TopicFlow: Visualizing Topic Alignment of Twitter Data over Time

Sana Malik*, Alison Smith*†, Timothy Hawes†, Panagis Papadatos*, Jianyu Li*, Cody Dunne*, Ben Shneiderman*
*University of Maryland, College Park, MD 20740 USA
†DECISIVE ANALYTICS Corporation, Arlington, VA 22202 USA
{maliks, amsmit}@cs.umd.edu, timothy.hawes@dac.us, squawk@umd.edu, {jli, cdunne, ben}@cs.umd.edu

***Abstract*—Social media, particularly Twitter, provides an abundance of real-time data. To account for this volume, researchers often use automated analysis and visualization techniques to produce a high-level overview of a Twitter stream. Existing techniques for understanding Twitter data make use of hashtags or word-pairs and may ignore the complex trends in discussions over time. To remedy this, we present an application of statistical topic modeling and alignment (binned topic models) to group related tweets into automatically generated topics and TopicFlow, an interactive tool to visualize the evolution of these topics. The effectiveness of this visualization for reasoning about large data sets is demonstrated by a usability study with 18 participants.**

I. Introduction

Twitter users produce half a billion tweets per day [1]. The subject matter discussed on Twitter is diverse and users often react to current events as they happen, so the content is constantly evolving. Social scientists and researchers are interested in performing analysis of Twitter data to gain insights from broad trends to detailed user preferences. Processing Twitter data presents many challenging problems, two of which we focus on in this paper: extracting useful information from a medium that limits users to 140 characters can be difficult, and it is necessary to display the results of processing in a way that supports in-depth analysis. Traditional analysis of Twitter data focuses on frequencies of words, word pairs and hashtags, which prevents identifying and analyzing complex topics of discussion. While this type of analysis may capture specific events such as a presidential debate or sequestration, it is much harder to capture a general idea, in this case politics, with a single bigram

Statistical topic modeling is a technique for discovering the "topics" that occur in a collection of documents. We present binned topic models, a novel application of statistical topic modeling, which highlights complex topics of discussion to empower deeper insight into events as they are occurring. Binned topic models go beyond simple statistical topic models by accounting for changes in topics across time, as well as identifying the emergence of new topics within the time range. We apply binned topic models to Twitter data to model its constantly evolving and diverse nature. Binned topic models are topic models generated independently for adjacent time slices of Twitter 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 demonstrate a solution to this problem with our visualization tool, TopicFlow, which visualizes the emergence, convergence, and divergence of complex topics in a Twitter stream.

In this paper, we

1) Describe a novel analysis technique for Twitter data over adjacent time slices, *binned topic models and alignment*, which is an application of LDA 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 into events that would otherwise go unnoticed, and
3) Present an evaluation of TopicFlow with 18 participants that shows its usefulness for following the flow of topics on Twitter.

II. Related Work

TopicFlow covers two main areas: topic modeling to automatically generate topics from a high volume of tweets and visualizing topic trends over time.

A. Topic Detection

Existing tools follow trends in user-generated web content, however, these either only deal with short phrases [2] or are primarily concerned with locating spikes in activity rather than analyzing the trend throughout the full time range [3].

Latent Dirichlet Allocation (LDA) [4] 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 [5] is a topic model that captures time jointly with word co-occurence 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. [6] 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.

B. Trend Visualization

Many existing visualizations explore the domain of social media. Much of this related work focuses on analyzing Twitter hashtags. Conference Monitor [7] performs hashtag analysis over time to analyze the trends of discussion at academic conferences. Spark Clouds [8] integrates spark lines into tag clouds in order to convey trends between multiple tag clouds. Nokia Internet Pulse [9] visualizes the evolution of a discussion on Twitter with a time series of stacked tag clouds. A similar tool, FeatureLens [10], visualizes the evolution of patterns in any text collections. FeatureLens allows for the exploration by frequent patterns found in the text, where the patterns are frequently used words or phrases.

The primary motivation for TopicFlow is to go beyond hashtags by analyzing the evolution of discovered topics for Twitter streams. Many existing topic model visualizations, such as ParallelTopics [11], TopicViz [12], or topic model visualization [13] are not particularly well suited for visualizing topic evolution. Other tools exist for visualizing the "flow" of texts and temporal data [14], [15], [16]. However, these do not account for merging and splitting flows.

Two trend visualizations that are closely related to TopicFlow are ThemeRiver and TextFlow. ThemeRiver [17] 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 [18], 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 with 15 topics. Also TextFlow inhibits access to the underlying data, which limits analysis.

