Context Preserving Dynamic Word Cloud Visualization

Weiwei Cui Yingcai Wu * Shixia Liu Furu Wei Michelle X. Zhou†
Hong Kong University of Science and Technology IBM China Research Lab
Huamin Qu‡
Hong Kong University of Science and Technology

ABSTRACT

In this paper, we introduce a visualization method that couples a trend chart with word clouds to illustrate temporal content evolutions in a set of documents. Specifically, we use a trend chart to encode the overall semantic evolution of document content over time. In our work, semantic evolution of a document collection is modeled by varied significance of document content, represented by a set of representative keywords, at different time points. At each time point, we also use a word cloud to depict the representative keywords. Since the words in a word cloud may vary one from another over time (e.g., words with increased importance), we use geometry meshes and an adaptive force-directed model to lay out word clouds to highlight the word differences between any two subsequent word clouds. Our method also ensures semantic coherence and spatial stability of word clouds over time. Our work is embodied in an interactive visual analysis system that helps users to perform text analysis and derive insights from a large collection of documents. Our preliminary evaluation demonstrates the usefulness and usability of our work.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1 INTRODUCTION

In recent years, tag clouds, which use a compact visual form of words, have been used widely to provide the content overview of a website (e.g., on Flickr and del.icio.us.) or a set of documents. Existing efforts in producing effective tag clouds have achieved certain success especially in addressing many aesthetic issues, for example, preventing overlapping tags, large whitespace and adhering to specific boundaries [17]. However, existing tag clouds are inadequate in portraying temporal content evolution of a set of documents (e.g., illustrating temporal content similarity or discrepancies).

For example, to understand how the US presidential speeches have varied during the last decade could be a difficult task if we just visualize the presidential speech collections one by one using tag clouds. A simple animation between different tag clouds at different time points would be inadequate to preserve the context for effectively tracking the evolution of the content to find the sequential patterns or correlations. Especially, if the changes from one word cluster to another is substantial (e.g., multiple words disappearing or appearing), users might get lost in such changes. More importantly, it is difficult to locate key frames, important word clusters at certain time points, in a long animation sequence. In particular, users may need to go through the whole animation sequence before they can find interested word clusters. Alternative solutions include line graphs and standard time series views showing trends for different words. However, these techniques may not reveal complex correlations among multiple words.

To facilitate the understanding of temporal content evolution in a set of documents, we propose a visualization method that couples a trend chart with word clouds to visually illustrate the content evolution. First, we show how to use a trend chart to encode the overall evolution of document content over time. In our work, content evolution of a document collection is modeled by varied significance of document content, represented by a set of representative keywords, at different time points. Second, we explain how we use a layout algorithm to display word content and their content differentiations at different time points. Specifically, we use geometry meshes and an adaptive force-directed model to lay out word clouds to ensure semantic coherence and spatial stability, which in turn helps preserve the visual context.

To the best of our knowledge, our work is the first to generate context-preserving visualization that uses tag clouds to depict evolving text content over time. As a result, our work offers two unique contributions:

- Two-level visualization that couples a trend chart and dynamic word clouds to illustrate temporal content evolution at multiple levels of detail.
- Time-based tag cloud layout that balances semantic coherence of content and spatial stability of the visualization to help users easily perceive content updates as well as ensure smooth visual transitions between successive tag clouds.

2 RELATED WORK

Over the past years, much effort has gone into the tag cloud visualization [17, 12]. Tag clouds have been used in a wide diversity of applications ranging from the analytical to the emotional [20, 21], even though tag clouds might be more difficult to navigate with than simple lists of words [10, 14] The tag cloud visualization can roughly be categorized into two groups: static tag cloud visualization and dynamic tag cloud visualization. The static tag cloud visualization focuses on addressing common issues (e.g., to avoid overlapping) to improve the overall readability while the dynamic tag cloud visualization illustrates the content evolution in a stream of documents.

