TopicNets: Visual Analysis of Large Text Corpora with Topic Modeling

BRYNJAR GRETARSSON
JOHN O'DONOVAN
SVETLIN BOSTANDJIEV
and
TOBIAS HÖLLERER
University of California Santa Barbara

ARTHUR ASUNCION
DAVID NEWMAN
and
PADHRAIC SMYTH
University of California Irvine

We present *TopicNets*, a web-based system for visual and interactive analysis of large sets of documents using statistical topic models. A range of visualization types and control mechanisms to support knowledge discovery are presented. These include corpus and document specific views, iterative topic modeling, search, and visual filtering. Drill-down functionality is provided to allow analysts to visualize individual document sections and their relations within the global topic space. Analysts can search across a data set through a set of expansion techniques on selected document and topic nodes. Furthermore, analysts can select relevant subsets of documents and perform real-time topic modeling on these subsets to interactively visualize topics at various levels of granularity, allowing for a better understanding of the documents. A discussion of the design and implementation choices for each visual analysis technique is presented. This is followed by a discussion of three diverse use cases in which *TopicNets* enables fast discovery of information that is otherwise hard to find. These include a corpus of 50,000 successful NSF grant proposals, 10,000 publications from a large research center, and single documents including a grant proposal and a PhD thesis.

Categories and Subject Descriptors: I.2.7 [**Text analysis**]: Natural Language Processing; H.5.3 [**Web-based interaction**]: Group and Organization Interfaces

Additional Key Words and Phrases: Topic Modeling, Text Visualization, Graph Visualization

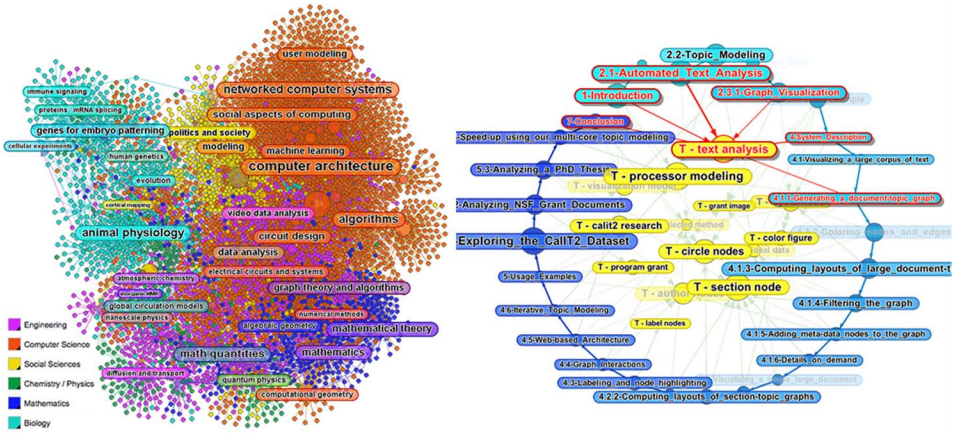


Fig. 1: LEFT: A graph of topics of over 10,000 research articles crawled from faculty pages at CalIT2. Clusters of well-known research fields occur naturally based on salient topics within the modeled documents, revealing interdisciplinary relations such as the blue Mathematics nodes spanning into the Biology cluster. RIGHT: A *TopicNets* graph of the present paper, which reveals the sections related to the "text analysis" topic.

1. INTRODUCTION

The unprecedented growth in the amount of text data accessible by the typical Web user highlights the need for more efficient and more powerful data exploration tools. In this paper, we demonstrate that interactive visualization of salient topics in large collections of text documents can provide useful insight in a manner that complements traditional data exploration mechanisms such as keyword-based search. Visualization can guide iterative topic modeling for overview and detail analysis. Since humans are unable to process large numbers of documents all at once, we focus on efficient ways to navigate, group, organize, search and explore these sets.

To this end, we visualize documents and topics as nodes in a node-link graph (see Figure 1). This representation is very flexible, adapts well to these different information understanding tasks, and our layout techniques ensure that similar topics are positioned close to each other and documents are positioned close to related topics.

In this paper, we define topics as focused probability distributions over the words in a set of documents, as learned from data via statistical topic modeling (also known as Latent Dirichlet Allocation) [**?**]. By leveraging topics as a means to link text documents, we create the basis of an informative, coherent, flexible and *reproducible* visualization, over which a user can interact to discover information in the data from many perspectives. Existing topic modeling algorithms tend to be run offline and produce results for later analysis. A key contribution of this work is that TopicNets supports fast, iterative topic modeling which can be run directly from the interactive visualization in near real time to provide better insight into subsets of interest within the larger corpus.

Recent advances in topic modeling [**?**; **?**] have made it possible to learn topic models on very large data sets using distributed computing [**?**] and in near real-time for smaller data sets using a fast collapsed variational inference algorithm known as CVB0 [**?**]. Our system uses a multi-processor version of CVB0 to speed up learning and facilitate interactive *real-time* topic modeling and visualization. Specifically, documents are partitioned across

processors and local CVB0 inference steps are performed on each processor, with sufficient statistics being globally synchronized at each iteration. CVB0 produces a matrix of topic counts for each document, N_{td} and counts of topics assigned to the different words, N_{wt}. These count matrices are converted to conditional probabilities of interest, the distributions of words given topics, $p(w|t)$, and the distributions of topics given documents, $p(t|d)$.

We discuss the design and implementation of *TopicNets*, a web-based topic visualization system for large sets of documents. Figure 2 illustrates the *TopicNets* information flow paradigm, wherein documents, topics, and additional selected semantic entities are treated as connected nodes of different types in an interactive graph. For large document sets (e.g. tens of thousands of research papers or grant proposals) modeling is performed in advance, and iterative modeling is performed on the fly over smaller subsets. During an interactive browsing session, an analyst can invoke a computationally-fast topic-modeling algorithm over selected subsets of documents of interest, unveiling detailed topical connections that may not have been discoverable using the coarser-grain topic models derived from the entire data universe. Facilitating dynamic generation of new topics in this manner allows users to interactively explore topical links between documents in the corpus that are difficult to find with traditional text analysis tools.

In summary, this paper presents a novel visualization system for large text corpora. The key contributions of this paper include novel mechanisms of visualizing topically similar documents. In our approach, documents and topics are laid out as a node-link graph. A novel aspect of the layout includes the use of topic similarity to determine node positions, thus creating visual clusters of topically similar documents. The techniques described in this paper also enable visualization of topic similarity among individual sections of a single large document. An additional contribution of this work is the use of fast topic modeling techniques to facilitate iterative topic modeling and visualization of subsets of a large data set. Moreover, the described implementation is both flexible and web accessible, allow it to be easily deployed on a broad scope of document sets across multiple domains.

Section 2 discusses related work in the field of topic modeling and visualizing topic models, as well as a discussion of relevant graph visualization techniques. Section 3 gives an illustrative example of the main benefits of the system. The core techniques used in *TopicNets* are detailed in Section 4. Section 4.1 describes how *TopicNets* enables visual analysis of a large set of documents. Two usage scenarios of this technique are presented in Section 5.1 and Section 5.2. Section 4.2 describes our techniques for visual analysis of individual documents. These techniques are then deployed in a sample scenario analyzing the structure of a PhD thesis (Section 5.3). Section 6 demonstrates the computational cost involved with the various components of the system.

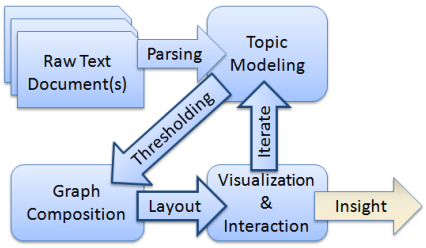


Fig. 2: Information flow in *TopicNets*.

2. BACKGROUND AND RELATED WORK

The literature relevant to this work spans three well-defined but overlapping research areas: automated text analysis, topic modeling and associated information visualization techniques. The following sections describe the state of the art in each of these areas.

2.1 Automated Text Analysis

Visualizing large quantities of text is a challenging problem. Because text can be unordered, ambiguous, abstract and highly combinatorial, many different approaches have

been investigated to support the visualization of text in large sets of documents. These include early approaches such as [**?**; **?**; **?**; **?**] to more recent work in [**?**; **?**; **?**; **?**; **?**; **?**; **?**]. Since humans cannot process a large document set quickly, all of these systems employ some form of filtering mechanism such as a visual overview and then detail on demand [**?**]. Matheo Analyser [**?**] (visual) and Questel's q-pat (text-based) [**?**] are commercial patent search tools that use supporting meta data to refine a result set based on a user's search. *TopicNets* builds on this idea, but contrasts in that search results are visualized in an iterative fashion. None of the previously mentioned systems show explicitly the connections of documents through topics in a single network view. *TopicNets* does this and allows interactions with this visualization to further aid the information discovery process. Text remains the primary communication medium on the web [**?**] and despite a small amount of semantically well-structured text information such as DBPedia [**?**], the majority of text on the web is written in an unconstrained "free form" manner. Because of the high dimensionality, multiple meanings, and complex relations that are typically present in text, text visualization is inherently more challenging than visualization of categorical or numerical data [**?**]. One approach is to use automated summarization techniques as a means to cope with increasingly large document sets [**?**]. Summarization techniques are generally sentence-based or keyword-based. Applied to a global document set, summarization does not directly provide information about relationships between the documents in the set. Since our technique uses generated topics to represent or "summarize" a document, we can use common topics to link similar documents together into meaningful clusters, thereby highlighting information about the broader set that would be difficult to unearth with existing summarization techniques.

A key challenge in the visualization of large text collections is to map highly complex text down to lower dimensional representations while retaining as much of the original meaning as possible. Traditionally, matrix decomposition and factorization methods such as singular value decomposition (SVD), principal component analysis (PCA) and neurocomputation methods such as self organizing maps have been popular for producing visual representations of large bodies of text. For example, Luminoso [**?**] and GGobi [**?**] are text visualization systems that both employ SVD. Computations can be memory intensive for SVD and the resulting topics are not easily interpretable. Other multi-variate analysis techniques are also popular in analysis of large text collections. For example PhraseNets [**?**] supports search for user-provided bigrams (word pairs), which are then used to drive graph visualizations of large texts. The blogosphere and news articles have driven many such innovations in text visualization, producing spacial segmentation visualizations such as NewsMap [**?**], graph and chart-based timelines such as MoodViews [**?**] and ThemeRiver [**?**].

