the path between a topic and its related topics across all time slices.
 4. Identify topics by the words that describe them. A user may be interested in how one or more words appear throughout the dataset. By identifying the topics that are related to these words, a user can understand how the context of a word changes throughout the dataset, as well as discover other words related to it.
 5. Compare the top words in two topics that are related. By comparing two topics, a user can identify which words contributed to the topics having a high or low similarity score.
 6. Filter topics by size, type or similarity weight. Users may want to view only highly similar or highly popular topics, and filtering will allow them to hide the topics in which they are not interested.

The resulting TopicFlow visualization is composed of four coordinated windows (Figure 1): the flow diagram, topic list, document list, and filter panel, that support detailed analysis of the topic trends and underlying data.

4.2 Flow Diagram

The TopicFlow visualization employs a Sankey diagram [16] implemented in the Data-Driven Documents library [3] for displaying the topic evolution where nodes in the graph represent the topics and the paths between nodes at neighboring time slices represent topic similarity. The paths are weighted by the similarity of the topics as calculated by the cosine similarity metric. This graph is ideal for visualizing convergence and divergence of topics, represented by more than one path entering or exiting a topic, respectively. The color of the nodes is used to distinguish topics by their evolution state: emerging, ending, continuing, or standalone. The nodes are sized by the number of documents attributed to the topic, and they are ordered horizontally from the top by decreasing size. In future work, we intend to provide a number of ordering criteria for users to choose from, such as evolution state or user-specified importance. By ordering based on node size, the most prevalent topics are at the top of the graph, and users can quickly see how the frequency of a topic evolves over time. The design of this diagram was motivated by Use Cases 1 and 2 and is successful in providing insights about the prevalence and life-cycle of the topic.

4.3 Coordinated Panels

In addition to the main data visualization, the TopicFlow tool includes three coordinated panels, a topic panel, document panel, and filter panel that support deeper exploration and filtering of the underlying data.

The topic panel contains a visual representation of the topics discovered for the data. The topics are grouped by their corresponding time slice. The topic in the topic

panel can be expanded to gain additional information. The expanded view includes a histogram that represents the distribution of words in the topic.

The document panel contains the underlying documents of the corpus. The individual documents can be expanded to provide users with the full text of the document and a histogram of the five topics with the highest probability for the document, $P(\text{topic}|\text{document})$. In the case of Twitter, users can also follow a link to the author's Twitter page or to view the original Tweet.

Finally, the filter panel, which was designed in support of Use Case 6, includes the following data for filtering the visualization: by node size (number of documents attributed to the topic), node type (emerging, ending, continuing, or alone), and path weights (based on cosine similarity values).

4.4 Interaction

Interaction with a visualization is essential to analysis, because a user must drive the visualization to highlight areas of interest and generate insight with respect to the underlying data. TopicFlow supports the following set of interactive elements.

Topic Search The Topic Panel includes a search functionality for locating topics containing particular keywords. A user can search for a word of interest, and the topic list is filtered to show only topics containing the search term. Additionally, the remaining topics are highlighted in the flow diagram. This functionality supports Use Case 4, by allowing a user to focus analysis on topics containing specific words.

Graph Highlighting When a topic is selected in either the topic panel or the flow diagram, the corresponding node and topic are highlighted. Also, in the flow diagram, the nodes that have a path to the selected topic are highlighted while unconnected topics are greyed out, in order to display the selected node's subgraph (Figure 3). The selected topic is also expanded in the topic panel. Finally, to support Use Case 3, the document panel is filtered to show the ranked list of documents for the topic.

Topic Comparison When an edge is selected within the flow diagram, a topic comparison box is displayed that uses mirrored histograms to highlight the words the two topics have in common and provide information about the probability of the words for the topics. This supports Use Case 5 (Figure 4).