Static Tag Cloud Visualization The most popular static tag cloud visualization is a rectangular tag arrangement with alphabetical or relevance sorting in a sequential line-by-line layout. Many researchers have implemented variants of this approach. There has been some work that focuses on addressing common issues such as large white spaces, overlapping tags and restriction to specific boundaries. For example, To reduce white spaces, Kaser and Lemire [12] introduced algorithms to optimize the display of tag clouds by leveraging prior work in typesetting, rectangle packing, and electronic design automation. Seifert et al. developed a family of algorithms which inscribe tags into arbitrary convex polygons with little white space [17]. More recently, Feinberg [9] developed Wordle to efficiently use the typographical space. Clark [4] used word "relatedness" to control positioning in a tag cloud layout. In

*e-mails: {weiwei,wuyc}@cse.ust.hk
†e-mails: {liusx, weifuru, mxzhou}@cn.ibm.com
‡e-mail: huamin@cse.ust.hk

IEEE Pacific Visualisation Symposium 2010
2 - 5 March, Taipei, Taiwan
978-1-4244-6686-3/10/$26.00 ©2010 IEEE

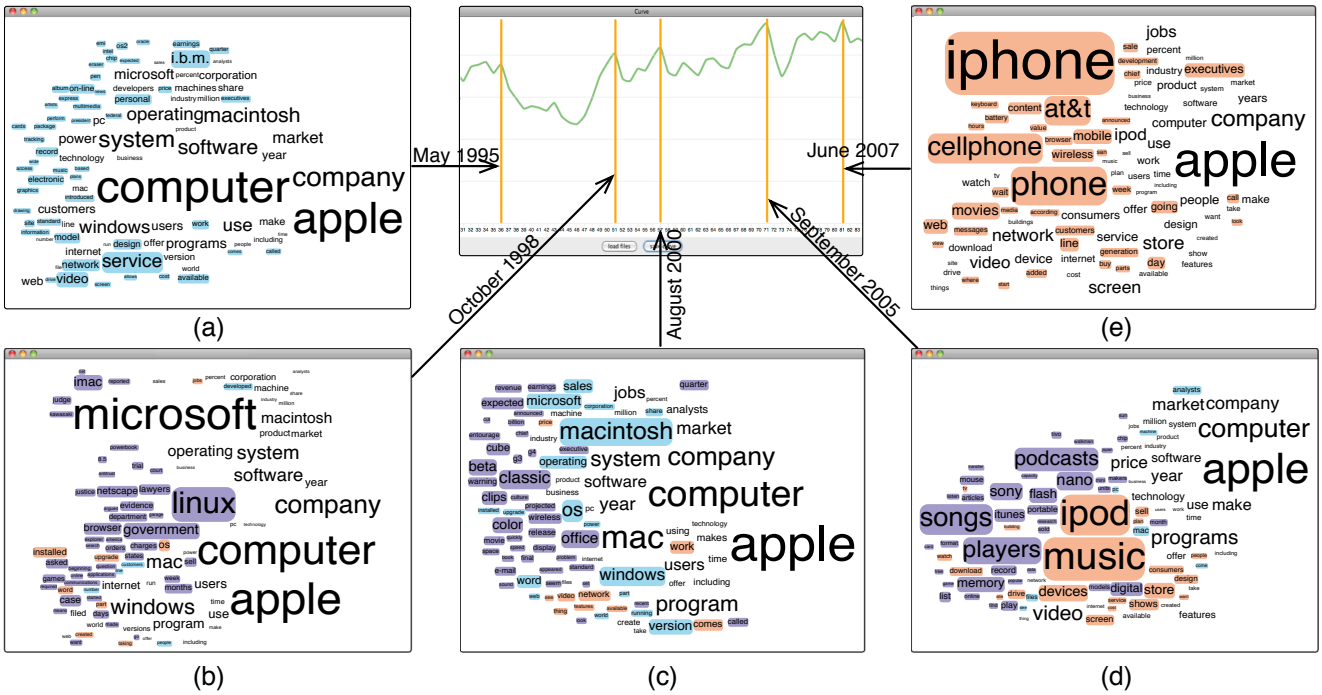

Figure 1: System overview. The top center of the figure presents a significance trend chart viewer which shows a significance curve extracted from a collection of documents with different time stamps. The x-axis encodes the time and the y-axis encodes the significance of the word clouds. The green curve in the chart represents the measured significance of the word clouds at different time steps. Five word clouds ((a)-(e) in the figure) are created using our algorithm for five selected time points where high significance values are observed.

addition, researchers have approached other visual metaphors to illustrate the popular words used in a set of text documents. Bielenberg [1] proposed a circular layout to display word clouds where important words are placed closer to the center. Shaw [18] proposed to display tag clouds using a graph layout whose nodes represent tags and edges indicate the relations between tags. Stefaner [19] presented an algorithm to generate elastic tag clouds where tags are placed in a nearly 2D circular space based on PCA and CCA. In contrast, our work aims at using dynamic word clouds to analyze the content evolution in a stream of text documents and thus our work mainly focuses on balancing the semantic coherence and spatial stability of word clouds over time.

Dynamic Tag Cloud Visualization Dubinko et al. [8] proposed to visualize the evolution of tags in the Flickr, in which users can observe and interact with interesting tags as they evolve over time. A tool called *cloudalicious* [15] was also developed to visualize the evolution of tag clouds over time. Compared to these two approaches which merely generate an animation of tag evolution, our approach provides a significance trend chart depicting the variation of word clouds over time, such that users can get a visual overview of the varying trend of the word clouds over time. Furthermore, our approach employs a