Another class of dimensionality reduction techniques focuses on producing a set of concepts that span across documents and terms within documents. Dessus provides a good overview of these systems in [**?**]. Deerwater's latent semantic analysis (LSA) [**?**] uses a term-document frequency matrix to represent occurrences of a term. LSA techniques have been widely used in text analysis. For example, Fortuna et al. [**?**] use LSA to produce a labeled point for each term and employ multidimensional scaling [**?**] to map the LSA space onto a two dimensional plot. Aside from being computationally complex, a primary shortcoming of LSA is that the results, while making mathematical sense, can be difficult to interpret linguistically [**?**].

Latent Dirichlet Allocation or LDA was forwarded as a more interpretable alternative to LSA, originally as a graphical model in [**?**]. In the text document case, LDA assumes that each document is composed of a number of topics, and each word in the document is attributable to one of those topics [**?**]. While there have been some prior attempts at visu-

Mathematical Theory	theorem lemma proof follow constant bound exist definition
Software Engineering	software process tool project development design system developer
Gene Expression	protein genes expression network motif interaction pathway genome
Politics and Society	political social policy economic china law government national
Business and IT	business firm services customer technology management market product
Fluid Dynamics	flow velocity wall fluid turbulence reynold pressure channel

Table I: Examples of LDA topics learned on CalIT2 research papers

alizing results from LDA algorithms [**?**], most are static visualizations that do not support user interaction. Recent work by Liu et al. [**?**] discusses a more dynamic visualization which shows information from an LDA algorithm, where topics are visualized as a stacked graph and topic strength is mapped to the y-axis of a graph visualization. Liu et al. also discuss a node link diagram, which focuses on relations between a set of analyzed emails. The core contribution of this paper, is a highly dynamic, user configurable visualization that supports *control* of a fast, near real-time topic modeling algorithm through interaction with the visualization. Moreover, the system supports *iterative* topic modeling on selected subsets of interest.

2.2 Topic Modeling

Statistical topic modeling is a widely-used unsupervised machine learning technique for automatically extracting semantic or thematic topics from a collection of text documents [**?**; **?**; **?**]. The topics provide a high-level abstract representation of documents in a corpus, and can be used for searching, categorizing, and navigating through collections of documents. For example, Blei and Lafferty [**?**] show how topic models can be used to explore and browse 100 years of the journal *Science*[1]. The models are based on the assumption that each document d can be represented by a small number of topics t, where each topic is dominated by a small fraction of all possible words. Each topic is modeled as a multinomial distribution over words $p(w|t)$, for $t \in 1 \ldots T$, where T is the number of topics. Topics are often displayed by showing the top-n terms (i.e. the n-most probable terms) in the topic [**?**; **?**], for example the topic: *"software process tool project development design system developer community ..."* clearly relates to the subject of Software Engineering, and the first two or three terms form a fitting descriptive label (see Table I for more examples). Although it is possible to misinterpret such representations they are generally easy to interpret as confirmed by [**?**; **?**; **?**]. By default *TopicNets* displays only the top two words to avoid clutter, but this is user modifiable. Furthermore, there is recent work on automatically labeling topics [**?**] which can also be incorporated into *TopicNets*. There are existing toolkits that deal with topic modeling, such as the Stanford Topic Modeling Toolbox [**?**], which deal with performing inference over topics. *TopicNets* differs from such toolkits in that it provides an interactive graph-based visualization in addition to enabling the user to further drill down on the visualized graphs, e.g., by learning a topic model specifically for the subset of visualized documents.

In the topic model, each individual document d is modeled as a distribution over topics $p(t|d)$ for $d \in 1 \ldots D$, where D is the number of documents in the collection. The topics are a low-dimensional representation for each document, reducing the dimensionality from the vocabulary size (which could be as large as 100,000) to a T-dimensional vector of topics (where T could be on the order of 100 or less). The topic proportions for document d, $p(t|d)$, characterize the topical content of a document. For instance, if there are

[1] `http://topics.cs.princeton.edu/Science/`

$T = 4$ topics, and if $p(t|d) = [0.4, 0.1, 0.45, 0.05]$, then one can infer that this document is mainly comprised of topics 1 and 3. To find documents d_j that are semantically similar to a particular documents d_i, one can use measures such as the symmetric Kullback-Leibler divergence, or the L_1 distance, between the respective topic distributions $p(t|d_j)$ and $p(t|d_i)$.

A topic model is learned from a collection of text documents by approximately inferring the posterior distributions of the parameters of the model, given the observed data (the vectors of word counts for the documents). Recent advances have made it possible to learn topic models on very large data sets using distributed computing [**?**] and in near real-time for smaller data sets using a fast collapsed variational inference algorithm known as CVB0 [**?**]. Our system uses a multi-processor version of CVB0 to speed up learning and facilitate interactive *real-time* topic modeling and visualization. Specifically, documents are partitioned across processors and local CVB0 inference steps are performed on each processor, with sufficient statistics being globally synchronized at each iteration. CVB0 produces a matrix of topic counts for each document, N_{td} and counts of topics t assigned to the different words w, N_{wt}. These count matrices are converted to conditional probabilities of interest, the distributions of words given topics, $p(w|t)$, and the distributions of topics given documents, $p(t|d)$.

The topic model is *unsupervised*—no training data or training labels are required to learn the model. The only input to this algorithm is the set of documents to model, and a number of topics T, changeable in *TopicNets* by the user at run-time.

2.3 Visualizing Topic Data

Text-based topic browsers are often used for interacting with learned topic models. In such browsers, the topics are usually displayed using their top-n words. For each topic, a set of relevant documents can be displayed, and for each document, the topic proportions can also be displayed.

While there have been attempts to visualize document clusters based on a variety of methods [**?**; **?**; **?**], Network-based visualizations of topic models have mostly been limited to static visualization. For instance, Blei and Lafferty produced static graphs of topic relationships with their correlated topic model [**?**]. Newman et al. produced static entity networks to connect entities (such as people and businesses) mentioned in news articles using topics [**?**]. Iwata et al. developed a probabilistic latent semantic visualization model which provides an alternative approach to statically plotting topics [**?**]. A visualization with limited interactivity on software architectures was presented by [**?**]; however, this visualization only displayed the software architecture as a predefined graph, rather than a graph based on the learned topic model. In this paper we have chosen a graph-based representation to display relationships between topics and documents. Accordingly, we now discuss relevant research in graph visualization and in topic visualization.

2.3.1 *Visualizing Document Clusters.* Traditionally, use of document clustering was viewed as an inefficient way to search through large corpuses because of complexity issues [**?**; **?**; **?**; **?**]. However, Cutting et al. [**?**] showed that for certain classes of search tasks, clustering methods proved to be useful. For example, [**?**] shows that when a query is vague, document clustering can help focus the search better than traditional text matching approaches can. Research has shown that network visualizations are helpful for providing an overview of the clustered document space. Several tools build on the initial network visualization research performed by Fairchild et al in [**?**], and Will's addition of interaction techniques for exploring the network [**?**]. For example, InfoSky [**?**] is a tool for visualizing hierarchically structured knowledge spaces in a 2D representation . The analogy of a galaxy and constituent stars were used to represent the corpus and documents. In con-

trast with our approach, InfoSky relies on pre-defined relations between the documents. FacetAtlas [**?**] harnesses pre existing similarities between documents to form a network. For example, the system used common terminology across a set of rich text medical documents to form a network that a user can visually analyze. An advantage of the *TopicNets* tool is that the *relation* need not be pre-defined, and can be computed on the fly, and in an iterative manner based on the particular requirements for a search or discovery task. TopicIslands [**?**] uses a similar technique but represents the document corpus in terms of wavlet transforms. This technique was successful in defining thematic 'channels' to visualize, but had less success with complex writing styles. In addition, [**?**] used novel 3D techniques to perform stereoscopic text visualization, which helped users understand the information space.

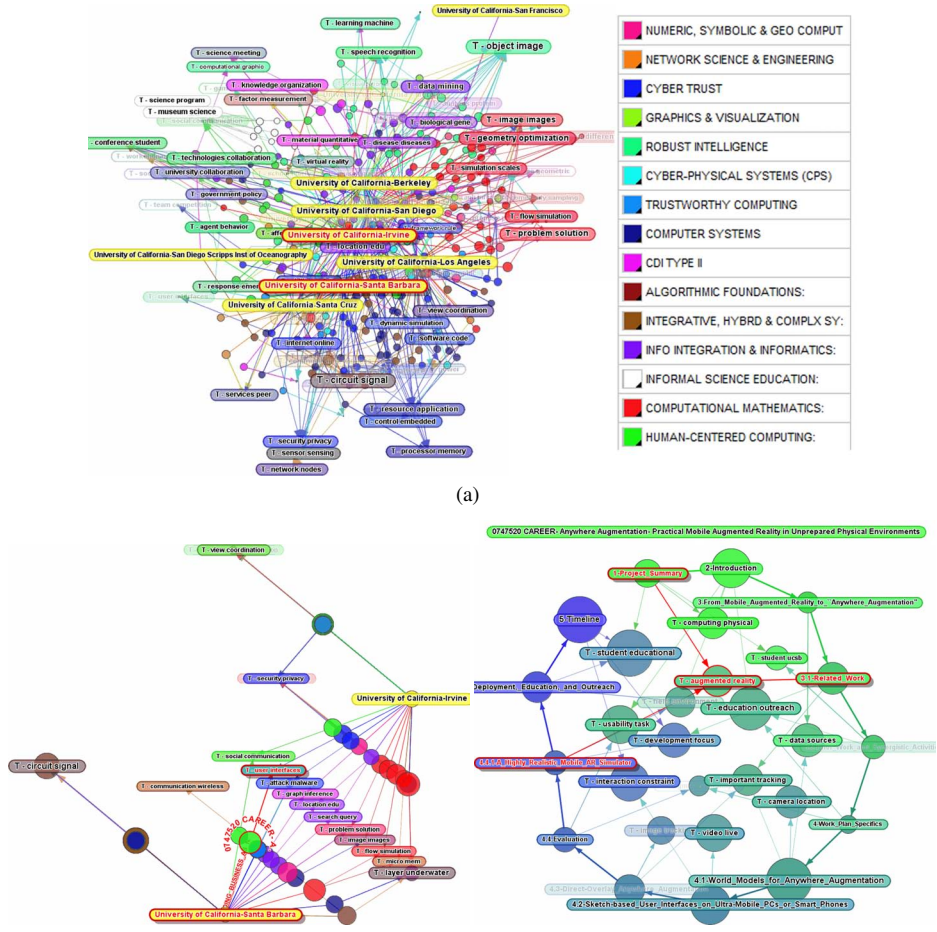
2.3.2 *Graph Visualization.* Prior approaches to topic visualization, such as [**?**] and [**?**] cluster together topically similar documents and don't visualize individual topics explicitly. This is an important distinction for *TopicNets* since we communicate the relations between a set of documents (or parts of documents) and the topics they contain via an interactive graph, consisting of different node types with connecting edges, as illustrated by Figure 1. While there are many other possible approaches for visualizing connectivity data (e.g. [**?**; **?**; **?**; **?**]) we employ an interactive graph because it intuitively displays node connections, cliques, clusters and outliers. Through simple interaction mechanisms (examples discussed in [**?**]), the graph can be molded by analysts into representations of their design. Additionally, the document-topic graphs are conceptually easy to work with, e.g. a user can perturb or "wiggle" a topic or document of interest and easily see the connected entities. Furthermore, graphs facilitate addition of new node types easily and have many visual dimensions that can be used to encode meta-data about the underlying texts and their associations. In addition to the primary graph, *TopicNets* supports a multi-panel interface with a variety of different controls and outputs, for example, document search and text display components.

