Big Data Research 16 (2019) 49–58

Contents lists available at ScienceDirect

Big Data Research

www.elsevier.com/locate/bdr

Interactive Visual Analytics for Sensemaking with Big Text

Michelle Dowling *, Nathan Wycoff, Brian Mayer, John Wenskovitch, Scotland Leman, Leanna House, Nicholas Polys, Chris North, Peter Hauck

ARTICLE INFO

Article history:
Received 16 September 2018
Received in revised form 22 March 2019
Accepted 17 April 2019
Available online 25 April 2019

Keywords:
Text analytics
Big data
Visualization
Interactive visual analytics
Semantic interaction
Topic modeling

ABSTRACT

Analysts face many steep challenges when performing sensemaking tasks on collections of textual information larger than can be reasonably analyzed without computational assistance. To scale up such sensemaking tasks, new methods are needed to interactively integrate human cognitive sensemaking activity with machine learning. Towards that goal, we offer a human-in-the-loop computational model that mirrors the human sensemaking process, and consists of foraging and synthesis sub-processes. We model the synthesis loop as an interactive spatial projection and the foraging loop as an interactive relevance ranking combined with topic modeling. We combine these two components of the sensemaking process using semantic interaction such that the human's spatial synthesis actions are transformed into automated foraging and synthesis of new relevant information. Ultimately, the model's ability to forage as a result of the analyst's synthesis activities makes interacting with big text data easier and more efficient, thereby facilitating analysts' sensemaking ability. We discuss the interaction design and theory behind our interactive sensemaking model. The model is embodied in a novel visual analytics prototype called Cosmos in which analysts synthesize structure within the larger corpus by directly interacting with a reduced-dimensionality space to express relationships on a subset of data. We then demonstrate how Cosmos supports sensemaking tasks with a realistic scenario that investigates the affect of natural disasters in Adelaide, Australia in September 2016 using a database of over 30,000 news articles.

1. Introduction

The overarching goal of this work is to computationally augment human sensemaking capabilities in the context of big text analysis problems. For example, intelligence analysts must forage large collections of text for relevant information and synthesize a coherent story from fragments. Such sensemaking activities are modeled by Pirolli and Card's "sensemaking loop" [1], which is composed of two primary, interconnected sub-loops: the foraging loop and the synthesis loop. Traditionally, much of this sensemaking activity, especially synthesis, requires human cognitive intelligence. However, to efficiently scale up sensemaking to big data, more semi-automated augmentation is needed. To support the human cognitive activity, it is important that the automation fits naturally into the human sensemaking workflow.

The sensemaking loop is a cognitive model. Thus, to support automation, one challenge is to concretize the sensemaking loop into a computationally-oriented model with formalized sub-components. In this work, we formally model the synthesis loop as an interactive data structuring process, and the foraging loop as an interactive relevance model driven by the result of the structuring model. A related challenge is the high-dimensional nature of text data, which makes it difficult to support real-time, interactive structuring methods. Our approach is to exploit topic modeling methods to reduce dimensionality between the foraging and the synthesis models.

Yet a further challenge in enabling this automation lies in the human-centered, interactive, and iterative nature of sensemaking. For example, in the "dual search" process [1] that connects synthesis and foraging, analysts simultaneously identify hypotheses that synthesize the supporting evidence while also foraging for additional evidence for the hypotheses. Through iteration, analysts *incrementally formalize* [2] their hypotheses and arguments. To support this user-driven nature of the models, we exploit the principles of semantic interaction [3] to steer semi-supervised machine learning algorithms, updating the models based on learned user interest. Semantic interaction methods seek to learn users' cognitive sensemaking intents by observing their interactions, such as their interactive structuring activities in the synthesis loop. This enables analysts to stay focused on their familiar sensemaking process rather than thinking about manipulating underlying statistical models. For our computational sensemaking model, this requires designing machine learning "inverses" [4,5] for the synthesis and foraging models that learn from user's structuring and searching

* Corresponding author.
E-mail address: dowlingm@vt.edu (M. Dowling).

https://doi.org/10.1016/j.bdr.2019.04.003
2214-5796/

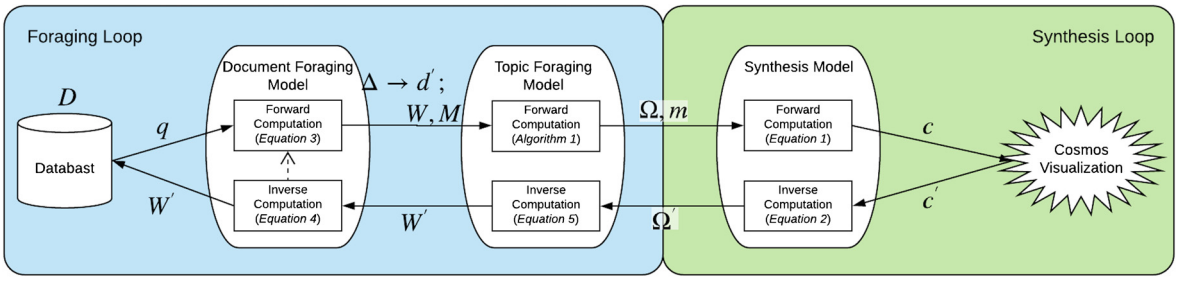

Fig. 1. A computational representation of how the sensemaking loop can be supported for big text analytics, following the conventions for depicting semantic interaction by Dowling et al. [5]. This pipeline is annotated with variables from Table 1 to show the transformation of data throughout the pipeline, including the equations and algorithms we use in Cosmos.

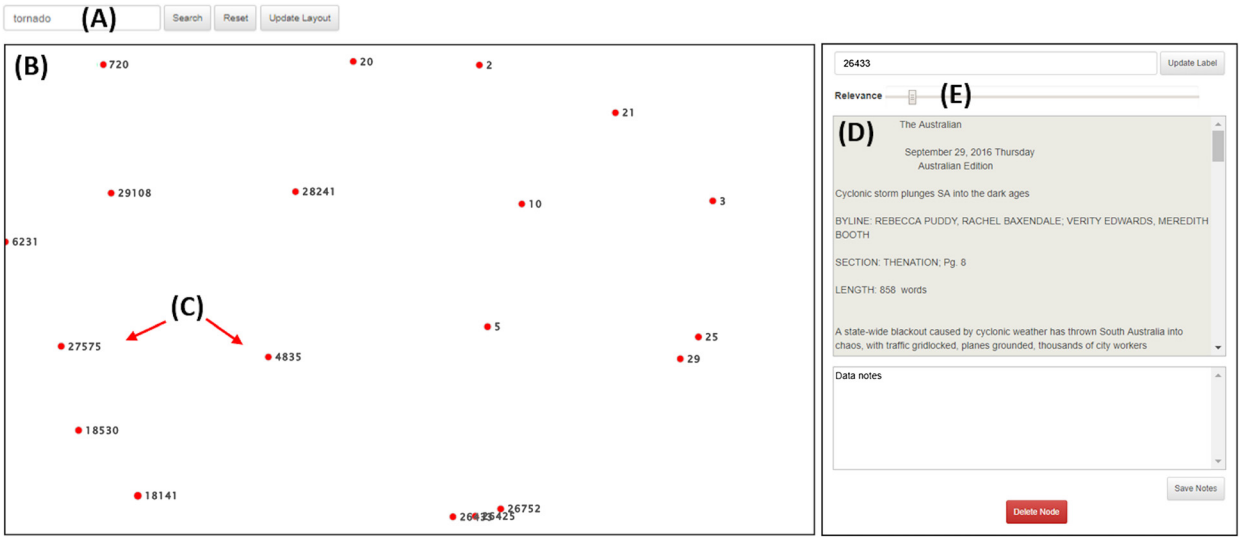

Fig. 2. An overview of the Cosmos system. **(A)** Analysts use keyword search foraging with a text field to begin populating **(B)** the synthesis visualization of the foraged subset of documents. **(C)** Documents within the visualization are projected according to similarity to each other. To the right of this visualization, **(D)** a selected document's text can be read in a scrolling panel. Just above, **(E)** the document's relevance and label can be updated.

actions. To support high-dimensional text data, the topic modeling approach therefore also needs to interactively update based on semantic interactions.