Tooltips TopicFlow offers tooltips when hovering over nodes and edges. On nodes, the tooltip displays a "word fade" of the related words, sized and ordered by the words' probabilities. When hovering over edges, the names of the two nodes the edge connects are shown.

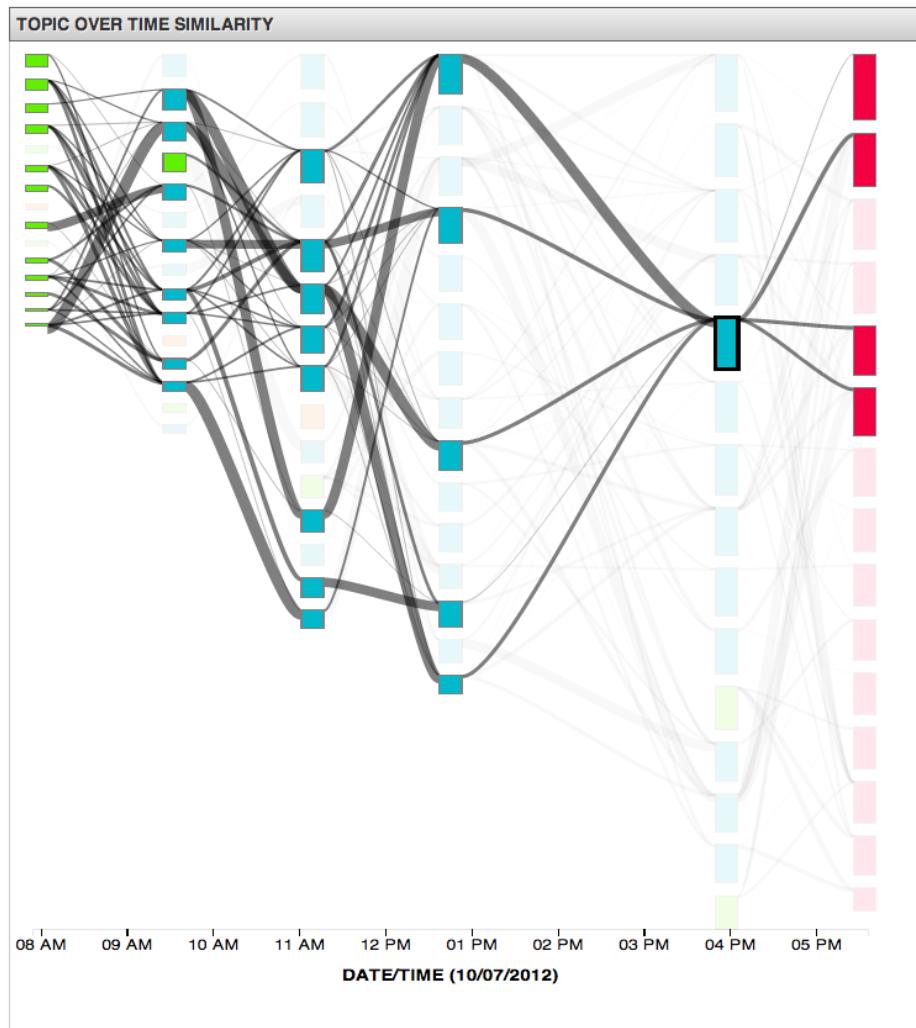


Fig. 3. When a topic is selected, the diagram is highlighted to show the flow of that topic over time.