Much research has been conducted on the visualization of such node-link graphs, e.g. [**?**; **?**; **?**; **?**]. Traditionally, graph visualization applications require a large amount of processing power and have been desktop based (e.g., Cytoscape [**?**], Pajek [**?**], Tulip [**?**]). Recently, increased web-accessibility and bandwidth improvements have triggered a general shift towards web-based graph visualization tools, capable of providing interactive and responsive graph interfaces through a web browser. Examples include IBM's Many Eyes [**?**] and Tom Sawyer Visualization [**?**]. *TopicNets* is deployed as a native web-based application built on top of the WiGis framework [**?**] and builds on a range of popular graph interaction and layout algorithms, including a fast, scalable interaction algorithm adapted from original work in [**?**]. Several constrained force directed layout techniques [**?**; **?**; **?**] are implemented in *TopicNets*, inspired by Dywer's work in [**?**]. A multidimensional scaling method similar to that discussed in [**?**] is used to visualize relations between topic nodes, as shown by the example on the left of Figure 1.

3. ILLUSTRATIVE EXAMPLE

TopicNets[2] provides ways to interactively explore topics of multiple documents while making it easy to drill down into subsets of documents and even going all the way into individual documents and visualizing the topic-section relationships. Figure 3 shows this flexibility of the system. In Figure 3a) we show NSF grants related to computer science that were awarded to any University of California campus along with the associated topics.

[2] `http://www.wigis.net/topicnets`



(a) (b) (c)

Fig. 3: University nodes are displayed in yellow. Topic nodes are labeled with "T - ..." and colored according to the connected documents.(a) A visualization of grants relating to computer science that were awarded to any University of California campus and the related topics. UCSB and UCI have been highlighted in this image. (b) NSF grants that were awarded to UCSB or UCI after the user has performed a couple of mouse drags using our interaction technique. In this image we have highlighted one particular grant which is then visualized in the next image. (c) A visualization focused on one NSF grant "CAREER- Anywhere Augmentation- Practical Mobile Augmented Reality in Unprepared Physical Environments". The section nodes go in a circle on the outside of this image, while the topic nodes are inside. We have highlighted one topic node and the connected sections.

Additionally we have nodes representing each of these campuses in yellow. Because topics are laid out based on their similarity and their positions in turn govern the positions of other nodes, we can infer which topics are popular for each campus based on their position within this graph. We have highlighted the two campuses that the authors of this paper are associated with. Then in Figure 3b) we focus in on these two campuses and remove grants not related to them. Using the interpolation method discussed in [**?**] we can easily arrive at the image presented in Figure 3b) with a couple of mouse drags. This image clearly

illustrates the topics that these two campuses have in common along the diagonal halfway between the two yellow university nodes. It also reveals other clusters of nodes, e.g. the diagonal just above University of California Santa Barbara contains grants awarded to UCSB that have topics in common with grants awarded to UCI. We have highlighted one of these nodes and in Figure 3c) we focus on the contents of this grant, iteratively refining the topic model to focus on topics more suited to the individual grant. Sections of the document are arranged in a circle starting at the top left and traversing clockwise from section to section in the order that they occur in the document. The color in this case is blended from the color that the document had in 3b) (i.e. green), towards blue at the last section.The user has the option of blending the color from the sections into the topic nodes or assigning their own custom color to topic nodes. In Figure 3c) color is blended from connected sections into the topic nodes, meaning that a green topic is mostly associated with sections at the start of the document. The topic nodes are placed inside the circle and get attracted to the associated sections. When a topic node is selected, like in Figure 3c) the associated sections get pulled towards the center of the circle to highlight their association with the particular topic of interest.

4. SYSTEM DESCRIPTION

Before we present our use cases, this section describes the design choices and methodology used to produce visualizations in *TopicNets*. Section 4.1 deals with visualization of a large corpus of text, in terms of graph generation, color mappings and choice of layout. Following this, a discussion of various filtering mechanisms is presented. Section 4.2 covers our design choices for focusing in on a single document. This is followed by a brief discussion of the supporting web-based architecture.

4.1 Visualizing a large corpus of text

Extracting and presenting relevant information from a large text corpus quickly is a non-trivial task [**?**] which we attempt to address with *TopicNets*. The system aids in the analytic process and facilitates easier discovery of information in large text corpora, by supporting iteratively refined topic models, controlled and guided by user interactions with the visualization.

4.1.1 *Generating a document-topic graph.* An overview of the central idea of this work is shown in Figure 2. The first step in the generation of a document-topic graph is topic modeling of raw text. We start by running a fast collapsed variational inference algorithm (CVB0) [**?**] to learn topics from the selected text corpus. As described earlier, the algorithm learns a topical representation of each document in the form of a probability distribution over topics $p(t|d)$ for each document d, interactively at runtime. To generate a graph visualization of this data we first apply a minimum threshold to $p(t|d)$ which can be altered through a slider in the interface. This threshold is used to determine if an edge should be created between a document and a topic node in the graph. For example, assuming a threshold of 30%, if $p(t|d) > 0.3$ then an edge is created between document d and topic t. Selecting a default threshold value to optimize all graphs is a challenging problem. We set a default simply as the maximum threshold that maintains at least one edge to every topic node in the graph. We believe that this is a reasonable default since it ensures that every topic is represented by at least one document. At the same time, this default value minimizes the probability of popular topics drawing a large number of edges, which can cause visual clutter. The probability of topics occurring in a document, $p(t|d)$, is used to determine the thickness of edges between nodes. For instance, a document-topic edge denoting 70% association probability is notably thicker than an edge with only 20% prob-

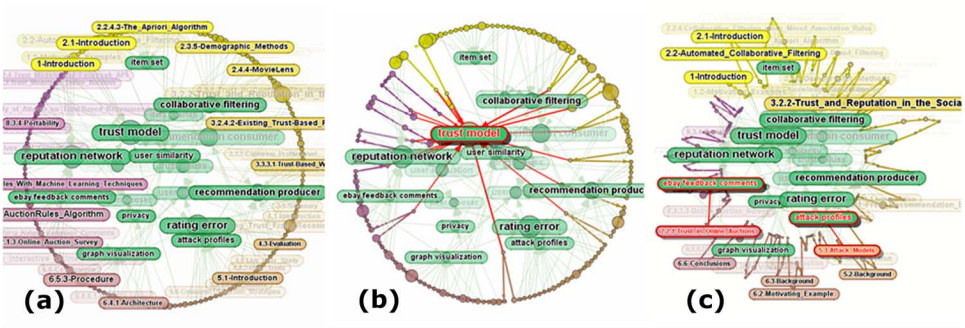


Fig. 4: Three snapshots of topic-based deformation in a section-to-topic layout. Transitions in *TopicNets* are smooth animations. (a.) shows linear structure of the document along the circumference, (b.) shows the distribution of a single topic across the document, and (c.) size and centrality shows the relevance of each section and topic to the central theme of the document.

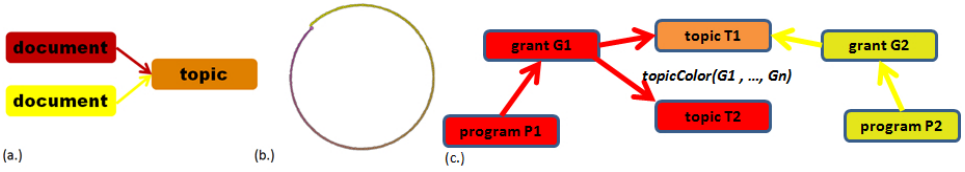


Fig. 5: (a.) Example of color bleeding across document-to-topic edges. (b.) Example of color interpolation across a time series. (c.) Custom color mappings from our NSF use case.

ability. Our preliminary tests indicate that by mapping probability information directly to edge-thickness we help analysts to understand likely topic associations at a quick glance. In addition to the probability of a topic occurring in a given document, the topic model also indicates the overall prevalence of each topic $p(t)$ across the entire corpus. The value of $p(t)$ is mapped directly to topic-node size in the generated graph. This technique produces a useful variety of sizes, with common topics becoming significantly larger than uncommon topics. In turn, document node size is set as a function of document length N_d. By default a topic node is labeled with the first n words of the topic, where n can be specified by the user (default $n = 2$ for labeling clarity). Based on our experiences, these are usually reasonable descriptive labels for a topic. However, in a small number of cases we have asked domain experts to manually label topics.

4.1.2 *Coloring nodes and edges.* Some data sets contain meta information about the documents themselves. In cases where such data is available, it becomes possible to map it to various other dimensions of the visualization. Consider the image on the left of Figure 1 as an example. In this graph, author affiliation in the CalIT2 data set (over 10,000 research papers mined from the faculty pages of CalIT2 members at UCI and UCSD) has been mapped onto color. *TopicNets* provides an interface for mapping meta-data information onto graph features. In this example, the departmental affiliation of each author is mapped onto a smaller number of research fields.