geometry-based method to generate word cloud layouts to well balance the semantic coherence and spatial stability of word clouds over time. Other related work includes stacked graphs, which are useful visualization techniques for visual analysis of quantitative variations of a set of items over time [3, 11]. Although they could be used for visualization of a dynamic tag cloud, the correlation among important words that can be conveyed by word clouds is lost. Collins et al. [5] introduced parallel tag clouds (PTCs) by taking the advantages of both parallel coordinates and traditional tag clouds. PTCs may also be used to visualize the content evolution of a stream of text documents when each column of the PTCs shows the important words of a stream of documents at a certain time point. Compared with PTCs, our method is more intuitive and does not require expertise on parallel coordinates

3 System Overview

Fig. 1 provides an overview of our system with its two main components: *trend chart viewer* and *word cloud generator*. The trend chart viewer depicts the varied significance of document content, represented by a set of representative words, over time. The green curve shown in the top of Fig. 1 presents the varied significance of the word clouds extracted from a stream of text documents. The x-axis encodes the time and the y-axis encodes the significance of the word clouds. Figs. 1(a)-(e) are the selected important word clouds which constitute a storyboard for visually presenting a story to users (Details can be found in the case study in Section 6). To further improve word readability, the words are assigned with different background colors depending on their appearing behaviors. The words in purple are the unique words that only appear in the current time point. The red background indicates that the corresponding words appear in the succeeding time point. The blue background shows that the corresponding words just appear in the preceding time point. The words which appear in both the preceding and succeeding time points do not have background color.

4 Significance Analysis

In this section, we analyze the dynamic behaviors of the words inside the clouds, i.e., their different spatio-temporal behaviors, and study how to visually present those dynamic behaviors to users effectively. Inspired by a research paper [22] in time-varying volume visualization, we introduce an information-theoretic approach for depicting the varied semantic significance of document content, represented by dynamic word clouds. In the paper [22], the volume data at a certain time point with more information by itself and less

information shared by those at other time points is considered to be more significant. Based on this observation, the importance of volume data at different time points is then evaluated by information measures in a quantitative manner. Our system estimates the significance of dynamic word clouds using the similar strategy. A word cloud is more significant if it conveys more information by itself with less information shared by other word clouds.