To address these challenges, we designed a sensemaking computational pipeline (summarized by Fig. 1), embodied in a novel visual analytics system for big text called Cosmos (Fig. 2). Specifically, our contributions are:

1. Computational modeling of the sensemaking loop using a semantic interaction pipeline to connect synthesis models to foraging models. The pipeline makes use of a user interest model based on weights on document terms and topics, which are learned via semantic interaction feedback.
2. Modeling the synthesis process as an interactive dimension-reduction spatialization in which users can express similarity relationships in collaboration with the user interest model.
3. Modeling the foraging process in two parts that collaborate with the user interest model:
   (a) a document foraging process that filters documents (acquired from a search engine) based on relevance to the user interest model,
   (b) and a dynamic topic foraging process that reduces dimensionality and updates in the presence of user interaction.

2. Related work

Our approach is designed from a synthesis of the following concepts from the literature and advances previous work in this area [3,5–9].

2.1. Information synthesis

A variety of visual analytics systems incorporate various synthesis models, including network-based synthesis [10,11], entity profile synthesis [12], spatial synthesis [13,14], and interactive clustering [15]. We focus this discussion on spatial synthesis, in which space is used to represent the cognitive model of the analyst. This often takes the form of a "proximity ≈ similarity" visual metaphor, in which similar documents and data points are displayed near each other while dissimilar items are positioned at a distance.

Previous studies have shown that human analysts often make use of physical space to organize and synthesize text data [13,16,17]. Such synthesis techniques have been implemented in a variety of systems. For example, Analyst's Workspace [18] supports a manual approach for spatial synthesis of documents, whereas ForceSPIRE [3], StarSPIRE [6], and BigSPIRE [19] add computational support to assist with the spatial organization. However, these methods were based on heuristics. Building on lessons learned from these existing techniques, systems such as Andromeda [20], SIRIUS [9], and InterAxis [14] use a semi-supervised machine learning approach for spatial synthesis. We leverage a similar approach in Cosmos to enable the interactive positioning of documents within a visual display in a statistically valid and data-supported fashion.

2.2. Information foraging and retrieval

Many foraging models have also been developed for information retrieval. These include techniques such as simple keyword

Table 1
A list of variables used throughout this paper and their descriptions. Variables that appear with a ′ indicate a change or update to that variable.

Variable	Description
D	Full set of documents in the corpus
q	Set of documents returned by a query to D
Δ	Set of relevant documents to add to the visualization; $\Delta \in q$
d	Set of documents to be visualized; $\Delta \cup d = d'$
c	Set of low-dimensional coordinates for each $d_i \in d$
R	A vector representation of relevances for all documents in q
T	All terms in D
t	Topics learned from d
M	A $d \times T$ matrix that describes each $d_i \in d$ in terms of each $T_i \in T$
m	A $d \times t$ matrix that describes each $d_i \in d$ in terms of its probability of belonging to each $t_i \in t$
W	A vector representation of weights on all terms in T
Ω	A vector representation of weights on all topics in t

search foraging, creation of user interest models [6,21], data-based dynamic query expansion [22], query-by-example systems [23], and recommendation systems [24]. The foraging system used by Cosmos falls under the user interest model category. We take a "content-based filtering" approach, in which documents are assigned a score determined by profiles of the item in question and the analyst exploring the collection of all items [25]. Past work has shown that these user models can both broaden queries and help analysts to overcome bias in their foraging process [26].

2.3. Learning through interactive visual feedback

Both the synthesis and foraging processes described in the last two subsections can be learned incrementally through iterative user feedback as part of a human-in-the-loop [27] process. To achieve this incremental learning, semantic interactions [28] can be incorporated in visual analytics systems, emphasizing a contextualized feedback loop between the system and the analyst. While methods implemented in past text analytics systems [3,6] make use of semantic interactions, the learning process implemented in those systems is heuristic. To scale up the benefits of semantic interaction, more rigorous modeling is needed.

Visual analytics frameworks such as V2PI [29] and BaVA [30] offer a potential solution to this modeling challenge. For example, Andromeda [20] enables analysts to interactively steer weighted multidimensional scaling (WMDS) projections of quantitative data. Analysts position a subset of the observations in the space to communicate desired similarity/dissimilarity relationships to the system. Andromeda uses those positions to learn a distance metric that is applied to the full projection. This technique for manipulating projections is referred to as observation-level interaction (OLI) [8,31] and is characterized by the learning step undertaken to generate the distance metric. In other words, the analyst's intent is inferred from their interactions, leading to a learned parameter change in the system. This is in contrast to parametric interaction (PI), in which the analyst directly communicates a desired parameter change to the system [8]. In Cosmos, we adapt these methods to support text data.

3. Sensemaking pipeline for big text

The sensemaking loop described by Pirolli and Card is comprised of two main sub-loops: the foraging loop and the synthesis loop [1]. Thus, we model the sensemaking loop by combining models for foraging and synthesis processes into a computational pipeline represented by Fig. 1. For reference, Table 1 describes frequently used variables throughout this paper.

3.1. Synthesis Model

From the analyst's perspective, the ultimate goal of sensemaking with big text is to synthesize information to formulate and support a hypothesis. This places particular emphasis on the synthesis loop, in which the analyst must explore relationships between documents and determine the relevance of information gathered. This is often accomplished by iteratively examining information, organizing it, and returning to the foraging loop to gather more. This process may also include testing alternative hypotheses, or rejecting or refocusing the hypothesis as additional information is synthesized. Thus, the synthesis loop is accomplished iteratively over time, so a model of the synthesis process must support this continuous exploration and organization of information.

Specifically for text datasets, we propose that there are two main methods for synthesis: leveraging document relevance or using document similarity. Representing similarity spatially for sensemaking has proven to be intuitive and powerful in other visual analytics systems [6,7,14,20,9,27,32–34], including those with big text [19]. These methods for visualizing similarities also enable exploration of different similarity and dissimilarity relationships by manipulating a weight vector. Thus, we propose that the Synthesis Model should be a projection method that takes document-topic matrix m and topic weight vector Ω to produce coordinates c in the visualization. This concept can be represented by the equation $c = Synthesize(m, \Omega)$. Ω thereby forms the first necessary component of a user interest model that represents how interested the analyst is in each topic, with topics defined by another model in the pipeline.

3.2. Foraging models

The synthesis loop becomes difficult for analysts to perform when there is too much information for the analyst to organize manually all at once. This leads analysts to direct their attention to information highly relevant to their investigation first and then forage for additional, related information that is also relevant to the investigation to either support or refute their hypothesis.

The use of relevance to forage for additional documents resulted in modeling one foraging component with a Document Foraging Model. This model filters documents to ensure that only highly relevant documents are displayed, focusing synthesis on just the documents relevant to the investigation. Such filtering is represented by the equation $R = DocRel(q)$. By applying thresholds, relevant documents, Δ, (and therefore visualized documents d') can be determined.