Both colors and mappings can be modified by the user in the interactive interface, resulting in highly customizable graph visualizations, capable of providing many perspectives on the underlying data set. When an author to color mapping has been provided, the system propagates color information to connected documents, e.g. documents written by a green author will be green and color information is bled along the document-topic edges

in the graph, as illustrated in Figure 5(a). There are potentially many incumbent edges to a given topic node. Color information from each document-topic edge is blended to produce a final color for that topic node. The result of this process is shown in Figure 6. Color mapping is flexible in that it allows for placing of topics within any meta data space, in the specific case of CalIT2, shown on the left of Figure 1, color is used to map topics onto a field of study.

Suppose that a user wants to map a continuous meta data variable, such as time of origin, onto document color. To facilitate such mappings, *TopicNets* allows a user to interpolate colors between sequential document nodes. In the example in Figure 5(b), node colors between the first and last document nodes are automatically interpolated relative to their position along the timeline. The colors are interpolated in RGB space which does not produce ideal results for every color pair, however the user can select the two colors in such a way that the interpolation properly portrays the sequence. A user then has the option to choose either a single color for all topic nodes, or to blend colors from connected documents to produce an informative visualization. The first option is demonstrated in Figure 4 while the latter option is shown in Figure 6. In the latter example, topic color gives an indication of the position in the timeline where that topic most frequently occurs.

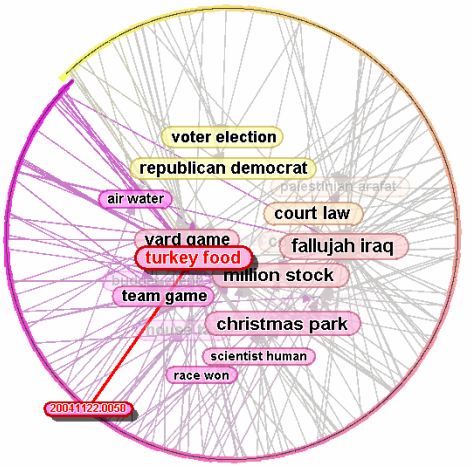


Fig. 6: Example of a *TopicNets* graph showing topics for NY Times articles for the month of November 2004. Documents are ordered by time and colored based on their position on the timeline. Topic nodes are colored based on the connected documents. "Turkey food" proves to have a lot of interest all month, and especially around Thanksgiving, whereas election related topics peak around election day.

4.1.3 *Computing layouts of large document-topic graphs.* We now discuss two distinct layouts used in *TopicNets* to illustrate relations between the documents: A layout based on topic similarity, and a layout that preserves different forms of linear structure in a single document or document set, for example, time of creation or section ordering.

4.1.3.1 *Topic-similarity layout.* To maintain an aesthetically pleasing layout while preserving information about topics produced by the LDA algorithm, we compute the symmetric Kullback-Leibler divergence between every pair of word topic distributions $p(w|t)$. The resulting dissimilarity matrix is then used as an input into a multidimensional scaling (MDS) algorithm[**?**], which determines a position for each topic node. Topic nodes are then fixed into position and a standard force-directed layout algorithm[**?**; **?**; **?**] is applied to place the document nodes in this topic space. We use the probabilities described in Section 4.1.1 to determine the optimum distance between document and topic nodes, where a higher probability means the document node is more attracted to a particular topic. It would be possible to add invisible edges between the topic nodes representing their dissimilarity and only run a force directed algorithm, but then we would not be guaranteed that the topic similarity has more effect on the final layout than the other forces in the force directed layout and the initial positions would influence the final layout. Our interactions with the system indicated that semantically interpretable clusters are usually shown,

consisting of topics with similar interpreted meaning and their associated documents.

4.1.3.2 *Order-preserving layout.* Many tasks in visual analysis require ordering of documents in some particular fashion to tell a story about the contents. For example an analyst might want to sort newspaper articles by time of publishing as seen in Figure 6. To facilitate such orderings, an alternate layout method was developed in which documents are laid out along the circumference of a circle with a layout constraint that a pre-specified node ordering is preserved. The circle is slightly spiraling, meaning that each node is slightly closer to the center than the previous one. This technique ensures that the first and last documents don't meet at a single point, as seen in Figure 6. While users might be more used to seeing a timeline as a straight line, our technique produces visuals that are less cluttered since topics can be connected to documents far apart on the timeline. In this layout, document nodes are first fixed in place, and a force directed layout is applied to connected topic nodes. This layout achieves interesting perspectives on the underlying data because document order drives the final positions of the topic nodes. In addition to the influence of document ordering, the layout attempts to place similar topics close to each other. To achieve this, we introduce invisible inter-topic edges which act to keep topics at the distance specified by their dissimilarity. These edges are much more flexible than the document-topic edges and thus don't adversely affect the layout. As illustrated in Figure 6 wherein documents are ordered by time, topics that are mostly used around a certain time period shown on the circumference (e.g.. "race won") gravitate towards that area of the visualization, while topics that are used all across the corpus (e.g. "turkey food") move towards the center of the visualization. A potential drawback of this approach occurs when a topic is only connected to two documents which happen to be on opposite sides of the circle. In such cases, the topic will be drawn close to the center of the circle, even though it is not necessarily central to the theme of the document. To counterbalance this issue, topic popularity is mapped onto node size, meaning that smaller topic nodes are not so relevant across the entire document. This approach produces a final visualization in which topic node position represents centrality of a topic and topic node size represents frequency of that topic, so a large topic node in the center of the graph should be the topic most relevant to the entire corpus. In case there are multiple nodes close to the center and their labels are overlapping, then the largest (and most popular) topic is always displayed. Interaction makes it easy for the user to zoom into a cluttered area of the graph to reveal more labels or simply to select the smaller topics to reveal their labels. This layout is also shown in Figure 3(c.).

4.1.4 *Filtering the graph.* In many cases an analyst is interested in a target subsection of the graph. *TopicNets* provides multiple methods for filtering the graph in order to get to a visualization of the target information. The user can type in any text and perform a search over the node labels, the top 10 words of all the topics, and/or entire documents. Once the search has completed the user can select a few of the returned nodes from a list, or simply select them all. Once the user has selected a set of nodes of interest he/she can click on a button to visualize only the selected nodes and their immediate neighbors in the graph. Once that button is pressed, all the other nodes are removed from the graph and a new layout is computed. The system provides a smooth transition from the old layout (with unrelated nodes removed) to the new one. Figure 7 shows the results of a search for "genetics" over the entire data set shown on the left of Figure 1. In case the user made a mistake or wants to explore another subset of the graph they can easily go back to the previous visualization with the click of a button. This filtering can also be repeated multiple times to find a subset within the subset. Topic modeling can be used alternatively to iteratively cluster the selected set based on topic associations. This is discussed in more

detail in section 4.6.

An alternative filtering method we provide is to manually select one or more nodes in the graph and then make the system remove all nodes not in or connected to the selected nodes. Figure 8 d) shows the results of selecting two nodes ("Tatiana D. Korelsky" and "Douglas H. Fisher") in the graph shown in Figure 8 c) and asking the system to show only nodes within graph distance of two from the selected set. *TopicNets* also provides an easy method of expanding the selected set by one edge from every node in the selected set. This can be a very powerful feature and for example it gives the user an easy way of getting to a graph of a selected topic, all the documents connected to that topic, and all the topics that those documents are connected to. The selection expanding can be repeated multiple times, until all the nodes in that connected component have been selected.

A third method for filtering the data is to use author information (in a different way than just searching for the author names). If author information for the documents in the corpus is available then our system gives the user the option of collapsing document nodes into author nodes resulting in an author-topic graph. This graph is then laid out using the *topic-similarity layout* showing where authors "live" in topic space. Figure 7 shows the results of the search query "genetics" over the CalIT2 data set after collapsing the document nodes into author nodes.

4.1.5 *Adding meta-data nodes to the graph.* So far we have only discussed visualizing graphs with two types of nodes, document nodes and topic nodes. In many cases there is additional information available about the documents, for example author name, author institution, etc. *TopicNets* keeps all that information and makes it accessible through a detail panel when a document node is selected as described in the following section. The previous section explains how we can collapse all documents by the same author into individual author nodes as a way of filtering the graph. Additionally, *TopicNets* allows the user to select any available meta-data field and generate additional nodes for them. These nodes are then positioned in the graph using a simple force directed layout. The user can then end up with a graph containing multiple node types, for example topics, documents, authors, and organizations, showing how documents and topics are connected through these other entities. In our example of the NSF grants dataset, we have a lot of meta-data for each grant and adding those entities onto the graph as nodes can lead to interesting insights about the data as detailed in Section 5.2. These additional nodes can be generated at any stage of the analysis process, i.e. on the entire dataset or after performing any of the filtering methods described in the previous section. The user can also choose a color from the interface to map onto these new nodes, e.g. make authors green and organizations blue.

4.1.6 *Details on demand.* When a user selects a node in the graph all the details of that node can be seen in a panel on the right hand side. If the selected node is a document node, a link to view that document is provided. If the selected node is a topic node then links to view all the connected documents are provided. Additional options such as select all neighbor nodes and delete node are also provided on this panel. If section information about the selected document node is available then the user is given the option to expand that node into its sections. Once that button is pressed, additional nodes representing each of that document's sections are added to the graph and those section nodes are connected to the related topic nodes. This can show the document sections in the context of other documents or we can just look at that document on its own, which leads nicely into the next section.

4.2 Visualizing a single large document

Visualizing a large text corpus is an important task, however we believe that taking it a step further and visualizing the contents of individual documents can help information discovery even more. After all, a user can find the document that he/she is interested in, but if that document is dozens or even hundreds of pages long then it would certainly be helpful to visualize the contents of that document to find the most interesting sections. In addition to all the features available for a large corpus there are some other features that can help the user better understand an individual document. In this section we will describe these additional features for visualizing individual documents.