III. Binned Topic Models

In this section, we present the application of LDA to a corpus of tweets *binned* into time slices followed by the alignment of the topics produced for the bins. To begin, the tweets are divided into bins; the number of bins is specified as an input parameter. Each bin represents a time slice of equal length with no restriction on the number of tweets it may contain.[1] TopicFlow uses an open-source LDA implementation [19]. Standard LDA requires as input the documents (tweets) and the number of topics [2] to discover, although algorithms exist to automatically determine an appropriate number of topics based on the data [20]. In binned topic models, LDA is applied independently for the tweets of each bin.[3] The algorithm employs a stop words list to remove common words that do not contribute significant meaning to topic modeling.[4]

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). Number of topics depends both on bin size–larger bins will typically contain more topics–and the level of topical detail the user 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 the user with the ability to select a topic of interest and see all corresponding tweets. To enable this, each tweet was assigned to the topic resulting in the highest $P(topic|doc)$. Additionally, in presenting this information to the user, we rank the tweets by probability, such that tweets 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 tweets across topics.

The topics generated at individual bins do not directly correspond to each other, so an alignment step is necessary. 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 vectors.[5] This metric is regularly used for the comparison of documents or the cohesion of clusters in the fields of text and data mining, respectively [21], and has also been previously used for the comparison of topics produced by LDA [22], [23]. While many metrics exist specifically for measuring the similarity or divergence between two probability distributions [24], [25], [26], 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.[6] Instead of assigning the one most similar topic at time $n+1$ for each topic at time n, we present links for any

[1] In future versions a non-parametric modeling approach or an approach based on expected tweet rate could be used to determine the bin size.

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

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

[4] The TopicFlow stop words list contains standard English, Twitter-specific (rt, retweet, etc.), some Spanish words, as well as the query terms used in data collection.

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

[6] 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.

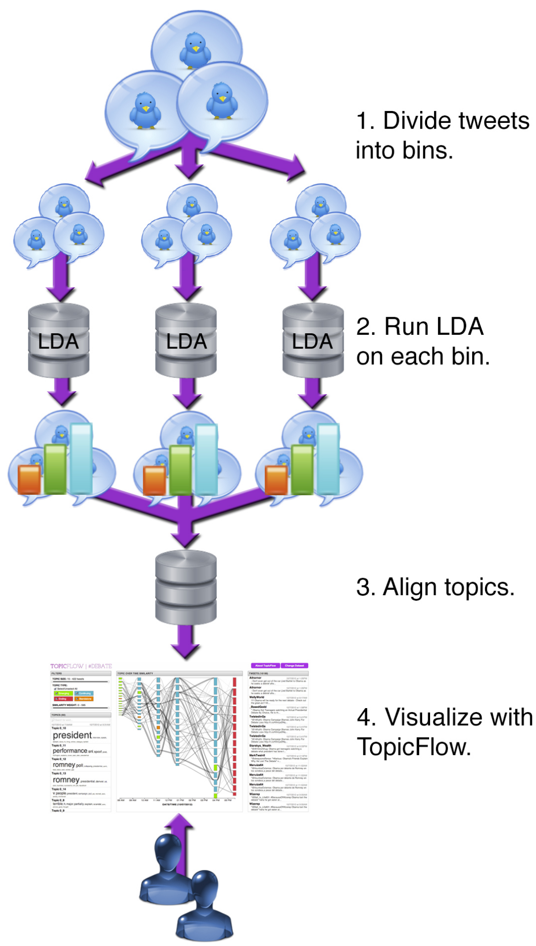

Fig. 2: System overview of the TopicFlow System. The system ingests a dataset of tweets for a given time range and splits the tweets into time slices, applies LDA at each time slice, and aligns the resulting topics from neighboring time slices. The results are then presented to the user through an interactive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

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.[7]

IV. TopicFlow

The purpose of TopicFlow is to allow interactive exploration and analysis of the evolution of topics generated from Twitter data. Figure 2 provides an overview of the TopicFlow system.

A. Design Methodology

TopicFlow[8] visualizes the evolution of topics of discussion for Twitter, and was designed to support six primary use cases:

1) Easily identify the most popular topics within each time slice. A topic is considered more popular if there are more tweets 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.
   - *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 tweets, 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, tweet list, and filter panel, which provide for detailed analysis of the topic trends and underlying data.