In addition to foraging for specific documents, analysts may wish to forage based on a topic, t_i. Indeed, analysts performing keyword search foraging often relate certain keywords with each other (e.g., synonyms or co-occurrences). Therefore, we argue that keyword search foraging is actually performed based upon topics of interest rather than one specific keyword. Additionally, using term-based representations of documents is problematic due to the sparsity of the data (i.e., the majority of terms do not occur across many documents), leading to documents being represented by a vector consisting primarily of zeros. This sparsity complicates the Synthesis Model, hindering its ability to scale to large datasets. Reducing the data to a "medium"-dimensional space by transforming sets of terms into topics permits the Synthesis Model to run more efficiently, even as the number of documents visualized increases.

To accomplish this translation of terms to topics, a Topic Foraging Model is also necessary. This model is responsible for dynamically detecting topics in the visualized documents based on the equation $m, \Omega = TopicForage(M, W)$. In other words, term weight

vector W forms the second necessary piece of the user interest model by capturing how interested the analyst is in each term.

Performing this topic foraging on only the visualized documents and not the entire dataset keeps the foraged topics closer to the analyst's notion of which topics exist in the dataset. This is because the analyst only knows of the subset that has already been investigated. Another benefit is that which documents to forage can be based on a set of weights on terms rather than just on the single term that is queried. This can produce richer results from the Document Foraging Model for the Topic Foraging Model to utilize, culminating in more relevant topics for the Synthesis Model to visualize. Using topic weights to reflect analyst interest in each topic results in a more accurate reflection of the analyst's notion of document similarity. Thus, the results of these models produce a succinct yet accurate representation of relevant documents for the analyst to synthesize via a similarity-based projection.

3.3. User interest model

To support the sensemaking process, the foraging and synthesis models must *learn* from the analyst's interactions and respond according to their sensemaking activity. This learning means the analyst's interest must be modeled (i.e., a user interest model). Closer inspection of the Foraging and Synthesis Models reveals an interesting feature: all three can be tied together using an interest model that is centered on W and Ω. Alterations to W and Ω can be accomplished by *learning* from user interactions. Following semantic interaction techniques, this learning can be accomplished by pairing each model's computation that helps produce the visualization with an inverse computation. This inverse computation *learns* the analyst's intent and updates the interest model appropriately. While this concept is more thoroughly discussed in [5], we note our use of a semantic interaction pipeline to represent our sensemaking pipeline in Fig. 1.

An example of this inversion is when an analyst drags documents within the projection to redefine their similarities/dissimilarities. This interaction then triggers a learning process to determine a new Ω that describes the analyst-defined layout. Thus, $\Omega' = Synthesize^{-1}(c', m)$. This supports analysts as their notion of similarity changes between investigations or throughout the course of a single investigation while their hypothesis becomes more refined. Analysts could also perform a semantic interaction to assert a desired relevance for a selected document, $M_{i,*}$, as described by $W' = DocRel^{-1}(R'_i, M_{i,*})$ for a user-specified relevance R'_i. These interactions lead to the Document Foraging Model *learning* a new set of term weights that best mirrors the analyst's interest.

These interactions could also be leveraged to perform semantic interaction foraging, an automated foraging technique defined by Wenskovitch et al. [26] that queries for new documents on behalf of the analyst. Which documents are foraged is based on the interest model (specifically the term weights derived from topic weights using $W' = TopicForage^{-1}(m, \Omega'))$.[1]

As a result, these interaction techniques – namely, (1) dragging documents within the projection to express synthesized relationships, or (2) adjusting the relevance rating of documents to express foraging feedback – provided through the synthesis and foraging models provide a natural interface to large-scale text data. By displaying documents highly relevant to the analyst's investigation, we avoid overwhelming the analyst. By using semantic interaction foraging, we *learn* which documents may also be relevant to the analyst and automatically add them into the visualization to further assist in synthesizing information.

[1] Note that M is not part of $TopicForage^{-1}$ since which terms appear in which documents never changes, meaning it is not necessary to include M in this inverse equation.

4. Example prototype

4.1. Design goals

In developing our sensemaking prototype for big text, we note a number of high-level design goals, with design choices relating to each model described in the next subsections. The primary design goal is to provide a simple prototype interface that naturally reflects the analyst's sensemaking process. The emphasis on a simple prototype means that Cosmos is meant to demonstrate how the different models of the pipeline support sensemaking activities as opposed to being a fully-functional system ready for thorough usability evaluation.

A related goal providing an intuitive interface, implying the analyst should not require knowledge of underlying algorithms to interact with the interface. Accomplishing this goal enables the analyst to remain focused on their synthesis processes rather than trying to learn the mathematical underpinnings for this specific system [3,13,27]. The tradeoff in achieving this goal is that details of the mathematical algorithms or user interest model will be hidden; there is no method for analysts to directly access this kind of information, including W and Ω. If the analyst feels that the visualization has missing documents or an inaccurate representation (i.e., an inaccurate W or Ω), then they must use one of Cosmos's interactions to rectify the situation.

Another goal is helping analysts focus on synthesis tasks as this is the part of the sensemaking process that they particularly excel in. To support such synthesis tasks, a similarity-based projection of documents produced by the Synthesis Model dominates the visualization. While a text field is also provided for notes on a single document in further support of goal, analysts may draw various relationships between documents and want to externalize them in a report. Such relationships include content similarity, when they occur relative to each other, coverage of a topic, and others [6, 17]. Rather than attempting to support all possibilities in this final phase of the sensemaking process, we focus on demonstrating how a simple prototype of our pipeline supports sensemaking tasks with big text, allowing analysts to use external mediums (e.g., word processors, hand-written notes, flow charts, etc.) for report writing activities. Specifically, we focus on synthesizing by spatially organizing document nodes in a 2D space. However, our prototype can easily be augmented to support other forms of synthesis for report writing activities, as discussed in Section 6.

4.2. Interface and interactions

In order to accomplish the aforementioned design goals alongside supporting the necessary components implied in the pipeline itself, we developed the web-based visualization depicted in Fig. 2. A similarity-based projection of the documents is the dominating component of the visualization (Fig. 2-B), which reflects the system's focus in supporting the analyst's synthesis processes. Here, we project each document $d_i \in d$ such that more similar documents are projected closer together and dissimilar documents farther apart (Fig. 2-C). In recognition of the fact that analysts may also wish to leverage document relevance, we map the relevance, R_i, of each d_i to the radius of the document's corresponding node in the projection. To the right of this projection, detailed information is provided for a single selected document, including the document label, its calculated relevance, and the document's contents (Fig. 2-D & E). A text field is also provided for the analyst to externalize notes on a selected document. Both the field to view the document's contents and to take notes scroll to fit their contents, allowing these fields to take up a fixed amount of space in the visualization while still scaling to varying amounts of text.

Table 2
A list of additional variables used to describe the algorithms in our prototype implementation and their time complexities.

Variable	Description
N	The total number of documents in D or $\|D\|$
n	The total number of documents in d or $\|d\|$
P	The total number of terms in T or $\|T\|$
k	The total number of topics in t or $\|t\|$

Within Cosmos, a number of interactions are afforded. Following interactions to alter the similarity-based projection, document nodes smoothly transition from one location to another, with new nodes appearing in one corner of the projection and transitioning to their specified location. The simplest interaction is keyword search foraging, which is enabled through a search box above the document projection (Fig. 2-A) and results in a higher weight for that term in W. Then, new topics, t, and their weights, Ω are then learned, causing the projection to update. The corresponding document node can also be clicked, causing the document's label, relevance value, contents, and notes to populate the area to the right of the projection (Fig. 2-D & E).