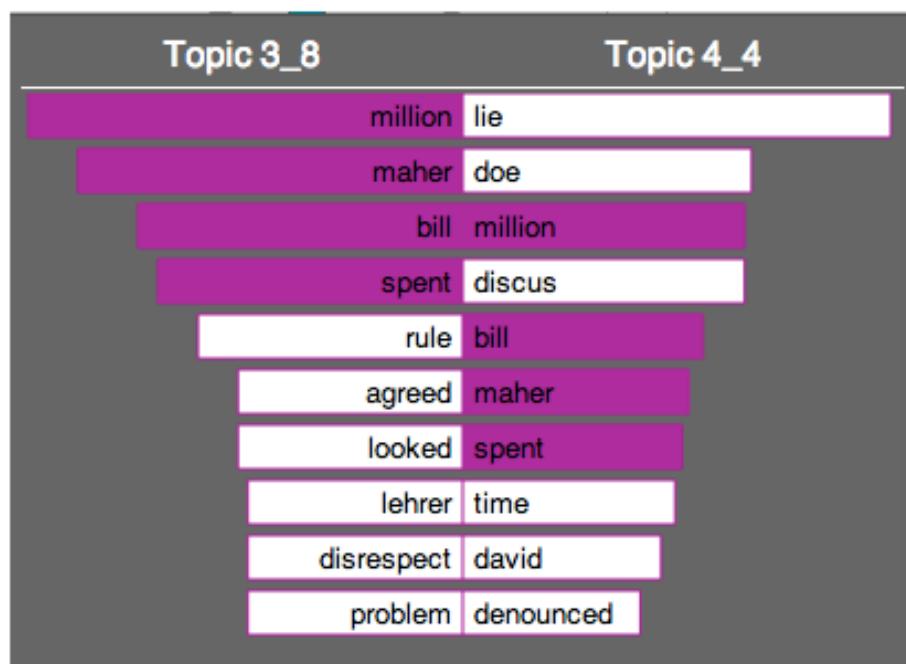


Fig. 4. The topic comparison box shows bar charts representing the two topics connected by the edge. The top words that the topics have in common are highlighted in purple.

Node Filtering The filter pane includes two double-ended range sliders, where users can limit the range of values for the topic sizes (by number of documents) and edge weights (topic similarities). Users can also limit topics by their type – emerging, ending, standalone, or continuing – with checkbox selectors. As nodes and edges are filtered from the graph, the visualization hides topics that become unconnected from the rest of the graph.

5 Evaluation

TopicFlow was evaluated in three stages: expert reviews with five participants during the design process, case studies performed during development, and one usability study with 18 participants at the end of development. These evaluations primarily used data collected from Twitter⁸ to demonstrate the functionality on streaming unstructured text data.

5.1 Expert Reviews

To drive the design process, expert reviews were conducted over two sessions with five different participants, all of whom had previously performed analysis of text data and had some graduate education or higher. The participants were recruited by email and by word-of-mouth.

The first sessions were conducted with three participants. After a brief introduction and explanation of the tool, we allowed the participants to have a freeform exploration of a data set (approximately 1500 tweets resulting from a search for the word “earthquake” over two days). They were instructed to describe everything that they were doing and why, as well as express any other comment that they might have (think-aloud method). Their comments and our observations (mistakes they made, unreasonable learning curves, bugs, confusing interface actions, missing items, etc.) were documented in handwritten and typed notes, taken by the researchers present during the session. The feedback from these sessions were incorporated into the design of the final tool.

6 Case Studies

During development, TopicFlow was used to analyze a variety of streaming text datasets, including real-time current events (Presidential debates and Hurricane Sandy), communities (University of Maryland), common interests (Modern Family and Big Data) and other historical data sets (CHI Conference). Each of the datasets contained between 1,500 and 16,000 tweets. We used 7 time bins and 15 topics for each dataset. These values were chosen to balance granularity and accuracy of the topics for the number of tweets and timespan of the datasets. The tweets were collected over varying time spans. A more detailed study was performed for the data gathered about the 2012 CHI (Computer Human Interaction) Conference[19].