During development, TopicFlow was tested on a variety of 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 approximately 1,500 tweets, except the Presidential debate set, which contained about 16,000 tweets [9]. The tweets were collected over varying time spans. We found that the binned topic models were most accurate and concise for real-time events which occurred over short time spans. Alternatively, more general data sets, such as University of

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

[8] TopicFlow is available for demo here: http://www.cs.umd.edu/~maliks/topicflow/TopicFlow.html

[9] 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

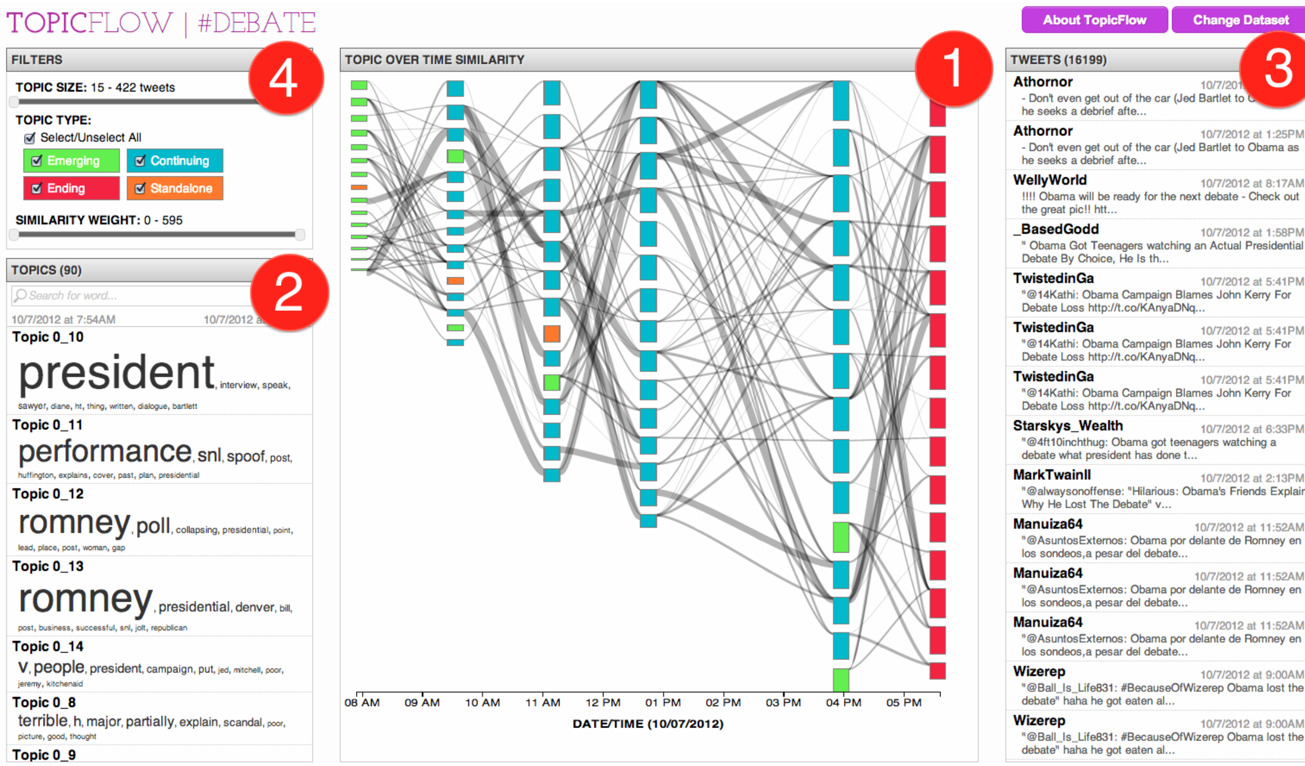

Fig. 1: TopicFlow consists of four coordinated windows: (1) the TopicFlow diagram, (2) a list of topics with their word summaries, (3) a list of the tweets in the dataset, and (4) a filter pane.

Maryland, did not have clearly defined or correlated topics due to the high number of diverse events that occur on the campus.

B. Flow Diagram

The TopicFlow visualization employs a Sankey diagram [27] implemented in the Data-Driven Documents library [28] 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 nodes are sized by the number of tweets attributed to the topic, and they are ordered horizontally from the top by decreasing size. Therefore, the most prevalent topics are at the top of the graph, and the user can quickly see how the frequency of a topic evolves over time. 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. Color is used in the graph to distinguish topics by their evolution state: emerging, ending, continuing, or standalone. 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.

C. Topic Panel, Tweet Panel, and Filter Panel

The topic panel contains a visual representation of the topics discovered for the data grouped by the corresponding time slice. The tweet panel contains the underlying tweet data. For each tweet, the user can explore the full text of the tweet, follow a link to the author's Twitter page, and view a histogram of the five topics with the highest probability for the tweet, $P(topic|tweet)$. Finally, the filter panel, which was designed in support of Use Case 6, allows for filtering the topics by size (number of tweets) or evolution state as well as filtering the edges by similarity.