Three semantic interactions are also afforded, which are enabled by learning new term weights, W, followed by calculating new topic weights, Ω, to update the projection. The first is to delete a node from the projection by clicking the "Delete Node" button below these fields. This decreases the term weight, W_i, for each term in that document. Two semantic interactions trigger automated foraging: OLI and manipulating the relevance slider. OLI always triggers semantic interaction foraging, but foraging after manipulating the relevance slider is conditional. When the relevance slider is increased, this is interpreted as analyst interest in the given document, implying a wish to see additional similar documents. In contrast, decreasing the relevance slider, like deleting a document node, only informs the system of what the analyst is *not* interested in, and does not provide enough information for what the analyst *is* interested in to perform such foraging. How our prototype enables these interactions is described in the remaining subsections, with additional variables described in Table 2.

4.3. The Synthesis Model

To develop our Synthesis Model, we drew inspiration from interactive dimension reduction systems like Andromeda [20], SIRIUS [9], and InterAxis [14] to support synthesis as an interactive process performed within a similarity-based projection of a small set of documents using WMDS [35]. We chose WMDS due to its ability to enable analysts to express their synthesis process through manipulation of document proximities to reflect their perceived similarity. Also, WMDS supports a variety of similarity metrics in both high- and low-dimensional space. Thus, WMDS enables us to fulfill the goals of the Synthesis Model while affording us the flexibility to define the high-dimensional similarity, $dist_H$, as a weighted Euclidean similarity and the low-dimensional-similarity, $dist_L$, as the projected Euclidean similarity. Equation (1) reflects this, where each document is represented as a single row of m, or $m_{i,*}$. Using an iterative implementation results in a time complexity of $O(n^2k)$ per iteration.

$$c = \underset{c_1,\ldots,c_n}{\arg\min} \sum_{i=1}^{n-1} \sum_{j>i}^{n} \left(dist_L(c_i, c_j) - dist_H(\Omega, m_{i,*}, m_{j,*})\right)^2 \quad (1)$$

We enable analysts to change these topic weights through OLI to denote the perceived similarity/dissimilarity between them. The Synthesis Model can then *learn* new topic weights (i.e., amount of interest) using the following equation:

$$\Omega' = \underset{\Omega'_1,\ldots,\Omega'_k}{\arg\min} \sum_{i=1}^{n-1} \sum_{j>i}^{n} \left(dist_L(c'_i, c'_j) - dist_H(\Omega', m_{i,*}, m_{j,*})\right)^2 \quad (2)$$

Note that this equation effectively inverts Equation (1). This inversion is achieved via gradient-based iterative minimization. As the objective's derivative takes $O(n^2k)$ operations to evaluate, and must be evaluated for each of k variables to optimize, the time complexity of this algorithm is $O(n^2k^2)$.

After learning new term weights from these topic weights the term weights can be leveraged to perform semantic interaction foraging [26], thereby automatically revealing additional relevant information after OLI. Further details on how we accomplish semantic interaction foraging in our prototype are provided in Section 4.4 and Section 4.5, with an example provided in Section 5.2.

4.4. The Document Foraging Model

In addition to traditional keyword search foraging, our Document Foraging Model enables semantic interaction foraging to automatically bring additional, relevant documents into the visualization after semantic interactions like OLI. To create this model, we drew inspiration from StarSPIRE [6], which uses a simple yet effective calculation for document relevances combined with thresholds to determine which of the foraged documents to display in the projection [26].

In our implementation, all foraging is accomplished through queries to an Elasticsearch database [36]. Each query to the database has a time complexity of $O(1)$ since the documents within the database are indexed and hashed. After receiving the query results, the model then determines the top 10 documents that are above a fixed relevance value.[2] This ensures that only highly-relevant documents are displayed while also guaranteeing that the analyst will not be overwhelmed by too many documents appearing at once. Thus, the analyst is provided with a simple yet effective interface to large scales of text data.

Our relevance computation represents each document as a vector of TF-IDF values, which effectively represents the document data as a Bag of Words or Vector Space Model [37]. These TF-IDF values combined with term weights leads to the following equation to compute the relevance of a single document, represented as a single row of document-topic matrix M or $M_{i,*}$:

$$R_i = M^T_{i,*} W. \quad (3)$$

Initially, these term weights are set to 1, and they update to reflect their level of importance to the analyst's investigation. Since this process is repeated for each document, the time complexity of this computation is $O(nP)$.

To support the semantic interaction of manipulating the relevance slider, we must determine how to calculate the specified relevance value, R'_i for a given document. This necessitates an inverse computation to determine new term weights to produce R'_i. The following equation performs this computation, where $M_{i,*}$ represents a single row of document-topic matrix M:

$$W' = W + M_{i,*} \frac{(R_i - R'_i)}{M^T_{i,*} M_{i,*}} \quad (4)$$

This equation rescales term weight vector W by another vector proportional to the document's TF-IDF values, $M_{i,*}$, whose relevance is being changed from R_i to R'_i. The time complexity of our implementation of $DocRel^{-1}$ is $O(P)$.

[2] These thresholds were based on our particular use case (described in Section 5) For other use cases, altering the thresholds may forage greater/fewer new documents at each iteration.

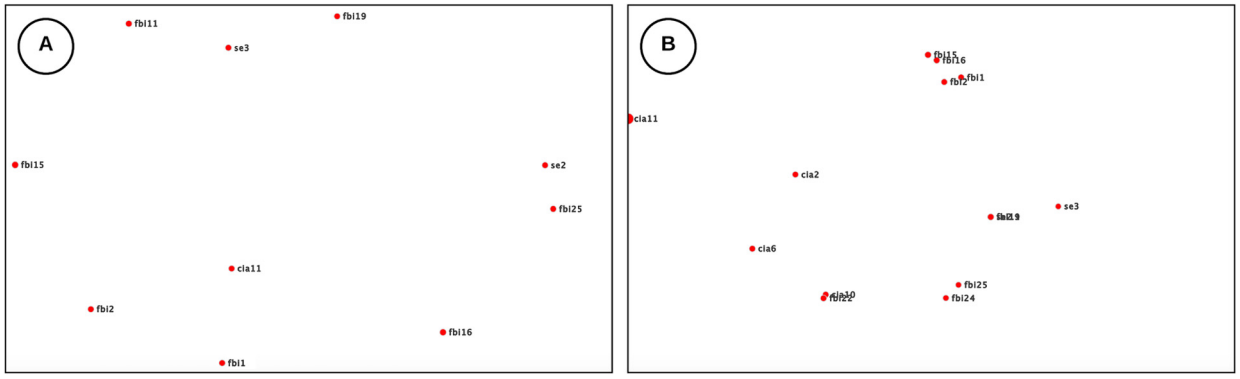

Fig. 3. A depiction of how the interaction with the relevance slider in Cosmos works. After a query for a person's name, **(A)** 5 documents appear in Cosmos. One of these documents (cia11) seems particularly relevant to the analyst's investigation. After increasing the relevance of this document using the slider, **(B)** new documents related to the document the analyst interacted with appear using semantic interaction foraging. The node sizes and positions in document projection are also updated based on this interaction.