4.1 Entropy and Information Theory

Before we go deep into our information-theoretic significance approach, we briefly introduce several important information measures from information theory including information entropy, mutual information, and conditional information.

In information theory, the information entropy measures the amount of information or uncertainty of a random variable. The information entropy can usually be computed as follows:

$$H(X) = -\sum_{x \in X} p(x) \log p(x) \tag{1}$$

where X is a discrete variable with n possible values $\{x_1 \ldots x_n\}$, and $p(x)$ is the marginal probability distribution function of X. When the system has only one value, the entropy reaches its minimum as 0. If the system takes all possible n values of X with equal probability, the entropy is maximized as $\log n$.

The mutual information, on the other hand, is a measure of the dependence between two discrete random variables X and Y. Given a joint probability distribution function $p(x,y)$ and two marginal probability mass functions $p(x)$ and $p(y)$, the mutual information can be defined as:

$$H(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{2}$$

The mutual information $H(X;Y)$ evaluates the information shared by X and Y. From the above definition, we can see that $H(X;Y)$ is symmetric, i.e., $H(X;Y) = H(Y;X)$. It is also nonnegative and is equal to zero if and only if X and Y are independent, namely, $p(x,y) = p(x)p(y)$. Thus, knowing X does not provide any knowledge about Y and vice versa. On the other hand, if X and Y are identical then the information contained in X is totally shared by Y, i.e., $H(X;Y) = H(X) = H(Y)$.

The conditional entropy $H(X|Y)$ of a random variable X given another random variable Y can be defined as the expected value of the entropies of $p(x|y)$, i.e., $-\sum_x p(x|y) \log p(x|y)$, averaged over Y.

$$H(X|Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log p(x|y) \tag{3}$$

which also can be calculated as:

$$H(X|Y) = -H(X;Y) + H(X) \tag{4}$$

For more details in the measures, interested readers can refer to an excellent book [6]. The conditional entropy $H(X|Y)$ is nonnegative. It is equal to 0 if and only if X and Y are identical, i.e., $H(X) = H(X;Y)$. Conversely, we have $H(X|Y) = H(X)$ if and only if X and Y are independent.

4.2 Information-Theoretic Significance Estimation

A word cloud is considered to be more significant if it contains more information while sharing less information with others. Therefore, the word cloud significance can be estimated in a quantitative manner based on the conditional information measure. To measure the word cloud significance, we first evaluate the information entropy $H_t(X)$ of a word cloud at time point t, and then estimate the mutual information $H_t(X;Y)$ between the word cloud and those in its neighboring time points. Following Equ. (4), we can derive $H_t(X|Y)$ representing the significance of the word cloud at time point t using the obtained $H_t(X)$ and $H_t(X;Y)$.

4.2.1 Information Entropy Estimation

To quantify the information entropy $H(X)$ for a word cloud, a feature vector is needed to characterize each word in the cloud from multiple perspectives. In our system, the feature vector consists of the word frequency (the font size) in the text, the word positions in the cloud, and the displayed color of the word. A multidimensional histogram can be built upon the feature vectors of the words in the cloud. Each bin in the multidimensional histogram counts the number of words that fall into a certain disjoint feature value intervals. To strike a balance between the performance and storage demand, after some initial experiments we set the number of intervals as 64 for each dimension in the histogram, which can provide acceptable results as we expect. The information entropy $H(X)$ of the word cloud is computed by the normalized count of every bin of the histogram, i.e., $p(x)$ in Equ. (1). The difference here is that we consider feature vector $\mathbf{x}$ instead of scalar value x. For example, the information entropy of a three dimensional discrete random variable X can be derived as follows:

$$H(X) = -\sum_{a \in X_1} \sum_{b \in X_2} \sum_{c \in X_3} p(a,b,c) \log p(a,b,c) \tag{5}$$

where X_1, X_2, and X_3 are element random variables of X in those three dimensions, respectively.