⁸Twitter’s open API and the fact that tweets are rich with metadata, specifically time stamps, makes it an appropriate data source for prototyping and testing

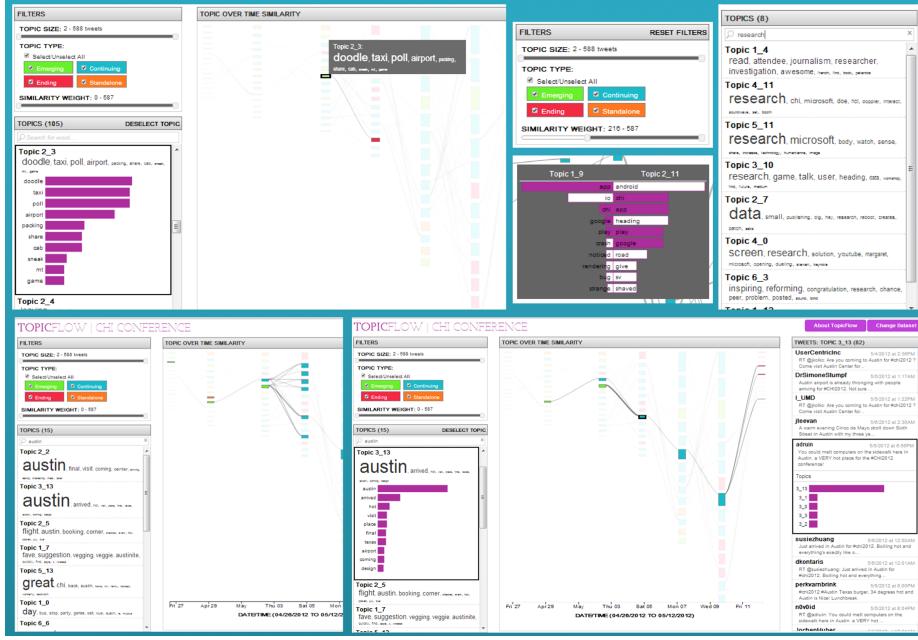


Fig. 5. Results of various interactions with the TopicFlow tool on data gathered about the 2012 CHI Conference. [Top Left] Highlighting an {airline, doodle, poll} topic. [Top Middle] The filter panel with the minimum similarity weight increased, and a comparison between two topics at consecutive time slices which are about a CHI-related app available on the Google Play store. [Top Right] The list of topics resulting from a search for the keyword “research”. [Bottom Left] The topics resulting after a search for “Austin”. [Bottom Right] Highlighting the {Austin, arriving, heat} topic and a specific tweet associated with the topic.

2012 CHI Conference TopicFlow was used to analyze 12,000 tweets gathered during the week of the 2012 CHI Conference, which contained the keyword “CHI”. The visualization shows the evolution of topics between 4/26/2012 and 5/12/2012. Figure 5 shows the TopicFlow tool on the CHI Conference data set. A few observations stand out as particularly interesting:

1. There is an emerging topic, {airline, doodle, poll}, prior to the conference, which is associated with a number of tweets discussing a doodle poll that the conference organizers sent out to determine how attendees were planning to commute to the conference.
2. After modifying the similarity weight to show only topics that are highly related, a link is prevalent between two topics at consecutive time slices where both are discussing a CHI-related app that at the time was available on the Google Play store.
3. A topic search for “research” shows that topics containing this term are consistent throughout the data set.

4. A topic search for “Austin”, the location of the conference, shows a strong trend of related topics from right before up until the end of the conference. In particular, a topic of people who are discussing “coming to Austin” exists prior to the conference and then shifts to a similar topic, {Austin, arrived, heat} where people are discussing that they have “arrived in Austin” and, in particular, are pointing out the heat.

For the case studies we performed, we found that the binned topic models were most accurate and concise for real-time events which occurred over short time spans. For example, TopicFlow on the corpus of documents related to Hurricane Sandy showed the discussion evolve from emergency preparation to the event itself, then to the aftermath and related recovery efforts. Alternatively, more general data sets, such as University of Maryland, did not have clearly defined or correlated topics due to the high number of diverse, unrelated events that occur on the campus.