With the new term weights, automated foraging is performed, followed by recalculating the relevance for all displayed documents using Equation (3). After learning new topics via Algorithm 1, the document projection updates via Equation (1). Thus, manipulation of the relevance slider triggers semantic interaction foraging enabled by the Document Foraging Model *learning* which terms are most important to the analyst's investigation. An example of this interaction is depicted in Fig. 3. Semantic interaction foraging after OLI is similarly handled by the Document Foraging Model once the Topic Foraging Model translates topic weights to term weights, as described in the next subsection.

Algorithm 1 Generative Model for LDA.

for $i = 1 : k$ **do**
  $t_i \sim \text{Dirichlet}(\eta)$
for $i = 1 : n$ **do**
  $m_{i,*} \sim \text{Dirichlet}(\alpha)$
  for $j = 1 : N_i$ **do**
    $z_{i,j} \sim \text{Multinomial}(m_{i,*})$
    $M_{i,j} \sim \text{Multinomial}(t_{z_{i,j}})$

4.5. The Topic Foraging Model

To transform a term-based representation of documents into a topic-based representation usable to the Synthesis Model, our Topic Foraging Model produces a vector of probabilities that mirror the prevalence of each topic in each document. This requires learning which terms belong to which topics, expressed as a probability distribution across terms. Given the terms that appear in each document, inferences can be made on the topic probabilities of that document.

We generate this topic-based representation of documents using Latent Dirichlet Allocation (LDA) [38]. We chose LDA because it enables us to represent the prevalence of topics within each document, which provides more fine-grained information to the Synthesis Model. Specifically, we use a weighted modification of the uncollapsed variational algorithm presented by Blei et al. [38] which has a time complexity of $O(nkP)$ per iteration. We fix the number of topics, k, to 5 for the purposes of our prototype.[3] The generative model for the topics and document contents is shown in Algorithm 1. Each topic, represented as a simplex valued variable, is denoted as t_i. We denote the vector containing document lengths as N, and $z_{i,j}$ a latent variable indicating which topic the j'th word of document i references. The proportion of each topic in document d_i, also simplex valued, is denoted by $m_{i,*}$. t_i and $m_{i,*}$ are each endowed with Dirichlet prior distributions with exchangeable concentration using parameters η and α respectively, which represent the number of times each word or topic reference was observed *a priori*. Once estimated, the $m_{i,*}$ for each document is passed to the Synthesis Model to use in projecting the documents in Cosmos.

These topics are also given weights to use in the Synthesis Model to denote the analyst's levels of interest across the different topics. Initially, these weights are set to 1. However, our foraging technique necessitates weights on terms rather than on topics. In such cases where new term weights, W', must be learned from topic weights, Ω, we use the following formula:

$$W'_i = t'_i \Omega \tag{5}$$

Repeating this equation for each W'_i transfers importance to words which have a higher probability in topics with a time complexity of $O(Pk)$.

5. Use case scenario

This example demonstrates using Cosmos in a realistic scenario to accomplish a sensemaking task, showing how the synthesis and foraging models interact jointly to accomplish an analytical goal. The scenario centers on a dataset of over 30,000 documents, primarily consisting of news articles of varying lengths. These documents were obtained from LexisNexis [40] using a keyword search on "adelaide" to create the foundation of a scenario focused on the worst series of storms that hit South Australia in 50 years [41]. These storms occurred on 28 September 2016. Additionally, 30 hand-picked articles specifically related to the storms were added to the dataset and assigned IDs less than 30.

In response to reports of tornadoes and severe weather, the United States plans to send humanitarian assistance to Adelaide, Australia [42]. A hypothetical analyst must assess the impact of these storms from the dataset and determine the level of support needed. This example analysis took 75 minutes to complete, including reading 20 documents that averaged 700 words. Given the average response time for interactions was 1–2 seconds, with the longest interaction taking 4 seconds (still well within interactive rates), this means that the analyst's time was focused on reading and synthesizing the information within the documents. Thus, through using the interactive similarity-based projection of documents in Cosmos, the analyst was able to successfully focus on their synthesis-related tasks to investigate the given scenario

[3] In general, choosing the number of topics is a challenging task and an open research problem (e.g., [39]). Given research suggesting people have difficulty thinking in more than 2–3 dimensions simultaneously [4], we chose a smaller k to mirror the analyst's perception.

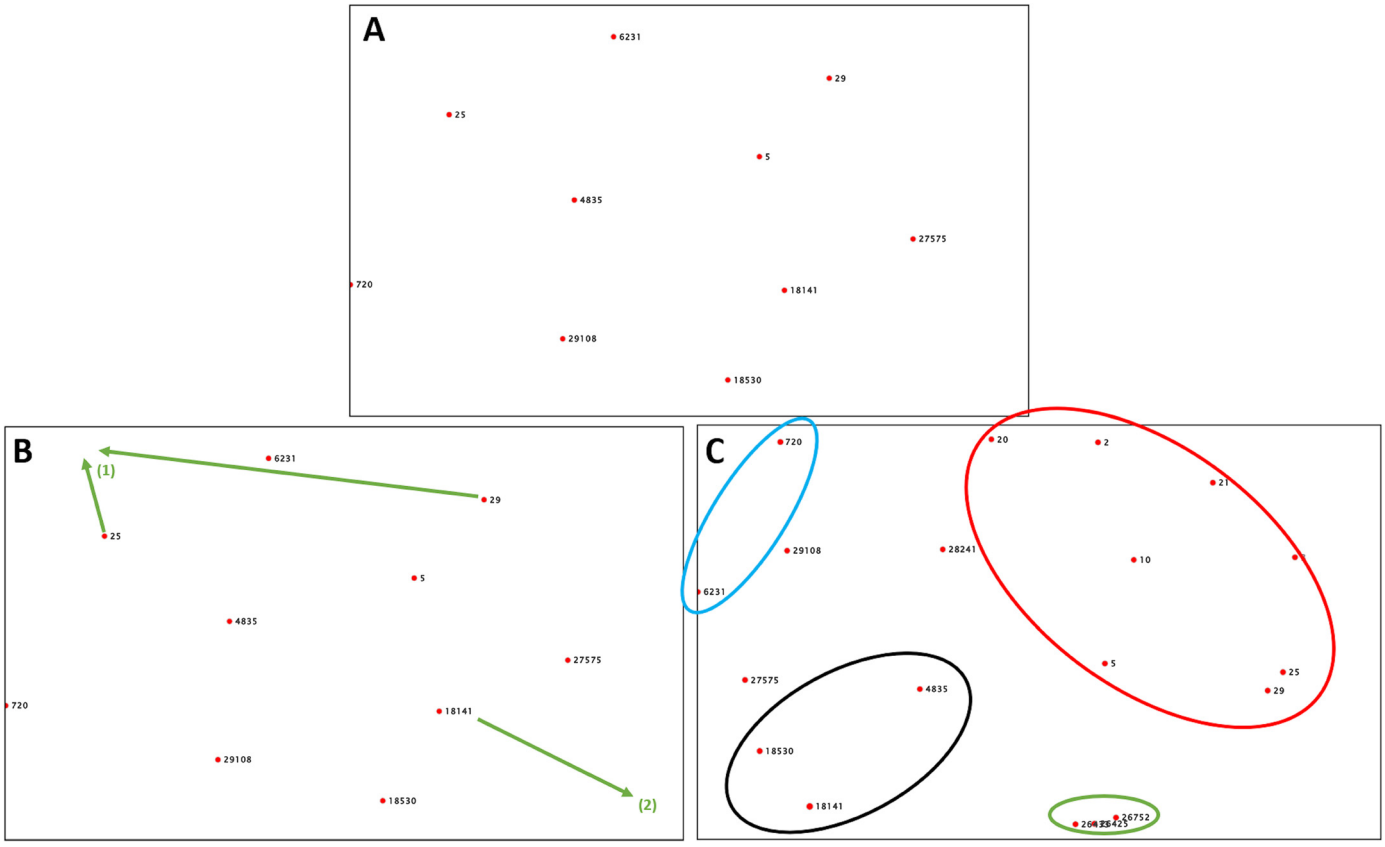

Fig. 4. After searching "tornado," Cosmos **(A)** visualizes foraged documents. The analyst **(B)** uses OLI to express perceived similarities/dissimilarities between documents, resulting in **(C)** an updated projection, including new documents from semantic interaction foraging. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

without needing to learn or directly interact with the underlying algorithms. However, it is important to note that a different analyst may take a different approach in the investigation, leading to a different set of initial insights regarding the impact of the storms.