D. 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 provides the following interactive elements:

1) Topic Search: A search functionality for locating topics containing particular keywords. The topics resulting from the search are highlighted in the topic panel as well as the flow graph. This functionality supports Use Case 4.

2) Graph Highlighting: When a topic is selected in either the topic panel or the flow graph, the corresponding node and topic are highlighted. Also, in the flow graph, 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). Finally, to support Use Case 3, the tweet panel is filtered to show the ranked list of tweets for the topic.

3) Topic Comparison: When an edge is selected within the Flow Graph, a topic comparison box is displayed which uses mirrored histograms to highlight the words the two topics have in common, which supports Use Case 5 (Figure 4).

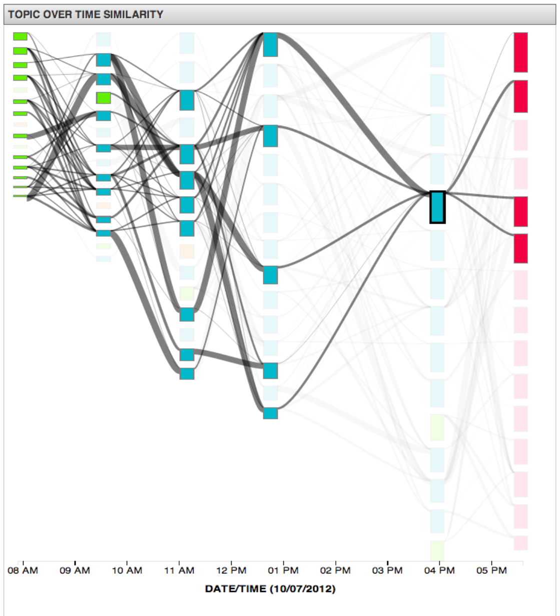

Fig. 3: When a topic is selected, the graph is highlighted to show the flow of that topic over time.

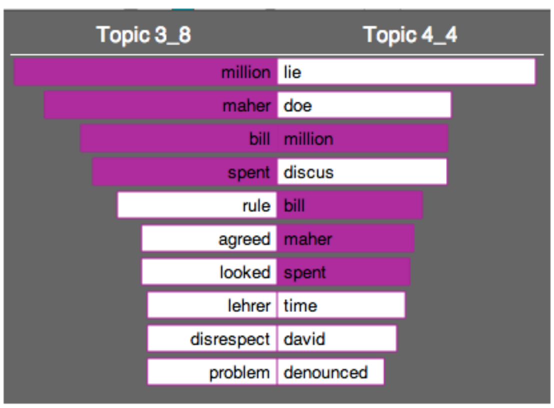

Fig. 4: The topic comparison box shows bar charts representing the two topics connected by the edge. The top words that the topics have in common are highlighted in purple.

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

5) 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 tweets) 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.

V. Evaluation

A. Method

To ensure the usability of TopicFlow for exploring Twitter data sets, 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 experience using a computer, and the rest had 11 or more years experience. Participants had varying levels of familiarity with Twitter: three of the participants used Twitter for six to ten hours a week, three others used Twitter for two to five hours a week, and the rest used twitter for less than 2 hours per week. 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 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 [29]: 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.

B. Results

The means and standard deviations of 18 participants on time, performance, effort, frustration, and effectiveness (Table I) 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),

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 I: Time, Performance, Effort, Frustration, and Effectiveness results 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)

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 tweet 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 tweet list to be re-sorted or if retweets were aggregated and displayed only once.

1) 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 and 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 performed consistently the worst, 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 tweet list pane was their least favorite feature and requested methods for sorting the tweets 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 re-tweeted tweet in the tweet 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.

VI. Future Work and Conclusion

Future work for TopicFlow includes extending binned topic models for other data types and modifying the interface to address feedback received from usability study. Although we use time slices of Twitter data for the purpose of this application, binned topic models is a general technique that can be applied to any data source, including other text streams such as chapters of a book as in the work by [30]. Additionally, the binning method will work with alternative 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 [31], 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 tweet list pane that can be incorporated into future work. Most notably, users requested a way to sort tweets by various metrics such as time, the retweet count, or the number of followers of the user.

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 [32]. 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 tweets 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 Twitter data, which allows for richer analysis of "topics" within the data, beyond just single words or hashtags. 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 tweets relating to a particular topic over time. TopicFlow further facilitates Twitter 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.