6.1 Usability Study

To assess the usability of TopicFlow for exploring text corpora, we conducted a preliminary usability study with 18 participants (8 female), aged 21–49 ($M = 26.5$, $SD = 6.41$). Five of the participants had six to ten years of experience using a computer, and the rest had 11 or more years of experience. Participants were recruited through on campus mailing lists and were compensated \$10 for their time.

The study was performed on a dataset of 16,199 tweets that were collected on October 7, 2012 (four days after the first 2012 presidential debate) between 8:00 AM and 7:30pm and which contain both the hashtag “#debate” and the word “Obama.” As there is no widely used tool for visualizing and interacting with topics over time, there is no baseline to which to compare TopicFlow. Instead, after a brief introduction to the tool and five training tasks, participants were asked to complete seven tasks that are based on the developed Use Cases.

1. Which topic appears most frequently in the second timeslice and how many tweets are associated with it?
2. What are the top two words for the least frequent topic in the third timeslice?
3. What topic emerges in timeslice 3?
4. Which two topics have the highest similarity index?
5. What is the longest chain of topics connected with weights of 400 or more?
6. Which topic is the word “Romney” *most* relevant to?
7. What is the text of the tweet responsible for the standalone topic in timeslice 3?

The participants then rated each task on a 20-point Likert scale (where a higher score is better) on four metrics based on the NASA Task Load Index [5]: performance, effort, frustration, and effectiveness of the tool. A score over 18.0 was considered to be excellent performance, 15.0–17.9 was considered above average, 12.0–14.9 was average, and a score below 12.0 was considered poor. Each session lasted approximately 30 minutes. At the end of the session, participants completed a feedback questionnaire and provided comments about the efficacy of TopicFlow’s features.

Task Number	Time (sec)		Performance		Effort		Frustration		Effectiveness	
	M	SD	M	SD	M	SD	M	SD	M	SD
1	29.8	29.4	17.4	3.6	16.3	3.6	17.3	3.4	17.5	2.7
2	9.2	4.4	19.2	1.6	18.9	1.7	18.2	4.1	19.4	1.3
3	10.4	11.2	18.2	3.9	18.5	2.0	17.2	4.8	19.1	2.2
4	39.7	25.6	17.8	2.9	15.1	3.8	17.3	4.4	15.9	2.0
5	47.3	29.2	18.0	2.8	13.7	5.3	17.2	3.7	15.9	4.4
6	16.9	18.8	18.0	3.7	18.0	3.3	18.7	1.6	18.6	2.5
7	81.2	48.4	14.6	4.8	11.7	4.8	13.8	4.2	13.3	4.5

Table 1. Mean (M) and Standard Deviation (SD) for Time, Performance, Effort, Frustration, and Effectiveness for each task. Time is measured in seconds, and performance, effort, frustration, and effectiveness were measured on a 20-point Likert scale (higher numbers indicate a more favorable rating)

6.2 Results

The means and standard deviations of 18 participants on time, performance, effort, frustration, and effectiveness (Table 1) vary widely across tasks. Time is measured in seconds, and performance, effort, frustration, and effectiveness were measured on a 20-point Likert scale (higher numbers indicate a more favorable rating).

The results show that the TopicFlow interface allows users to quickly and easily perform tasks which support the initially defined Use Cases. Participants performed the fastest for tasks involving identifying details about topics (Tasks 2, 3, and 6), on average taking 10 to 20 seconds. Tasks that involved details about the number of tweets in a topic (Task 1) or evaluating the edges in the graph (Tasks 4 and 5) took longer, about 30 to 50 seconds on average. Task 7, which required analyzing the document list for a topic, took participants the longest amount of time to accomplish (81.2 seconds on average). Many participants commented that they would have found it more helpful if the tool allowed the document list to be re-sorted or if retweets were aggregated and displayed only once.