4.2.1 *Generating a section-topic graph.* When visualizing a single document and its topics we need to first divide that document into its sections. In many cases we had the LaTeX source of the documents, which made this a simple process. However, in some cases we had to analyze the structure of the documents further to figure out where to split it into sections. Once we have split the document into it's sections we need to run a topic detection on the sections. The resulting topic model can then be used to connect sections of the document to topics, just like we did for whole documents before. One important difference here is that the sections have an inherent ordering among them, namely the order in which they appear in the document. In our visualization we represent this linear structure of the document by connecting each section node with a directed edge to the following section node. These edges are also made rather thick, compared to the topic edges so that they will be easily visible regardless of the layout method used. Here we also use the same coloring scheme as described in the second paragraph of Section 4.1.2, i.e. the user can specify a color for the first and the last section and then the color fades from the starting color to the finishing color along the sections. Figure 4 shows an example of a single document visualization. In this image the topic nodes are green while the section nodes fade from yellow to purple.

4.2.2 *Computing layouts of section-topic graphs.* We use the inherent ordering on the sections in a document to lay out the section nodes in a circle similar to the "Order-preserving layout" described in Section 4.1.3.2. Again, the circle is slightly spiraling to avoid the two ends meeting in one point. Like before, the topic nodes are positioned by a force directed layout algorithm with the addition of (weaker) topic-topic forces attempting to keep the related topics close to each other. Figure 4 a) demonstrates this layout technique. There are a couple of things that are different in this case though. First, the user can select one or more topic nodes and then the circle will deform so that all the sections connected to the selected topic node get pulled toward that node as shown in Figure 4 b). If the user selects multiple topic nodes then this will happen to all the sections connected to either one of them. The amount that they can move is defined by a user specifiable parameter. There is one important restriction on the section node movements – they are only allowed to move along an axis defined by the center of the circle and the center of the section node. This means that the section nodes always maintain their order along the circle and only move either towards or away from the center. Additionally this ensures that section-section edges will never cross and the linear structure of the text is maintained. Second, as shown in Figure 4 c) the user can give all the document nodes the freedom to move along their axes which can totally deform the shape of the circle, while still maintaining the linear structure of the document since there will be no crossing of section-section edges. In the resulting graph, the section nodes connected to the most central topics of the document will get pulled towards the center while the sections connected to less common topics will float to the periphery along with those topic nodes.

4.2.3 *Filtering and details on demand.* All the methods used to filter a large corpus can also be used to filter a single document. Similar to the description in Section 4.1.6, if a section node is selected then the node details panel will provide a link to the contents of that section. If a topic node is selected then the details panel will provide links to view the contents of all the connected sections.

4.3 Labeling and node highlighting

TopicNets supports various types of labels for the document and topic nodes. To distinguish between different node types, *TopicNets* uses icons; for example document nodes are represented with an image of a document and topics are represented with an image of a speach bubble. While we turned this feature off in some of our figures to reduce clutter at the small printed scale, Figure 8 gives an example. Icons become visible during interactive exploration when node sizes increase. Different shapes could also be used to represent different node types, which might be advantageous at small node sizes. However, we found that icons are more easily semantically associated with certain entities than shapes and work well at medium to large node sizes. In most of our images we show the textual labels in black with a colored background corresponding to the color of the node that the label applies to. When visualizing graphs of multiple nodes it is difficult to show all the labels at the same time. In *TopicNets* we address this issue by hiding some labels based on a priority function. If a node label collides with a node of higher priority then its label is faded out. If it collides with more than one node of higher priority then the label is faded even more until it finally disappears if it collides with multiple nodes. When a user zooms into specific regions of the graph the labels re-appear as more space becomes available. Selected nodes have the highest priority and will always have their labels drawn on top of other nodes. Next are the nodes connected to the selected nodes in order to highlight the connectivity of the selected node. Both of these sets of nodes are also highlighted with red text and a drop shadow on the labels. Third on the priority list are the topic nodes, since they are there to summarize the contents of the documents it's more important to show those labels. The fourth thing we consider is node size, since the larger topic nodes are the more frequent topics we want them to be visible whenever possible. Also, the larger document nodes represent bigger documents and thus they could be more important so we think it's more important to show those labels. Finally, if all else is equal we prefer to show shorter labels. This is because they take up less screen space thus there is less chance of them overlapping, resulting in potentially more visible labels. Future work includes a dynamic label deconflicting solution along the lines of [**?**] or [**?**].

4.4 Graph Interactions

Analysts frequently require many perspectives on a data set to perform complex information discovery tasks. *TopicNets* provides a graph interaction mechanism based on [**?**] which allows an analyst to directly manipulate the visualization by clicking on nodes and moving them around the screen. Single or multiple nodes can be selected through search or a ctrl-click and subsequently moved on the screen by click-and-drag mouse movement. Optionally, when node(s) are dragged, an interpolation algorithm is applied to the graph and other nodes transition smoothly in the same direction as the moved node(s), but by a *relative* amount, which is proportional to the graph distance from the moved node(s). Moreover, an effect parameter can be applied through a slider in the interface to specify a maximum graph distance for the interpolation. Using this technique, analysts can mold the graph to highlight interesting features. For example, selecting all documents by a given author and dragging them to one side will deform most graphs into a tree-like layout in which collaborators with the target author are easily identifiable.

4.5 Web-based Architecture

Most visualization software capable of generating highly scalable and interactive graph visualizations are implemented either as desktop applications, or as browser plug-ins which can be resource intensive for a client machine, for example [**?**; **?**; **?**; **?**; **?**; **?**]. *TopicNets* is built on the *WiGis* graph visualization architecture, previously developed by the authors in [**?**], and applied in [**?**; **?**]. The key advantage of this architecture is that it leverages an AJAX-based approach to graph visualization. This allows *TopicNets* to run *natively* in any major web browser, and scale interactively to graphs of hundreds of thousands of connected documents and topics. The system supports interaction by capturing mouse movements in a standard browser and sending them to a remote server which computes and renders a new view of the graph as a bitmap image based on the graph model and the incoming user interaction data. Images are streamed back to the client to provide an interactive experience. For graphs such as those presented here, the system achieves a rate of about 0.1 seconds per frame on a standard network connection. A potential drawback of this architechture is that it can be heavy on the server side resources and thus could be problematic to scale to multiple users. However, we believe that with the power of cloud computing and automatic load balancing this potential drawback can be circumvented.

4.6 Iterative Topic Modeling

An additional benefit of the web-based architecture is that all the "hard work" is being done by a central server. To make full use of this centralized approach we developed a parallel version of the topic modeling algorithm which spreads the work across multiple processors. This results in much faster topic modeling than previously possible without requiring the end user to have a powerful computer, allowing us to support fast iterative topic modeling of subsets of documents. Due to the fact that a topic modeling algorithm takes as input a set of documents and a number specifying how many topics should be produced, users might find themselves with a small set of documents and a much smaller number of topics after performing many filtering tasks on a large dataset. Allowing users to re-run the topic modeling at any point during the analysis session gives them complete control over the input into the topic modeling algorithm, and thus generating topics representative of the set of interest. As demonstrated in Figure 8 c) and discussed in Section 5.2 this iterative technique can help the user learn about sub-topics of a particular topic of interest.

5. USAGE EXAMPLES

We now present three usage scenarios which highlight the information discovery capabilities of the *TopicNets* system across a diverse set of data types. The first two examples show how *TopicNets* can be applied to a collection of documents, while the last demonstrates how the system enables analysing individual large documents. Our examples focus on expertise categorization in large organizations (NSF and CalIT2) and a PhD thesis in computer science. For each use case, graphs were generated by the authors, but interaction and exploration was performed by an individual with expert knowledge of the data set.

5.1 Exploring the CalIT2 Dataset

The California Institute for Telecommunications and Information Technology[3] (CalIT2) is a research institute within the University of California that is jointly administered by UCSD and UC Irvine. The institute encompasses a broad range of research projects and related activities in telecommunications and information technology. A key aspect of the institute

[3] www.calit2.net

is its interdisciplinary nature, bringing together over 400 researchers in computer science, engineering, biology, physics, social science, the arts, and more. A significant challenge facing the institute leadership is gaining a global understanding of "who does what," i.e., understanding the breadth and depth of research expertise spanned by institute members. While the process of manually visiting researchers' Web pages and reading their papers could provide a detailed understanding of what each individual researcher does, this would be a very slow manual process for an institute with several hundred researchers. Furthermore it would not provide a coherent global overview of how these researchers and their research projects are related. This is a common problem in many organizations, not just CalIT2. In particular, as research disciplines change and evolve it becomes increasingly difficult for organizational leaders (such as department chairs, deans, and program managers) to have a global view of researchers and research topics within their organization.

We crawled the Web pages of 464 CalIT2-affiliated researchers and downloaded the text of 10,403 papers that they had authored. These papers contained 24,407,545 word tokens and a unique vocabulary of 43,295 words after removal of standard list of stopwords. We fit a topic model with 150 topics to this data set. In earlier work we demonstrated a text-based Web interface for exploring this topic model[4] [**?**]. The text-based interface links topics and researchers in a sequential fashion, but it does not allow a user to gain an overall network view of how researchers and research topics are related.

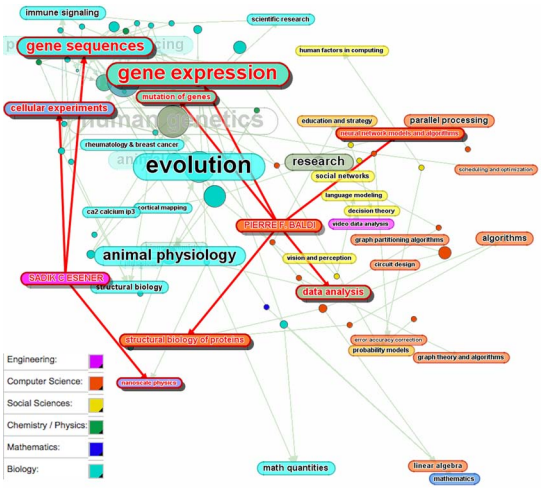


Fig. 7: Example graph of the CalIT2 data after searching for the term "genetics" and collapsing documents into author nodes. The visualization showed us two interesting researchers with relation to genetics, namely a CS professor at UCI and an ECE faculty member at UCSD.