5.1. Initiating the investigation

To begin the investigation, the analyst queries documents using the search term "tornado". Ten foraged documents then populate the document projection (Fig. 4-A). Three of the documents (181411, 18530, and 4835) reference previous storms. One document (720) is about a sports team called the "Tornadoes." Another document (6231) references a person named "Adelaide" and is clearly unrelated to the storms. Two other documents (29108 and 27575) mention the storm but do not focus on it. The final three documents (29, 25, and 5) are all planted documents related directly to the storm.

5.2. User-driven synthesis modeling using OLI

The analyst begins to form an initial mental model about the storms, focusing on how some discuss previous storms and others describe the recent storms of interest. To clarify this distinction, the analyst uses OLI to express these perceived similarities/dissimilarities between the documents. This is accomplished by moving the document nodes related to the recent storm (25 and 29) together in one corner (Fig. 4-B-1) and a document node referencing previous storms (1814) in an opposing corner (Fig. 4-B-2). The analyst then clicks "Update Layout," which triggers the Synthesis Model to *learn* which topics in the dataset best reflect these similarities/dissimilarities. After the Topic Foraging Model *learns* new terms weights, the Document Foraging Model uses these term weights to automatically forage for new documents to add to the visualization.

Note that through the analyst's interpretation of the visualization and the subsequent OLI, the analyst was able to focus on their synthesis task and also express the results of their synthesis to the system. In response, semantic interaction foraging occurred, meaning that the analyst did not have to switch their cognitive focus to foraging tasks to continue synthesizing information. Additionally, the analyst did not require any knowledge of the underlying algorithms in order to perform this interaction; the analyst only needed to understand the "proximity $\approx$ similarity" visual metaphor of the similarity-based projection.

5.3. Exploring foraged and synthesized documents

The resulting document projection (Fig. 4-C) includes nine new documents from the semantic interaction foraging triggered by OLI. Documents related to the storm are in the red (rightmost) group. The unrelated documents in the blue group are farthest from the red group. The historical storm documents in the black group are slightly closer but still separate. All newly foraged documents were related to the storm due to the Document Foraging Model's ability to identify their relevance from the OLI made by the analyst. The visual structure created by the Similarity Model reinforces the analyst's mental model about how these documents relate to each other and helps in quickly sorting through the newly added documents (which were placed in the red and green groups).

The analyst now explores the new documents in the important red group and the nearby (and therefore similar) green group. The analyst reads one of the newly foraged document nodes (26433) from the green group and finds out that the storms have caused statewide power outages (Fig. 5-Left). Based on this, the analyst

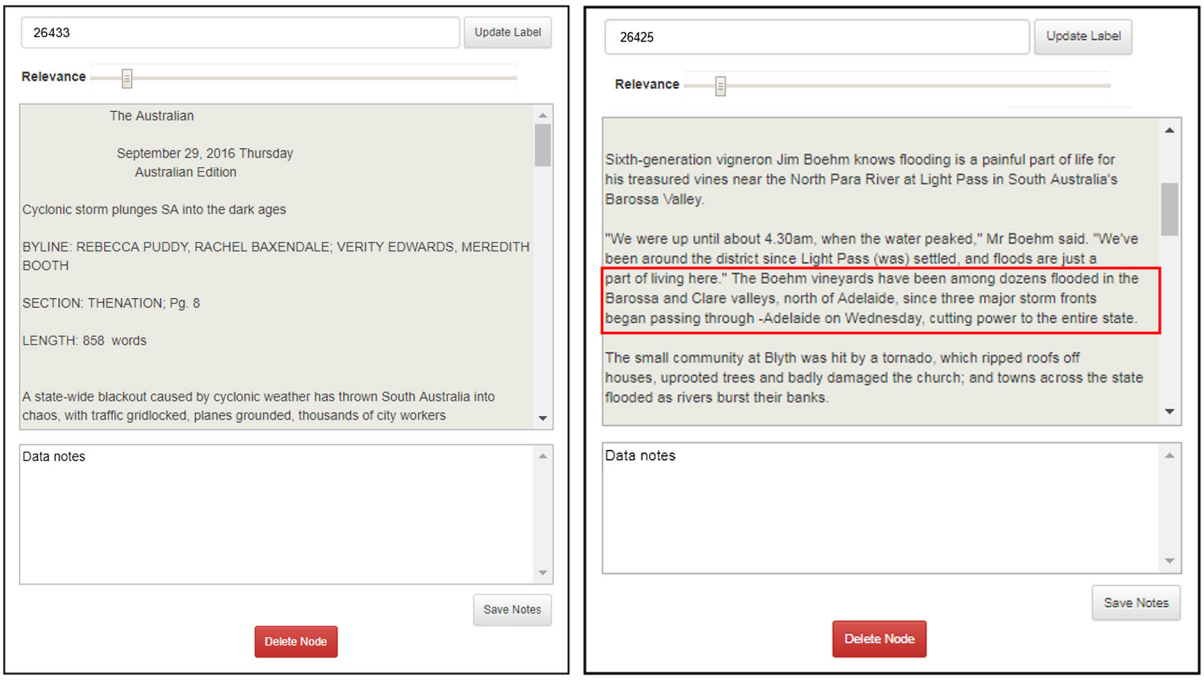

Fig. 5. The contents of foraged documents in Cosmos that reveal how the storms have impacted Adelaide and surrounding areas.

determines that disaster relief will need to consider massive power outages in addition to assistance for the physical damage that the storm has caused. Since this document does not directly reference tornadoes, this insight was possible due to semantic interaction foraging triggered by OLI.

Following this insight, investigating another document in the same group (26425) reveals areas north of Adelaide also without power and experiencing extensive flooding (Fig. 5-Right). Thus, relief efforts must also consider flood-related issues (e.g., people trapped in houses). Note this document is not directly focused on the storm and would likely not be found by reading storm reports. Continuing to read and interact with documents to gain additional insights can lead semantic interaction foraging to quickly uncover other relevant information.

6. Discussion

Cosmos provides a foundation for future research in sensemaking for big text. As shown in Section 5, our prototype enables analysts to investigate large sets of documents and begin to quickly draw conclusions from them. Our multi-model approach to forage for documents and allow analysts to synthesize this information is at the core of our approach. We next discuss some limitations of the Cosmos system and describe future work to resolve the issues we uncovered.

Firstly, we recognize that the concepts we chose to model (as represented in Fig. 1) may not be optimal. For example, there may be other concepts that better reflect an analyst's notion of synthesis than similarity-based methods. Alternatively, perhaps more models are needed to properly capture the complexity of the foraging and synthesis processes. These types of alternatives warrant further investigation (perhaps through comparative user studies) regarding the tradeoffs in different computational models and visualizations and whether they accurately embody components of the sensemaking loop.