4.2.2 Mutual Information Estimation

Given two word clouds X and Y, besides the marginal probability $p(x)$ and $p(y)$ described in Section 4.2.1, we need to establish a two-dimensional joint histogram to compute the joint probability $p(x,y)$ for estimating the mutual information $H(X;Y)$ according to Equ. 2. We define the information shared between X and Y as the common words that are in both clouds. The remaining words that are in one cloud but not in the other are considered to be independent. Thus, the joint histogram is constructed by counting the number of words that fall into a particular interval of a combination of the feature values including the word frequency, position, and color values of one word cloud. The mutual information $H(X;Y)$ is then computed by the normalized count of every bin of the joint histogram, i.e., $p(x,y)$, as well as $p(x)$ and $p(y)$ measured based on the multidimensional histogram in Section 4.2.1.

4.2.3 Conditional Entropy Estimation

For a given word cloud X at time point t, we consider several neighboring word clouds preceding or succeeding X to estimate the conditional information entropy of X. This means that we choose the word clouds of the time points within a given window centered at t for the conditional entropy estimation for X. Suppose t_i is the weight of the word cloud Y_i in the window and *Size* is the size of the window, the significance of X can be defined as follows:

$$S(X) = \sum_{i=1}^{Size} t_i \cdot H(X|Y_i) = \sum_{i=1}^{Size} t_i \cdot (H(X) - H(X;Y_i)) \tag{6}$$

Here, the summarization of all t_i is one, i.e., $\sum_i^S t_i = 1$.

4.3 Significance Trend Chart

With the estimation of conditional information, a significance curve showing the varied semantic significance of document content can be presented to users. This curve eases the difficulty for users to find the important documents, represented by word clouds, from a large collection of documents quickly. The user interactions supported by the system are as follows:

- **Selecting word clouds:** A sliding bar is provided at the bottom of the chart. Users can selectively visualize a specific word cloud by sliding the bar on the chart.

- **Expanding the significance curve:** Whenever users select the significance curve by clicking the mouse, the chart will perform animation to expand the green curve to a belt for providing more details.
- **Creating a storyboard:** Users can create a storyboard of the documents by selecting significant word clouds based on the trend chart and putting them together to tell a story. The important word clouds of the storyboard are selected either automatically by choosing the word clouds at the time points where a peak on the significance curve is found, or manually by indicating the important word clouds on the curve. For example, in Fig. 1, users clicked five time points on the curve. After that, five separated windows were opened to show the corresponding word clouds.

5 Layout of Word Clouds

Visualizing the documents with different time stamps one by one using conventional word clouds is not an easy task for most text analysts, because the word clouds are typically designed for static documents. Words between two consecutive word clouds usually vary with font sizes and positions. Some words may even appear or disappear frequently over time. Too many of those word variations often distract users. Thus, text analysts may find the words hard to follow and track over time. In this section, we introduce a new flexible method to create word cloud layouts specifically for documents with different time stamps. The method can organize the layouts according to different semantic coherence criteria, including *a similarity criterion*, *an importance criterion*, and *an co-occurrence criterion*, to meet different user requirements.

Fig. 2 shows our pipeline for creating word cloud layouts for the documents with time stamps. The pipeline begins with an initial set of important words extracted from a collection of documents (see Fig. 2(a)). The extracted words are then placed on the 2D plane based on their attributes (see Fig. 2(b)). For every time slot, the words that are unimportant or unrelated to the documents are filtered out from the 2D plane (see Fig. 2(c)). After that, our system performs Delaunay Triangulation of the remaining points, each of which locates at the center of a word, to generate a triangle mesh. The font size of each word is determined based on the corresponding word frequency in the time slot, as shown in Fig. 2(d). Finally, an adapted force-directed algorithm is used to adjust the point positions for obtaining an appropriate layout (see Fig. 2(e)).