Task Load Index The Task Load Index ratings reflected the results of the time taken for each task. Tasks 2, 3, and 6 had consistently excellent (above 18.0) ratings for all four metrics, while Tasks 1 and 4 had consistently above average ratings (between 15.1–17.8) on all metrics. Task 5 had excellent ratings for performance (18.0), but required much more effort to achieve this level of performance (13.7). Task 7 was consistently the most difficult, with average ratings for each metric (13.3–14.6).

The feedback questionnaire allowed participants of the usability study to provide qualitative comments about TopicFlow’s features. The participants’ favorite features included the responsiveness of the main visualization to interactions (e.g., hovering and clicking for topic information and subgraph highlighting). One participant stated that these features are “very straightforward” and that the tool “answers questions about dominating themes within trends very well.” Participants also appreciated the tooltips when hovering over nodes and edges. Since standard topic modeling does not provide

descriptive names for the resulting topics, the users found it helpful that the visualization displays the top words of a topic, so they could quickly understand the topic’s meaning. Similarly, for the edges of the flow diagram, users appreciated the side-by-side bar charts representing the similarity between topics. One user commented that the coloring of the topics facilitated analysis; for example, using the emerging topic color to “find which topics ‘trigger’ other topics.”

Most of the participants noted that the document list pane was their least favorite feature and requested methods for sorting the documents by various metrics (time, number of retweets, etc). Because of the lack of quantifiable feedback, participants were often not confident in their answers for Task 7 (which was to identify the most retweeted tweet in the document list). In addition, participants felt the filter pane needed improvements — updating the graph by the sliders sometimes had a delayed response or choosing a specific value for a filter was imprecise due to lag in the slider feedback.

7 Future Work and Conclusion

Future work for TopicFlow includes modifying the interface to address feedback received from usability study. Although we use time slices for the purpose of this application, binned topic models is a general technique that can be applied to any data source under any binning criteria, such as geographical location or author. To account for the occasionally confusing results of topic modeling, binned topic models could implement a technique such as Interactive Topic Modeling [7], which allows users to refine the topics generated by the model. While TopicFlow garnered particularly favorable reviews for its interface, there were suggestions regarding the document list pane that can be incorporated into future work. Most notably, users requested a way to sort documents by various metadata such as time or author.

The scalability of the TopicFlow system is dependent on the algorithm for generating binned topic models and the interface. Open-source LDA implementations exist that are scalable to very large datasets [24]. The binning technique partitions the data to allow multiple LDA runs to be done in parallel, which further increases scalability of the algorithm. The TopicFlow visualization is scalable in terms of the number of documents displayed, as paging is used to handle overflow of data to the interface. In the current version, the screen space provides a limit to the number of topics and bins that can be visualized effectively; however, overview visualization methods could be used to support visualizing thousands of topics or bins.

TopicFlow provides a novel visualization of the alignment of topics over time. Our approach applies the statistical NLP method of topic modeling to text data, which allows for richer analysis of “topics” within the data. When LDA is run over an entire corpus, it produces a high-level overview of the corpus’ content. Alternatively, TopicFlow splits the corpus into a set of time slices and applies LDA on each time slice. This method provides for a more granular set of topics and allows for meaningful exploration of topic emergence, convergence, and divergence. Because topics between time slices are not directly correlated, providing our metric for the similarity between two topics allows users to follow the evolution of the word distributions over time. Our evaluation demonstrated that TopicFlow allows users to easily view the frequency of

documents relating to a particular topic over time. TopicFlow further facilitates data exploration by providing details-on-demand about automatically extracted topics through hovering and filtering interactions. The use of colors and tooltips provides users with a quick summary of individual topics.

8 Acknowledgements

We would like to thank Timothy Hawes, Cody Dunne, Marc Smith, Jimmy Lin, Jordan Boyd-Graber, Catherine Plaisant, Peter David, and Jim Nolan for their input throughout the design and implementation of this project and thoughtful reviews of this paper. Additionally, we would like to thank Jianyu (Leo) Li and Panagis Papadatos for their assistance in designing, developing, and evaluating the initial version of the tool.