We also note several limitations to our Cosmos prototype. Beginning with the visualizations, the time complexities of WMDS (Equation (1)) and its inverse (Equation (2)) limit the number of documents that can be efficiently projected into the display and interacted upon. We are already researching optimizations of both of these equations to permit visualization and interaction on even larger sets of documents. Similarly, we recognize that altering the relevance threshold would impact the documents added to the visualization. We plan to reformulate the Document Foraging Model to enable variable relevance thresholds, allowing the analyst to control how many documents should be foraged and the density of the projection. This can be accomplished via direct user input (e.g., slider bars) or by developing new semantic interactions that *learn* these parameters.

Additionally, our implementations of each model were based on our previous research [5,6,9,20,26] as well as algorithm commonality, simplicity, and flexibility. We acknowledge that alternative methods for implementing the same model components exist (e.g., using weighted or otherwise interactive variants of t-SNE [33], PCA [43], LAMP [34], or other similarity-based methods in place of WMDS; Rocchio [44] or PageRank [45] can replace our relevance calculations; and LSA [46] or clustering algorithms could be leveraged as alternatives to LDA). Each alternative method implies different tradeoffs in the visualization and/or interactions. For example, we chose WMDS because it provides a "proximity $\approx$ similarity" visual metaphor, which matches how analysts naturally organize information [17]. However, the resulting projection is nondeterministic, meaning changes in the projected pairwise distances may not accurately reflect changes in the user interest model, and the projection can rotate between states. Thus, while these changes are not meaningful, the analyst may mistake them to be. These issues can be solved by further improving our current WMDS implementation or experimenting with other projection methods.

We can address these limitations regarding how we implemented Cosmos by performing user studies using an iteration of Cosmos as described in this paper against alternative versions. These other versions would implement existing concepts in different manners (e.g., using other similarity-based projections), perhaps drawing inspiration from other systems that support synthesis or foraging models (e.g., TIARA [47]). Such user studies would be immensely informative for future visual analytics systems for big text by helping researchers understand the tradeoffs implied by different modeling and implementation methodologies.

So far, we have focused on supporting spatial structuring forms of synthesis. An interesting future opportunity is to explore another form of synthesis that is incorporated in the final step of the sensemaking loop: report writing. In this final step, the analyst must synthesize all relevant text into a narrative. Such a narrative may involve a number of relationships between documents, as indicated in Section 4.1. Thus, to support externalizing such a narrative, Cosmos should be extended to incorporate additional visual components or interactions.

Other extensions to Cosmos include enabling analysts to see and interact with the top terms in the top topics or the top terms among the visualized documents. Such interactions may enable manipulation of the term or topic weights from the Topic Foraging Model more directly. Alternatively, the document projection can be augmented with information such as uncertainty in the projection resulting from Equation (1). This may help mitigate the aforementioned issue of analysts misunderstanding changes to the projection. Interaction in such a projection can allow the analyst to express uncertainty, providing additional feedback for underlying algorithms to learn from. Cosmos could also be augmented with geographical component of the user interest model, which the analyst can interact with via a map visualization. Finally, Cosmos might also be extended with a collaborative mode, allowing multiple analysts to interact with the same data. Such collaborations may help analysts reach conclusions faster, result in higher confidence in the results, or assist in disseminating the information learned.

7. Conclusion

This paper introduced a new computational model of the human sensemaking process to enable systems that support interactive big text analytics. This model takes the form of a pipeline, which is comprised of a series of smaller computational models (namely, a Document Foraging Model, Topic Foraging Model, and Synthesis Model) that mirror the foraging and synthesis loops within the sensemaking loop [1]. By leveraging a user interest model, these computational models are connected and interactive, allowing the analyst to iteratively investigate the dataset and dynamically refine their investigation. We demonstrated a prototyped implementation of these models through Cosmos, a new visual analytics system for big text data. We described the mathematics and functionality of each implemented model in detail, and demonstrated how they support the exploration of a 30,000 document collection in a realistic use case.

Acknowledgements

Many developers, researchers, and domain experts played a role in the development and analysis of this system. The authors wish to recognize the contributions made by Ian Crandell, Mai Dahshan, Sajal Dash, Wu-chun Feng, J.T. Fry, Joseph Grube, Zack Hauck, Lata Kodali, Theo Long, Sierra Merkes, Nate Miller, Srijith Rajamohan, Dasha Savina, Matt Slifko, Kenneth Worden, and Bright Zheng. This work was funded in part by NSF grant IIS-1447416 and grant DGE-1545362 as well as by General Dynamics Mission Systems.

References

[1] P. Pirolli, S. Card, The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis, in: Proceedings of International Conference on Intelligence Analysis, vol. 5, 2005, pp. 2–4.

[2] F.M. Shipman, C.C. Marshall, Formality considered harmful: experiences, emerging themes, and directions on the use of formal representations in interactive systems, Comput. Support. Coop. Work 8 (4) (1999) 333–352, https://doi.org/10.1023/A:1008716330212.

[3] A. Endert, P. Fiaux, C. North, Semantic interaction for visual text analytics, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, ACM, New York, NY, USA, 2012, pp. 473–482.

[4] J.Z. Self, M. Dowling, J. Wenskovitch, I. Crandell, M. Wang, L. House, S. Leman, C. North, Observation-level and parametric interaction for high-dimensional data analysis, ACM Trans. Interact. Intell. Syst. 8 (2) (2018) 15, https://doi.org/10.1145/3158230 (36 pp.).

[5] M. Dowling, J. Wenskovitch, P. Hauck, A. Binford, N. Polys, C. North, A bidirectional pipeline for semantic interaction, in: IEEE VIS 2018 Workshop on Machine Learning from User Interaction for Visualization and Analytics, 2018.

[6] L. Bradel, C. North, L. House, S. Leman, Multi-model semantic interaction for text analytics, in: 2014 IEEE Conference on Visual Analytics Science and Technology, VAST, 2014, pp. 163–172.

[7] J. Wenskovitch, C. North, Observation-level interaction with clustering and dimension reduction algorithms, in: Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics, HILDA'17, ACM, New York, NY, USA, 2017, pp. 14:1–14:6.

[8] J.Z. Self, X. Hu, L. House, S. Leman, C. North, Designing usable interactive visual analytics tools for dimension reduction, in: CHI 2016 Workshop on Human-Centered Machine Learning, HCML, 2016.

[9] M. Dowling, J. Wenskovitch, J.T. Fry, S. Leman, L. House, C. North, Sirius: dual, symmetric, interactive dimension reductions, IEEE Trans. Vis. Comput. Graph. 25 (1) (2019) 172–182, https://doi.org/10.1109/TVCG.2018.2865047.

[10] R.H. Mathams, The Intelligence Analyst's Notebook, no. 151, Research School of Pacific Studies, Australian National University, Strategic and Defence Studies Centre, 1988.

[11] J. Stasko, C. Görg, Z. Liu, Jigsaw: supporting investigative analysis through interactive visualization, Inf. Vis. 7 (2) (2008) 118–132, https://doi.org/10.1057/palgrave.ivs.9500180.

[12] E.A. Bier, E.W. Ishak, E. Chi, Entity workspace: an evidence file that aids memory, inference, and reading, in: S. Mehrotra, D.D. Zeng, H. Chen, B. Thuraisingham, F.-Y. Wang (Eds.), Intelligence and Security Informatics, Springer, Berlin, Heidelberg, 2006, pp. 466–472.