5.1 Word Extraction

Consider n documents: $T = \{T_1, T_2, \cdots, T_n\}$ with different time stamps. For document T_i, we first remove the most common words that are unimportant and uninteresting. For example, word "a", "the", "that", "thus", and so on, will be removed. After that, the system builds a histogram $Hist_i$ to indicate the frequency of all unique words used in the document T_i. We then employ the *Porter Stemming Algorithm* [13] to combine similar words and their corresponding bins in the histogram based on whether they have the same root. The grouped words are represented by the most common variation in the document. For instance, our approach may group the words "fishing", "fish", "fisher", and "fished" under the most common variation "fishing" in a document. Finally, the remaining words in the histogram $Hist_i$ with frequency higher than a user-specified threshold are selected as the candidate word set denoted by W_i for T_i to create the word cloud, as it is usually unnecessary to present all the words to users. Finally, we can obtain a set of extracted words $W = \{W_1, W_2, \cdots, W_n\}$ for the document set T. The word set W_i represents the set of important words that will be displayed in the word cloud for document T_i.

5.2 Initial Word Placement

With the extracted word sets W, we place all the important words, i.e., the $\bigcup W$, on the 2D plane to create an initial word layout where words are semantically grouped. This can improve the readability of the word clouds because the words are being organized and displayed as semantically coherent clusters rather than being presented in alphabetical order in the clouds. Thus, users can understand the major content of the documents efficiently, as they do not have to examine all words one by one but just need to quickly look at the word clusters instead. Clustered words can also ease the difficulty of tracking the document content over time using word clouds.

Our system generates the layout in different styles by the following semantic coherence criteria to meet different user requirements.

- **Importance criterion:** this criterion aims at creating a layout where words are clustered based on their importance values at different time points. In other words, it can group the words that have similar variation of their font sizes over time, as the importance values are represented by font sizes in word clouds. Hence, semantically important words will appear together.
- **Co-Occurrence criterion:** this criterion can ensure that the words with similar appearing or disappearing behavior over time will be clustered in the resulting layout, namely, the words that appear or disappear simultaneously in most time will have a high chance to be grouped together. Hence, semantically similar words will be updated simultaneously.
- **Similarity criterion:** this criterion is used to create a layout such that the semantically similar words can be clustered. This means that the words with similar semantic meanings in the documents will be close to one another in the layout. As a consequence, semantically similar words will appear together.

To apply these criteria, we need to establish an appropriate feature vector for each criterion to perform clustering. Given an extracted word $wd_p \in \bigcup W$, we can define three types of feature vectors, namely, importance vector V_i, co-occurrence vector V_a, and similarity vector V_s, as follows.

- **Importance vector:** The importance vector is used to capture the font size variation of each word over time. The feature vector V_i is defined as $V_i = \{v_1, v_2, \cdots, v_n\}$ where n is the number of time points of the documents. v_j is the importance value (i.e., the font size) of wd_p at time point j.
- **Co-Occurrence vector:** The co-occurrence vector encodes the characteristic of the appearing or disappearing behavior of each word over time. It can be defined as $V_a = \{v_1, v_2, \cdots, v_n\}$ where n is the number of time points of the documents. The element v_j equals 1 if the word wd_p is visible at time point j, otherwise v_j becomes zero.
- **Similarity vector:** We employ a well-established method from [16] to define a feature vector for each word to characterize its semantic relations to other words. The feature vector can be defined as $V_s = \{v_1, v_2, \cdots, v_m\}$ where m is the number of words in $\bigcup W$. The element v_q represents the number of times that the word $wd_q \in \bigcup W$ occurs close to wd_p (within a sentence or a larger context) in the documents. Intuitively, we can roughly estimate the semantic similarity between two words by measuring the amount of overlap between their corresponding vectors, since semantically similar words usually share similar neighbors, namely, their vectors have considerable overlap.