Figure 1 left shows a global view of the CalIT2 topic model as visualized by *TopicNets*. The larger named nodes are learned topics and the smaller nodes are individual documents, where documents are color-coded by the research affiliation of the author. The layout allows a user to quickly gain a global sense of the research activities in CalIT2. Papers from computer scientists, engineers, and mathematicians make up a large fraction of the graph, but there are also considerable contributions from social science, biology, and chemistry and physics. Topics such as "data analysis", "evolution", "numerical methods" are seen to be highly interdisciplinary, while topics such as "immune signaling", "networked computer systems", and "mathematical theory" are less so.

Using the global view of the CalIT2 data as a springboard, users can focus on specific academic areas of interest. For instance, one can use the search utility to find all documents and topics related to "genetics" and one can restrict the visualization to these nodes. Our tool also allows for authors to be visualized in place of their documents. In Figure 7, we show the author-topic network for the "genetics" query, and this visualization allows users to quickly find faculty members at UCI/UCSD who have published on various topics

[4] `http://datalab-1.ics.uci.edu/calit2`

relating to genetics. Figure 7 highlights two such faculty members: Pierre F. Baldi at UCI, and Sadik C. Esener at UCSD. The visualization reveals that Baldi is related to the following topics: "gene expression", "mutation of genes", "structural biology of proteins", "data analysis", and "neural network models and algorithms". This depiction is accurate since Baldi is a computer science professor working on machine learning algorithms such as neural networks, and Baldi is also the director of the Institute for Genomics and Bioinformatics at UCI. Meanwhile, Esener is linked to the following topics: "cellular experiments", "gene sequences", and "nanoscale physics". Esener is an engineering professor working on nanotechnology for various biological applications. This example reveals that our interactive network visualizations can be used to gain valuable information about authors and documents. Since the topics are positioned according to their semantic nature, users can easily find interdisciplinary authors and documents by looking for nodes in-between topics or topical clusters.

5.2 Analyzing NSF Grant Documents

It is important for both researchers and funding agencies to understand the complex and rapidly changing funding landscape. Although research grants often span several research areas, individual grants are usually managed by a single program officer, within a specific program. Topic modeling can provide useful information to go beyond this hard categorization of grants, and can be used to topically characterize or define program officers and programs. To demonstrate this ability, we used a collection of over 50,000 NSF awards from 2006 to 2009[5].

Figure 8 demonstrates how the *TopicNets* system is used to learn meaningful information about different programs and program officers within NSF using the topics learned from the grant documents. The layouts are very informative because similar topics are positioned close to each other. In turn, that explains why the cyan and pink topics in Figure 8 a) and b) are far away from most other nodes in the graph, since these topics have little overlap with the three programs selected in this example. After rerunning the topic modeling algorithm on only the visible set (see Figure 8 c)) we see how some less-related topics are removed and new topics more related to the visible set of documents are generated instead. Finally in Figure 8 d) we have removed all nodes not related to the two program officers Korelsky and Fisher. This image shows that although both these program officers are mostly connected to the same program, the topics of the grants they are in charge of are different. Fisher is in charge of just one grant in "Human-Centered Computing" (green) but many of his "Robust Intelligence" (red) grants are connected to green topics, meaning that these topics are typically found in "Human-Centered Computing" grants.

5.3 Analyzing a PhD Thesis

To provide an example of use with a single document from an authors perspective, a recent PhD thesis by one of the authors of this paper entitled "Trust on the social web, applications in recommender systems and online auctions" was uploaded to *TopicNets*. The entire document contained approximately 20K words over 167 sections. Figure 4 (a.) shows an initial layout of the sections within the document, each positioned along the circumference of a circle. The linear structure of the original document is preserved in the visualization, with the introductory sections shown with lighter yellow shading and fading to a darker mauve color towards the end of the document. Length of a section correlates to node size on the graph. Topic nodes are positioned within this outer circle of document content. These nodes are associated with various section nodes based on a thresholded probability value

[5]Available from `http://www.nsf.gov/awardsearch/`

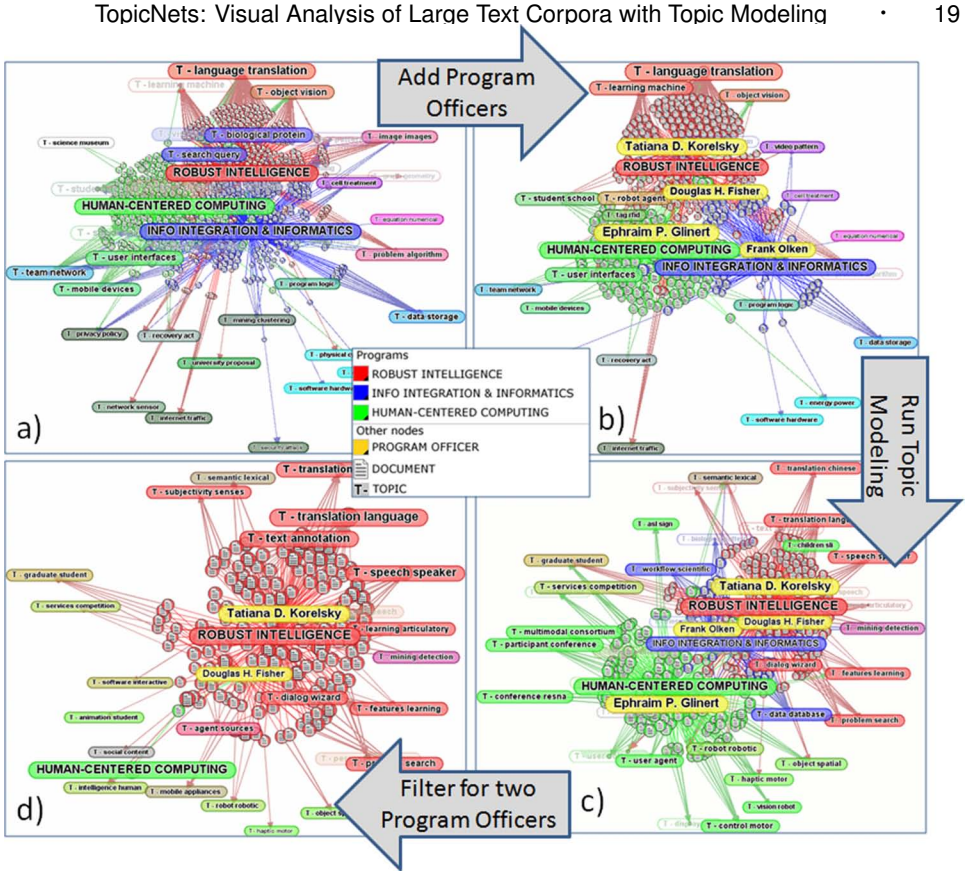
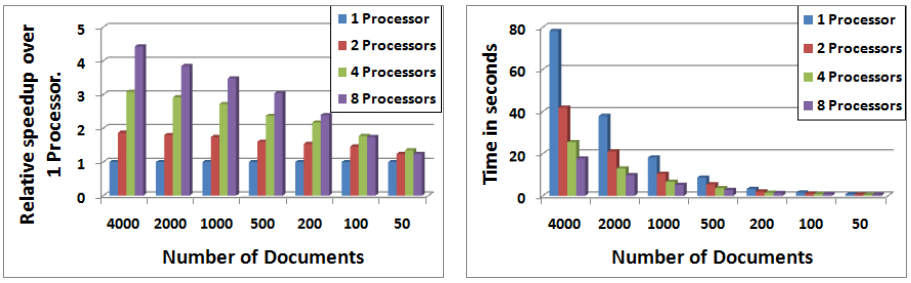


Fig. 8: Visualization of NSF grants and their topics using the topic-similarity layout. Distance between any two topics indicates their similarity, and the topic color is blended from the connected documents. a) shows all the grants related to the three programs "Robust Intelligence", "Human-Centered Computing", and "Info. Integration & Informatics". Grant document nodes are colored depending on which program they belong to, e.g. "Robust Intelligence" documents are red. In b) we added nodes representing the program officers in charge of these grants and focused on four particular program officers, Korelsky, Fisher, Glinert, and Olken. c) shows the same grants, programs, and officers, but after running topic modeling on only the visible set of grant documents. This rerunning of the topic model removes distant topics like "processor memory" while adding topics more specific to these documents, such as "language natural". In d) we have focused further on Fisher and Korelsky who are the two program officers closest to the program "Robust Intelligence". This final panel shows mostly red documents and topics since both officers are primarily in charge of grants under "Robust Intelligence". This view also shows that Fisher has some overlap with "Human-Centered Computing" since many of Fisher's grants are connected to green topic nodes.

that was produced in the topic modeling step. Topic nodes are shown in green and have automatically generated labels, except for a small number of manually added labels, which were simply a reordering of the two highest probability terms in the that topic's word list. Centrality of topic nodes gives a feel for the importance of a topic in the overall document. For example, "trust model" and "reputation network" are clearly important topics, while "graph visualization" and "ebay feedback comments" are of lesser importance because of their peripheral positioning on the graph and smaller sizes. Figure 4 shows snapshots of the author's interactions. Figure 4 (b.) shows a deformation of the sections based on association with a single topic; in this case "trust models". A cursory glance at the outer layer



(a) (b)

Fig. 9: (a) Time in seconds to run topic modeling for 45 topics for varying numbers of processors and corpus sizes. (b) Speed-up of topic modeling for the CVB0 multi processor approach against a traditional single processor implementation.

shows gaps where sections have been drawn inwards by the layout. These gaps represent the distribution of a selected topic across the document. Figure 4 (c.) shows a deformation that is based on associations with all topics in the graph. Section nodes are drawn inwards towards associated topic nodes, again based on a weight controlled by the user via a slide bar. This graph highlights the relevance of individual sections to the central theme of the document. For example, the sections highlighted in red ("Trust in Online Auctions"' and "Attack Models") are not central to the thesis, whereas "trust and reputation in the social web", is a very important section, dealing with the main theme of the work, as verified by the author. In *TopicNets*, transitions between any two views are animated smoothly to provide the user with a good frame of reference from the previous layout.