[13] A. Endert, P. Fiaux, C. North, Semantic interaction for sensemaking: inferring analytical reasoning for model steering, IEEE Trans. Vis. Comput. Graph. 18 (12) (2012) 2879–2888, https://doi.org/10.1109/TVCG.2012.260.

[14] H. Kim, J. Choo, H. Park, A. Endert, Interaxis: steering scatterplot axes via observation-level interaction, IEEE Trans. Vis. Comput. Graph. 22 (1) (2016) 131–140, https://doi.org/10.1109/TVCG.2015.2467615.

[15] D.M. Russell, M. Slaney, Y. Qu, M. Houston, Being literate with large document collections: observational studies and cost structure tradeoffs, in: Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS'06, vol. 3, 2006, p. 55.

[16] A.C. Robinson, Collaborative synthesis of visual analytic results, in: 2008 IEEE Symposium on Visual Analytics Science and Technology, 2008, pp. 67–74.

[17] C. Andrews, A. Endert, C. North, Space to think: large high-resolution displays for sensemaking, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10, ACM, New York, NY, USA, 2010, pp. 55–64.

[18] C. Andrews, C. North, Analyst's workspace: an embodied sensemaking environment for large, high-resolution displays, in: 2012 IEEE Conference on Visual Analytics Science and Technology, VAST, 2012, pp. 123–131.

[19] L. Bradel, N. Wycoff, L. House, C. North, Big text visual analytics in sensemaking, in: 2015 Big Data Visual Analytics, BDVA, 2015, pp. 1–8.

[20] J.Z. Self, R.K. Vinayagam, J.T. Fry, C. North, Bridging the gap between user intention and model parameters for human-in-the-loop data analytics, in: Proceedings of the Workshop on Human-In-the-Loop Data Analytics, HILDA '16, ACM, New York, NY, USA, 2016, pp. 3:1–3:6.

[21] T. Ruotsalo, J. Peltonen, M. Eugster, D. Glowacka, K. Konyushkova, K. Athukorala, I. Kosunen, A. Reijonen, P. Myllymäki, G. Jacucci, S. Kaski, Directing exploratory search with interactive intent modeling, in: Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management, CIKM '13, ACM, New York, NY, USA, 2013, pp. 1759–1764.

[22] N. Ramakrishnan, P. Butler, S. Muthiah, N. Self, R. Khandpur, P. Saraf, W. Wang, J. Cadena, A. Vullikanti, G. Korkmaz, C. Kuhlman, A. Marathe, L. Zhao, T. Hua, F. Chen, C.T. Lu, B. Huang, A. Srinivasan, K. Trinh, L. Getoor, G. Katz, A. Doyle, C. Ackermann, I. Zavorin, J. Ford, K. Summers, Y. Fayed, J. Arredondo, D. Gupta, D. Mares, 'beating the news' with embers: forecasting civil unrest using open source indicators, in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, ACM, New York, NY, USA, 2014, pp. 1799–1808.

[23] M.Q.W. Baldonado, T. Winograd, Sensemaker: an information-exploration interface supporting the contextual evolution of a user's interests, in: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '97, ACM, New York, NY, USA, 1997, pp. 11–18.

[24] F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, in: Recommender Systems Handbook, Springer, 2011, pp. 1–35.

[25] P. Brusilovski, A. Kobsa, W. Nejdl, The Adaptive Web: Methods and Strategies of Web Personalization, Springer Science & Business Media, 2007.

[26] J. Wenskovitch, L. Bradel, M. Dowling, L. House, C. North, The effect of semantic interaction on foraging in text analysis, in: 2018 IEEE Conference on Visual Analytics Science and Technology, VAST, 2018.

[27] A. Endert, M.S. Hossain, N. Ramakrishnan, C. North, P. Fiaux, C. Andrews, The human is the loop: new directions for visual analytics, J. Intell. Inf. Syst. 43 (3) (2014) 411–435.
[28] A. Endert, Semantic interaction for visual analytics: toward coupling cognition and computation, IEEE Comput. Graph. Appl. 34 (4) (2014) 8–15, https://doi.org/10.1109/MCG.2014.73.
[29] S.C. Leman, L. House, D. Maiti, A. Endert, C. North, Visual to parametric interaction (v2pi), PLoS ONE 8 (3) (2013) 1–12, https://doi.org/10.1371/journal.pone.0050474.
[30] L. House, S. Leman, C. Han, Bayesian visual analytics: BaVA, Stat. Anal. Data Min. 8 (1) (2015) 1–13, https://doi.org/10.1002/sam.11253.
[31] A. Endert, C. Han, D. Maiti, L. House, S. Leman, C. North, Observation-level interaction with statistical models for visual analytics, in: 2011 IEEE Conference on Visual Analytics Science and Technology, VAST, 2011, pp. 121–130.
[32] E.T. Brown, J. Liu, C.E. Brodley, R. Chang, Dis-function: learning distance functions interactively, in: 2012 IEEE Conference on Visual Analytics Science and Technology, VAST, 2012, pp. 83–92.
[33] L. van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.
[34] P. Joia, D. Coimbra, J.A. Cuminato, F.V. Paulovich, L.G. Nonato, Local affine multidimensional projection, IEEE Trans. Vis. Comput. Graph. 17 (12) (2011) 2563–2571.
[35] J.B. Kruskal, M. Wish, Multidimensional Scaling, Quantitative Applications in the Social Sciences Series, vol. 11, Sage Publications, Newbury Park, 1978.
[36] C. Gormley, Z. Tong, Elasticsearch: The Definitive Guide, 1st edition, O'Reilly Media, Inc., 2015.
[37] R. Feldman, J. Sanger, Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data, Cambridge University Press, New York, NY, USA, 2006.
[38] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[39] Z. Chen, H. Doss, Inference for the number of topics in the latent Dirichlet allocation model via Bayesian mixture modelling, J. Comput. Graph. Stat. 0 (ja) (2018) 1–44, https://doi.org/10.1080/10618600.2018.1558063.
[40] Lexisnexis, http://www.lexisnexis.com/hottopics/lnacademic/, 2017. (Accessed 15 March 2017).
[41] M. Slezak, South Australia's blackout explained and no, renewables aren't to blame, https://www.theguardian.com/australia-news/2016/sep/29/south-australia-blackout-explained-renewables-not-to-blame, 2016.
[42] E. Henson, Seven tornadoes hit SA on day of massive blackout, www.adelaidenow.com.au/news/south-australia/seven-tornadoes-hit-sa-on-day-of-massive-blackout-bureau-of-meteorology-report/news-story/e888d155c01b910778132d68e93c9d6a, 2016.
[43] S. Wold, K. Esbensen, P. Geladi, Principal component analysis, in: Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists, Chemom. Intell. Lab. Syst. 2 (1) (1987) 37–52, https://doi.org/10.1016/0169-7439(87)80084-9.
[44] G. Salton, The SMART Retrieval System—Experiments in Automatic Document Processing, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1971.
[45] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: bringing order to the web.
[46] T. Hofmann, Probabilistic latent semantic indexing, in: Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99, ACM, New York, NY, USA, 1999, pp. 50–57.
[47] F. Wei, S. Liu, Y. Song, S. Pan, M.X. Zhou, W. Qian, L. Shi, L. Tan, Q. Zhang, Tiara: a visual exploratory text analytic system, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10, ACM, New York, NY, USA, 2010, pp. 153–162.