References

1. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proc. 23rd International Conference on Machine Learning. pp. 113–120. ACM (2006)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3, 993–1022 (2003)
3. Bostock, M.: Data Driven Documents (d3). <http://d3js.org> (2012)
4. Cui, W., Liu, S., Tan, L., Shi, C., Song, Y., Gao, Z., Qu, H., Tong, X.: TextFlow: Towards better understanding of evolving topics in text. *IEEE Transactions on Visualization and Computer Graphics* 17(12), 2412–2421 (2011)
5. Hart, S., Staveland, L.: Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Human Mental Workload* 1, 139–183 (1988)
6. Havre, S., Hetzler, B., Nowell, L.: ThemeRiver: Visualizing theme changes over time. In: Proc. IEEE Symposium on Information Visualization. pp. 115–123 (2000)
7. Hu, Y., Boyd-Graber, J., Satinoff, B., Smith, A.: Interactive topic modeling. *Machine Learning Journal* (2013)
8. Kleinberg, J.: Bursty and hierarchical structure in streams. In: Data Mining and Knowledge Discovery. pp. 373–397 (2003)
9. Kullback, S., Leibler, R.A.: On information and sufficiency. *Annals of Mathematical Statistics* 22, 49–86 (1951)
10. Leskovec, J., Backstrom, L., Kleinberg, J.: Meme-tracking and the dynamics of the news cycle. In: Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 497–506 (2009)
11. Lin, J.: Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory* 37(1), 145–151 (1991)
12. Liu, Y., Niculescu-Mizil, A., Gryc, W.: Topic-link LDA: joint models of topic and author community. In: Proc. 26th Annual International Conference on Machine Learning. pp. 665–672. ACM (2009)
13. Malik, S., Smith, A., Hawes, T., Dunne, C., Papadatos, P., Li, J., Shneiderman, B.: Topicflow: Visualizing topic alignment of twitter data over time. In: The 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining
14. Mimno, D., McCallum, A.: Organizing the oca: learning faceted subjects from a library of digital books. In: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries. pp. 376–385. ACM (2007)

15. Nikulin, M.: Hazewinkel, Michiel, Encyclopaedia of mathematics : an updated and annotated translation of the Soviet "Mathematical encyclopaedia. Reidel Sold and distributed in the U.S.A. and Canada by Kluwer Academic Publishers (2001)
16. O'Brien, W.L.: Preliminary investigation of the use of Sankey diagrams to enhance building performance simulation-supported design. In: Proc. 2012 Symposium on Simulation for Architecture and Urban Design. pp. 15:1–15:8. Society for Computer Simulation International (2012)
17. Ramage, D., Hall, D., Nallapati, R., Manning, C.D.: Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In: Proc. 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1. pp. 248–256. Association for Computational Linguistics (2009)
18. Shuyo, N.: LDA implementation. <https://github.com/shuyo/iir/blob/master/lda/lda.py> (2011)
19. Sopan, A., Rey, P., Butler, B., Shneiderman, B.: Monitoring academic conferences: Real-time visualization and retrospective analysis of backchannel conversations. In: ASE International Conference on Social Informatics. pp. 63–69 (2012)
20. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison Wesley, 1 edn. (2005)
21. Teh, Y., Jordan, M., M.J., B., D.M., B.: Hierarchical Dirichlet processes. 101, 1566–1581 (2006)
22. Wang, X., McCallum, A.: Topics Over Time: a non-markov continuous-time model of topical trends. In: Proc. 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 424–433 (2006)
23. Wilbur, W.J., Sirotkin, K.: The automatic identification of stop words. Journal of information science 18(1), 45–55 (1992)
24. Zhai, K., Boyd-Graber, J., Asadi, N., Alkhouja, M.: Mr. LDA: A flexible large scale topic modeling package using variational inference in mapreduce. In: ACM International Conference on World Wide Web (2012)