6. COMPUTATION COSTS

This section evaluates the processing time for each component of *TopicNets*. The system uses the *WiGis* framework, a scalable web-based graph visualization framework presented in [**?**]. That paper describes the scalability of the system in terms of interaction. For example, interacting with a graph of 10,000 nodes takes about 108 ms per frame end to end (9-10 frames per second), which includes running an iteration of the interaction algorithm, rendering the new image, transmitting it across a network and displaying it in the client's browser. This scalability is important for *TopicNets* to be able to provide an interactive experience for the end user. However, *TopicNets* also has some other components which can be computationally expensive. These include running the topic modeling algorithm, generating a graph from the resulting topic model, and finally computing a layout for the resulting graph. We discuss the computational costs for each of these components in this section. It is worth noting that although each of these components are computationally expensive, the results are automatically saved to disk for faster access the next time any user wants to view the same data.

6.1 Topic Modeling

The first part of *TopicNets* is to generate a topic model from the set of documents. Since *TopicNets* does all the expensive computation on a powerful server we were able to use that to our advantage by distributing the topic modeling over multiple processors. We performed a number of experiments to evaluate the speed-up of our multi-processor CVB0 algorithm over a single core topic modeling algorithm. Figure 9 (a) shows the amount of time it takes to compute a topic model for 4000, 2000, 1000, 500, 200, 100, and 50 documents, while Figure 9 (b) shows the relative speed-up over the single processor version

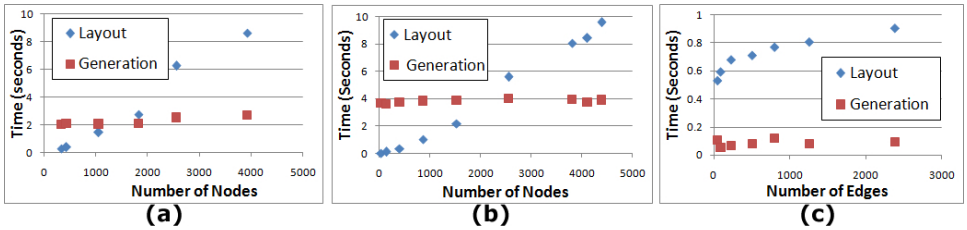


Fig. 10: Time in seconds taken to generate and lay out graphs of varying sizes for three different types of graphs and layouts. These were generated by varying the topic-document association threshold. (a) Generating document-topic graphs and laying them out using the *topic-similarity layout*. (b) Generating document-topic graphs and laying them out using the *order-preserving layout*. (c) Generating section-topic graphs and laying them out using the *section-topic layout*.

for each of the document sets. In all cases we computed 45 topics and used 50 iterations before stopping the topic modeling algorithm. For 4000 documents on one processor it took about 78.3 seconds to complete, while with 8 processors it took about 17.6 seconds. This is a relative speed-up of about 4.4, i.e. it takes 4.4 times longer to compute the model using one processor than with 8 processors. As the document sets get smaller, the relative speed-up is smaller, but the overall time gets smaller. For example, for 100 documents or less it takes less than 1 second to compute the model on 2 processors or more. We can compute a topic model for 4000 documents on 8 processors in about the same time it would take to compute a topic model for 1000 documents on one processor, and 2000 documents can be computed in about the same time as 500 documents on one processor. This speed-up is important for our interactive system, since it enables us to give the user the option to iteratively run topic modeling on a subset of the entire dataset.

6.2 Generating the Graphs

Once the topic model has been computed and stored we need to create a node-link graph of the results as described in Sections 4.1.1 and 4.2.1. Every dataset has a fixed number of documents and topics and to generate the graph the system goes through every document and every topic and determines if it should create an edge between the two, taking into account a user-defined threshold. Documents are then added to the graph if they are connected to at least one topic node. This indicates a time complexity of $O(|D||T|)$ where D is the set of documents and T is the set of topics, meaning that for the same dataset (where D and T remain unchanged) this should result in very similar timings even though the threshold value may be changed. This is confirmed by the red squares in Figure 0**??**.

6.3 Computing the Layouts

After creating a node-link graph we compute a layout to position the nodes as described in Sections 4.1.3 and 4.2.2. Here we present timing data to compute these layouts. Since all our layout techniques use a force directed method, the expected cost of computing a layout is $O(|N|^2 + |E|)$ per iteration, where N is the set of nodes and E is the set of edges. Figure 0**??** (a) shows the time taken to compute the *topic-similarity layout* described in Section 4.1.3.1 for graphs of varying sizes, while Figure 0**??** (b) shows the time taken to compute the *order-preserving layout* described in Section 4.1.3.2. Even though the number of edges is not constant across the different graphs, the cost is mostly driven by the number of nodes in the graph and thus we used number of nodes as our x-axis in these figures. The blue diamonds in Figure 0**??** (a) and (b) show a quadratic growth in layout time relative to number of nodes. When visualizing a single document's sections and their topics, we always keep all the section nodes on the graph so the number of nodes doesn't change much when we change the association threshold. For this reason we plot the number of edges

against the time taken to compute a layout. The plot in Figure 0**??** (c) shows a more or less linear growth in time relative to number of edges, as expected. In summary, the scalability of the system will be constrained by the quadratic time complexity of force-directed layout as the number of nodes increases, although we note that the results are saved to disk for faster access by the next user.

7. CONCLUSION

We have presented *TopicNets*, a system for interactive visual analysis of large document corpora, based on the associations formed by topic modeling. Several techniques for incorporating topic models into the mechanics of graph visualization have been presented, including topic-based deformation of ordered sets of nodes in a graph, collapsing of nodes based on semantic association, single or multiple topic influences, varying of topic association thresholds, interaction with topic and document nodes through interpolation over mouse movements, iterative text-search-and-visualization steps, semantic clustering based on topic similarity, and a range of more generic graph interaction methods. Conversely, interactive graph visualization also informs topic modeling, by enabling users to refine and rebuild ever more detailed topic models while going from an overview of an information landscape into increasingly detailed views of the content and topics of their interest. To highlight the flexibility of the *TopicNets* system and its ability to quickly provide new perspectives on, and insights into large document sets, we presented three diverse use cases. In the CalIT2 example (Figure 1), prominent research areas were highlighted by the *TopicNets* graphs, including interdisciplinary fields such as numerical methods for example. Additionally, faculty members were positioned in topic-space based on their publications, again highlighting areas of expertise and potential collaborators. In the PhD thesis example, it was shown how the system can be used as an editor's or examiner's tool to highlight the relevance of sections to the central theme of the document, and its ability to quickly illustrate esoteric sections of text which differ from the main theme. The NSF example showed how iterative search-and-visualize steps can quickly yield answers to complex, multi-part queries that are inherently difficult to answer in a text based search interface. In summary, the combination of fast topic modeling and graph-based interactive visualization enables powerful novel analysis tools for large text repositories.

8. ACKNOWLEDGEMENTS

This work was partially supported by NSF grant IIS- 0635492 through funds from the IARPA KDD program, by NSF grant CNS-0722075, as well as an ARO MURI award for proposal #56142-CS-MUR. This research was partially sponsored by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on. AA was supported by an NSF graduate fellowship. DN has been supported by NICTA. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

REFERENCES

Touchgraph navigator. Available at www.touchgraph.com.

2009. Stanford Topic Modeling Toolbox. Available at http://nlp.stanford.edu/software/tmt/tmt-0.3/.

2009. Tom Sawyer Visualization. Available at www.tomsawyer.com.

2010. The kitware public wiki. http://www.itk.org/Wiki/ITK_FAQ.

ANALYZER, M. 2010. Matheo analyzer, database analysis and information mapping. http://www.matheo-analyzer.com/.

ANDREWS, K., KIENREICH, W., SABOL, V., BECKER, J., DROSCHL, G., KAPPE, F., GRANITZER, M., AUER, P., AND TOCHTERMANN, K. 2002. The infosky visual explorer: exploiting hierarchical structure and document similarities. *Information Visualization 1*, 166–181.

ASUNCION, A., WELLING, M., SMYTH, P., AND TEH, Y. 2009. On smoothing and inference for topic models. In *Proc. of the 25th UAI*.

ASUNCION, H., ASUNCION, A., AND TAYLOR, R. 2010. Software traceability with topic modeling. In *Proc. of the 32nd ICSE*. ACM.

AUBER, D. 2001. Tulip. In *9th Symp. Graph Drawing*. LNCS, vol. 2265. Springer-Verlag, 335–337.

BATAGELJ, V. AND MRVAR, A. 1998. Pajek - program for large network analysis. *Connections 21*, 47–57.

BELL, B., FEINER, S., AND HÖLLERER, T. 2001. View management for virtual and augmented reality. In *ACM Symposium on User Interface Software and Technology (UIST'01)*. Orlando, FL, USA, 101–110.

BIZER, C., LEHMANN, J., KOBILAROV, G., AUER, S., BECKER, C., CYGANIAK, R., AND HELLMANN, S. 2009. Dbpedia : a crystallization point for the web of data.

BLEI, D. AND LAFFERTY, J. 2006a. Correlated topic models. In *Advances in NIPS 18*. MIT Press, Cambridge, MA, 147–154.

BLEI, D. M. AND LAFFERTY, J. D. 2006b. Dynamic topic‘ models. In *ICML*.

BLEI, D. M., NG, A. Y., AND JORDAN, M. I. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research 3*, 993–1022.

BORG, I. AND GROENEN, P. 2005. *Modern Multidimensional Scaling: Theory and Applications*. Springer.

CALLAHAN, S. P., FREIRE, J., SCHEIDEGGER, C. E., SILVA, C. T., VO, H. T., AND INC, V. 2008. Towards provenance-enabling paraview. In *IPAW*.

CAO, N., SUN, J., LIN, Y.-R., GOTZ, D., LIU, S., AND QU, H. 2010. Facetatlas: Multifaceted visualization for rich text corpora. *IEEE Transactions on Visualization and Computer Graphics 16*, 1172–1181.

CHANG, J. 2009. Reading Tea Leaves: How Humans Interpret Topic Models.

CUTTING, D. R., KARGER, D. R., AND PEDERSEN, J. O. 1993. Constant interaction-time scatter/gather browsing of very large document collections. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '93. ACM, New York, NY, USA, 126–134.

DANIS, C. M., VIEGAS, F. B., WATTENBERG, M., AND KRISS, J. 2008. Your place or mine? Visualization as a community component. In *CHI*. ACM, New York.

DEERWESTER, S., DUMAIS, S. T., FURNAS, G. W., LANDAUER, T. K., AND HARSHMAN, R. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*, 391–407.