The similarity between two vectors V_p and V_q can be evaluated by the cosine measure.

$$\cos(\theta) = \frac{V_i \cdot V_j}{||V_i|| \cdot ||V_j||} \tag{7}$$

The higher the value of cosine, the more similar the two corresponding words. The value of the cosine is 1.0 if and only if two words

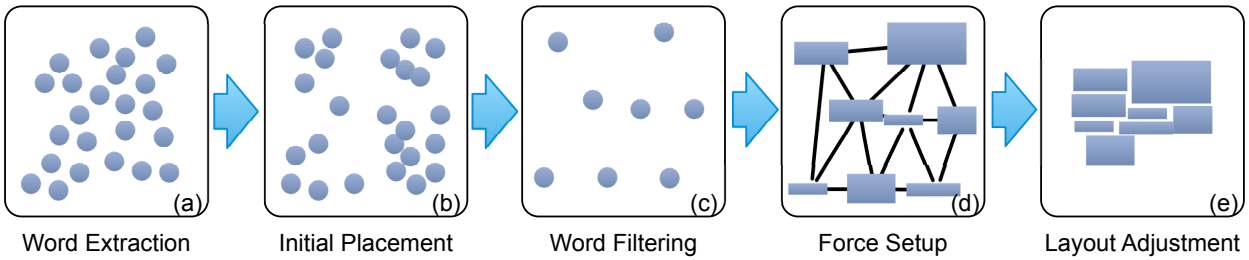

Figure 2: Pipeline for creating a semantic and stable word cloud layout: (a) Extracting an initial set of words from all the documents with different time stamps; (b) Placing the extracted words on the 2D plane using multidimensional scaling; (c) Filtering out unrelated words for a specified time slot; (d) Triangulating the remaining words; (d) Optimizing the layout by a force-directed algorithm.

share exactly the same characteristics, i.e., the two words are of perfect match. In contrast, the value of the cosine is 0.0 if and only if two words are totally irrelevant.

With the vector representations and the similarity measurement, we can create a dissimilarity matrix Δ where its element $\delta_{p,q}$ represent the similarity ($\cos(\theta)$ in Equ. 7) between word p and word q. With Δ, we then employ *Multidimensional Scaling* (MDS) [2] to reduce each high-dimensional vector to a two-dimensional point, so that we can obtain an initial word layout where related words are placed in a semantically clustered manner on the 2D plane.

5.3 Delaunay Triangulation

The initial word layout contains all important words (i.e., $\bigcup W$) of the whole collection of the documents. Nevertheless, users may only want to visualize some documents in a short time slot. In this case, the system filters out the unimportant or unrelated words in the initial layout. This often creates a very sparse layout (see Fig. 2(c)) where much space is wasted. To reduce the empty space between the remaining words, we need to pack the words in the layout. On the other hand, the semantic relations between the words are represented by the relative positions between the words implicitly. This information is critically important for the analysis of the documents. Thus, the relative positions should be preserved in the packed layout. We achieve this by using a triangle mesh as the control skeleton to maintain the original relative positions. We perform *Delaunay Triangulation* [7] on the word positions to obtain the mesh which can be denoted as an initial graph $G = (V, E)$. With the graph, we can rearrange the word positions on the 2D plane flexibly to reduce empty space while keeping the semantic relations between the words.

5.4 Force-Directed Model

With the initial graph G, we can start to build a word cloud layout. An adapted force-directed algorithm is proposed to reposition the vertexes V in the graph G and remove most empty space. In this process, we can largely preserve the semantic relations between the words since the topology of the graph G which encodes the underlying semantic word relations remains unchanged.

We establish three design principles to design an appropriate force-directed algorithm which can maintain the graph topology while removing most empty space between the words. These principles are listed as follows.