VII. Acknowledgements

We would like to thank 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.

References

[1] J. Martinez. (2012) Twitter CEO Dick Costolo reveals staggering number of tweets per day. http://www.complex.com/tech/2012/10/twitter-ceo-dick-costolo-reveals/-staggering-number-of-tweets-per-day.

[2] J. Leskovec, L. Backstrom, and J. Kleinberg, "Meme-tracking and the dynamics of the news cycle," in *Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 497–506.

[3] J. Kleinberg, "Bursty and hierarchical structure in streams," in *Data Mining and Knowledge Discovery*, 2003, pp. 373–397.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[5] X. Wang and A. McCallum, "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*, 2006, pp. 424–433.

[6] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proc. 23rd International Conference on Machine Learning*. ACM, 2006, pp. 113–120.

[7] A. Sopan, P. Rey, B. Butler, and B. Shneiderman, "Monitoring academic conferences: Real-time visualization and retrospective analysis of backchannel conversations," in *ASE International Conference on Social Informatics*, 2012, pp. 63–69.

[8] B. Lee, N. H. Riche, A. K. Karlson, and S. Carpendale, "SparkClouds: visualizing trends in tag clouds," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1182–1189, 2010.

[9] J. J. Kaye, A. Lillie, D. Jagdish, J. Walkup, R. Parada, and K. Mori, "Nokia internet pulse: a long term deployment and iteration of a twitter visualization." ACM, 2012, pp. 829–844.

[10] A. Don, E. Zheleva, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman, and C. Plaisant, "Discovering interesting usage patterns in text collections: integrating text mining with visualization," in *Proc. 16th ACM conference on Conference on Information and Knowledge Management*. ACM, 2007, pp. 213–222.

[11] W. Dou, X. Wang, R. Chang, and W. Ribarsky, "Paralleltopics: A probabilistic approach to exploring document collections," in *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2011, pp. 231 –240.

[12] J. Eisenstein, D. H. Chau, A. Kittur, and E. P. Xing, "TopicViz: interactive topic exploration in document collections," in *CHI Extended Abstracts'12*, 2012, pp. 2177–2182.

[13] A. Chaney and D. Blei, "Visualizing topic models." in *International AAAI Conference on Social Media and Weblogs*, 2012.

[14] F. B. Viégas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," in *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, 2004, pp. 575–582.

[15] R. Nallapati, D. Mcfarland, and C. Manning, "TopicFlow Model: Unsupervised learning of topic-specific influences of hyperlinked documents," in *Artificial Intelligence and Statistics*, 2011.

[16] S. T. ORourke, R. A. Calvo, and D. S. McNamara, "Visualizing topic flow in students essays," vol. 3, pp. 4–15, 2011.

[17] S. Havre, B. Hetzler, and L. Nowell, "ThemeRiver: Visualizing theme changes over time," in *Proc. IEEE Symposium on Information Visualization*, 2000, pp. 115–123.

[18] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong, "TextFlow: Towards better understanding of evolving topics in text," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2412 –2421, 2011.

[19] N. Shuyo. (2011) LDA implementation. https://github.com/shuyo/iir/blob/master/lda/lda.py.

[20] Y. Teh, M. Jordan, B. M.J., and B. D.M., "Hierarchical Dirichlet processes." vol. 101, pp. 1566–1581, 2006.

[21] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, 1st ed. Addison Wesley, 2005.

[22] Y. Liu, A. Niculescu-Mizil, and W. Gryc, "Topic-link LDA: joint models of topic and author community," in *Proc. 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 665–672.

[23] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "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*. Association for Computational Linguistics, 2009, pp. 248–256.

[24] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[25] M. Nikulin, *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.

[26] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 49–86, 1951.

[27] W. L. O'Brien, "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*. Society for Computer Simulation International, 2012, pp. 15:1–15:8.

[28] M. Bostock. (2012) Data Driven Documents (d3). http://d3js.org.

[29] S. Hart and L. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," *Human Mental Workload*, vol. 1, pp. 139–183, 1988.

[30] T. E. Clement, "A thing not beginning and not ending: Using digital tools to distant-read Gertrude Stein's The Making of Americans," *Literary and Linguistic Computing*, vol. 23, no. 3, pp. 361–381, 2008.

[31] Y. Hu, J. Boyd-Graber, and B. Satinoff, "Interactive topic modeling," Under Review.

[32] K. Zhai, J. Boyd-Graber, N. Asadi, and M. Alkhouja, "Mr. LDA: A flexible large scale topic modeling package using variational inference in mapreduce," in *ACM International Conference on World Wide Web*, 2012.