DESSUS, P. 2009. An overview of lsa-based systems for supporting learning and teaching. In *Proceeding of the 2009 conference on Artificial Intelligence in Education*. IOS Press, Amsterdam, The Netherlands, The Netherlands, 157–164.

DOGRUSOZ, U., KAKOULIS, K. G., MADDEN, B., AND TOLLIS, I. G. 2007. On labeling in graph visualization. *Information Sciences 177,* 12, 2459 – 2472.

DWYER, T. 2009. Scalable, versatile and simple constrained graph layout. *Comput. Graph. Forum 28,* 3, 991–998.

EADES, P. 1984. A heuristic for graph drawing. *CN 42*, 149–160.

EADES, P. AND HUANG, M. 2000. Navigating clustered graphs using force-directed methods. *J. Graph Alg Appl. 4,* 3, 157–181.

EICK, S. G. AND WILLS, G. J. 1993. Navigating large networks with hierarchies. In *Proceedings of the 4th conference on Visualization '93*. VIS '93. IEEE Computer Society, Washington, DC, USA, 204–209.

FAIRCHILD, K., POLTROCK, S., AND FURNAS, G. 1999. Semnet: Threedimensional representations of large knowledge bases. In *R. Guidon (ed): Cognitive Science and Its Applications for Human-Computer Interaction.* Lawrence Erlbaum Associates, 201–233. Reprint in Card, S.K., et al. (Eds.) Information Visualization: Using Vision to Think. Morgan Kaufmann Publishers, Inc., 1999.

FORTUNA, B., GROBELNIK, M., AND MLADENIC', D. 2005. Visualization of text document corpus. *Informatica*, 497–502.

FREIRE, M., PLAISANT, C., SHNEIDERMAN, B., AND GOLBECK, J. 2010. Manynets: an interface for multiple network analysis and visualization. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*. ACM, New York, NY, USA, 213–222.

FRUCHTERMAN, T. M. J. AND REINGOLD, E. M. 1991. Graph drawing by force-directed placement. *Softw. Pract. Exper. 21,* 11, 1129–1164.

GRETARSSON, B., BOSTANDJIEV, S., O'DONOVAN, J., AND HÖLLERER, T. 2009. Wigis: A scalable framework for web-based interactive graph visualizations. In *GD'09: Proc. of the Intl Symposium on Graph Drawing*.

GRETARSSON, B., O'DONOVAN, J., BOSTANDJIEV, S., HALL, C., AND HÖLLERER, T. 2010. Smallworlds: Visualizing social recommendations. In *Eurographics/IEEE Symp. on Visualization, Bordeaux, France, June 2010*.

GRIFFITHS, T. L. AND STEYVERS, M. 2004. Finding scientific topics. *PNAS 101,* Suppl 1, 5228–5235.

HAVRE, S., HETZLER, E., WHITNEY, P., AND NOWELL, L. 2002. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics 8,* 1, 9–20.

HEARST, M. A. 1995. Tilebars: visualization of term distribution information in full text information access. In *CHI '95: Proc. of the SIGCHI Conf.* ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 59–66.

HEARST, M. A. 1997. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics 23,* 1 (March), 33–64.

HERMAN, MELAN, G., AND MARSHALL, M. S. 2000. Graph visualization and navigation in information visualization. *IEEE TVCG 6,* 1, 24–43.

HOFMANN, T. 2001. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning 42,* 1, 177–196.

INC., Q. 2010. Qpat, intellectual property patent and trademark searching. `http://www.qpat.com/`.

IWATA, T., SAITO, K., UEDA, N., STROMSTEN, S., GRIFFITHS, T. L., AND TENENBAUM, J. B. 2007. Parametric embedding for class visualization.

IWATA, T., YAMADA, T., AND UEDA, N. 2008. Probabilistic latent semantic visualization: topic model for visualizing documents. In *KDD*.

JONES, K. S. 2007. Automatic summarising: The state of the art. *Information Processing and Management 43,* 6, 1449–1481.

KOCH, S., BOSCH, H., AND ERTL, T. 2009. Towards content-oriented patent document processing. *IEEE Symposium on Visual Analytics, Science and Technology*, 203–210.

LACOSTE-JULIEN, S., SHA, F., AND JORDAN, M. I. 2008. Disclda: Discriminative learning for dimensionality reduction and classification.

LAUTHER, U. 2006. Multipole-based force approximation revisited - a simple but fast implementation using a dynamized enclosing-circle-enhanced k-d-tree. In *Graph Drawing*. 20–29.

LIU, S., ZHOU, M. X., PAN, S., QIAN, W., CAI, W., AND LIAN, X. 2009. Interactive, topic-based visual text summarization and analysis. In *CIKM '09: Proceeding of the 18th ACM conference on Information and knowledge management*. ACM, New York, NY, USA, 543–552.

MEI, Q., SHEN, X., AND ZHAI, C. 2007. Automatic labeling of multinomial topic models. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '07. ACM, New York, NY, USA, 490–499.

MILLER, N. E., CHUNG WONG, P., BREWSTER, M., AND FOOTE, H. 1998. Topic islands - a wavelet-based text visualization system. In *Proceedings of the conference on Visualization '98*. VIS '98. IEEE Computer Society Press, Los Alamitos, CA, USA, 189–196.

MISHNE, G., BALOG, K., DE RIJKE, M., AND ERNSTING, B. 2007. Moodviews: Tracking and searching mood-annotated blog posts. In *Proceedings Int. Conf. on Weblogs and Social Media (ICWSM-2007)*. 323–324.

MORVILLE, P. 2005. *Ambient Findability: What We Find Changes Who We Become*. O'Reilly Media, Inc.

NEWMAN, D., ASUNCION, A., SMYTH, P., AND WELLING, M. 2009. Distributed Algorithms for Topic Models. *JMLR 10*, 1801–1828.

NEWMAN, D., BALDWIN, T., CAVEDON, L., HUANG, E., KARIMI, S., MARTINEZ, D., SCHOLER, F., AND ZOBEL, J. 2010. Visualizing search results and document collections using topic maps. *Web Semantics: Science, Services and Agents on the World Wide Web 8,* 2-3 (July), 169–175.

NEWMAN, D., CHEMUDUGUNTA, C., AND SMYTH, P. 2006. Statistical entity-topic models. In *Proc of the 12th ACM SIGKDD*. New York, NY, USA, 680–686.

NEWMAN, D., LAU, J. H., GRIESER, K., AND BALDWIN, T. 2010. Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Los Angeles, California, 100–108.

NEWMAN, D., NOH, Y., TALLEY, E., KARIMI, S., AND BALDWIN, T. 2010. Evaluating topic models for digital libraries. In *JCDL '10: Proceedings of the 10th annual joint conference on Digital libraries*. ACM, New York, NY, USA, 215–224.

O'DONOVAN, J., GRETARSSON, B., BOSTANDJIEV, S., SMYTH, B., AND HÖLLERER, T. 2009. A visual interface for social information filtering. In *Proceedings of IEEE SocialCom*. IEEE.

OLSEN, K. A., KORFHAGE, R. R., SOCHATS, K. M., SPRING, M. B., AND WILLIAMS, J. G. 1993. Visualization of a document collection: the vibe system. *Inf. Process. Manage. 29,* 1, 69–81.

ONG, T.-H., CHEN, H., SUNG, W.-K., AND ZHU, B. 2005. Newsmap: a knowledge map for online news. *Decision Support Systems 39,* 4, 583–597.

SHANNON, P., MARKIEL, A., OZIER, O., BALIGA, N. S., WANG, J. T., RAMAGE, D., AMIN, N., SCHWIKOWSKI, B., AND IDEKER, T. 2003. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res 13,* 11 (November), 2498–2504.

SHEN, H. AND HUANG, J. Z. 2005. Analysis of call centre arrival data using singular value decomposition: Research articles. *Appl. Stoch. Model. Bus. Ind. 21,* 3, 251–263.

SHNEIDERMAN, B. 1996. The eyes have it: A task by data type taxonomy for information visualizations.

SIIRTOLA, H., LAIVO, T., HEIMONEN, T., AND RAIHA, K.-J. 2009. Visual perception of parallel coordinate visualizations. In *IV '09: Proceedings of the 2009 13th International Conference Information Visualisation*. IEEE Computer Society, Washington, DC, USA, 3–9.

SPANGLER, W., KREULEN, J., AND LESSLER, J. 2002. Mindmap: Utilizing multiple taxonomies and visualization to understand a document collection. *Hawaii International Conference on System Sciences 4*, 102.

SPEER, R. H., HAVASI, C., TREADWAY, K. N., AND LIEBERMAN, H. 2010. Finding your way in a multidimensional semantic space with luminoso. In *IUI '10: Proceeding of the 14th international conference on Intelligent user interfaces*. ACM, New York, NY, USA, 385–388.

STASKO, J. T., GÖRG, C., AND LIU, Z. 2008. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization 7,* 2, 118–132.

SWAYNE, D. F., LANG, D. T., BUJA, A., AND COOK, D. 2003. Ggobi: evolving from xgobi into an extensible framework for interactive data visualization. *Comput. Stat. Data Anal. 43,* 4, 423–444.

TRETHEWEY, P. AND HÖLLERER, T. 2009. Interactive manipulation of large graph layouts. Tech report, Dept of Computer Science, UCSB.

VAN HAM, F., WATTENBERG, M., AND VIEGAS, F. 2009. Mapping text with phrase nets. *IEEE Transactions on Visualization and Computer Graphics 15,* 6 (Nov.-Dec.), 1169–1176.

WANNER. 2008. Towards content-oriented patent document processing. *World Patent Information 30,* 1, 21–33.

WISE, J. A., THOMAS, J. J., PENNOCK, K., LANTRIP, D., POTTIER, M., SCHUR, A., AND CROW, V. 1995. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In *IEEE Information Visualization '95*, N. D. Gershon and S. Eick, Eds. IEEE Computer Soc. Press, 51–58.

WONG, P. C., HETZLER, E. G., POSSE, C., WHITING, M. A., HAVRE, S., CRAMER, N., SHAH, A. R., SINGHAL, M., TURNER, A., AND THOMAS, J. 2004. In-spire infovis 2004 contest entry. In *INFOVIS*.