- **Compact principle:** This principle aims at removing empty space between the words as much as possible so that the created layout is compact. Compared with other principles, it has the lowest priority.
- **Overlapping principle:** This principle requires that the words should not overlap one another. It takes the top priority over all other principles to guarantee the readability of each word in the resulting layout.

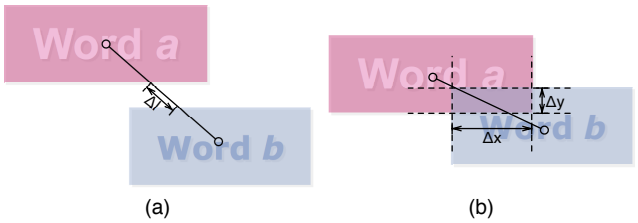

Figure 3: (a) Two separated words exert a spring force on the connected edge. (b) Two overlapped words exert a repulsive force on the connected edge.

- **Planar principle:** This principle is used to make sure that the controlling mesh (i.e., the initial graph G) should stay as planar as possible. It helps word clouds keep their semantic relations among words. The principle has lower priority than the overlapping principle and does not need to be strictly followed because: (1) keeping the semantic relations does not imply that the mesh should be strictly planar; (2) keeping the mesh strictly planar may lead to an unnecessary waste of space.

Following these principles, a force-directed model is developed to ensure that the created word cloud layout is compact, easy to read, stable, and semantically meaningful. The model has three basic forces, namely, a spring force, a repulsive force, and an attractive force, corresponding to the three principles, respectively.

We employ the spring force to remove empty space and pack words compactly. Suppose words a and b are connected in G. The spring force between them can be defined as follows.

$$f_s(a,b) = w_a w_b \Delta l \tag{8}$$

where w_a and w_b are the importance values of words a and b, respectively, and Δl represents the length of the connected edge that lies outside of both a and b as shown in Fig. 3(a).

The repulsive force is used to prevent a word being occluded by other words. The force becomes effective between two words if and only if they overlap each other, otherwise the force does not exist. The repulsive force f_r is formulated as follows:

$$f_r(a,b) = \begin{cases} k_r \min(\Delta x, \Delta y) & \text{if word } a \text{ overlaps word } b \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where k_r is a given weight, and Δx and Δy are the width and the height of the overlapping region as illustrated in Fig. 3(b).

We use the attractive force to make sure that our created layouts are stable and semantically meaningful. During the process of layout adjustment, if a mesh triangle is flipped, i.e., one vertex in the triangle goes to the other side of its subtense, the mesh will become nonplanar (see Fig. 4(b)). In this case, the attractive force between

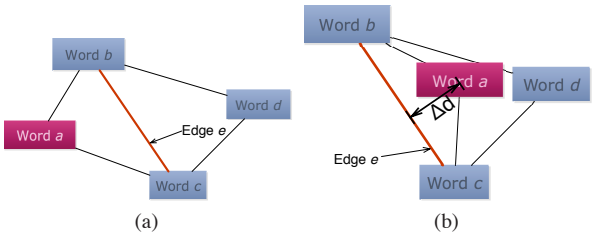

Figure 4: (a) The attractive force between edge e (drawn in red) and word a (drawn in red) is zero if the mesh is planar. (b) The attractive force becomes effective if a is flipped to the other side of e.

the subtense and the vertex will become effective to flip the triangle back (see Fig. 4). The force f_a is formulated as follows.

$$f_a(a,l) = \begin{cases} k_a \Delta d & \text{if word } a \text{ is flipped} \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

where k_a is a given weight and Δd is the distance between word a and its subtense e.

Since the three basic forces have different priorities, we should choose k_r and k_a appropriately according to the priorities of the design principles. For example, we can set $k_r >> k_a >> w_{max}^2$ where w_{max} is the maximum importance value of the words.

6 Experiments and Discussion

In this section, we tested our system components by several experiments. A case study was also conducted for showing the usefulness of our system. We finally discuss the limitations of our system and the future work.

6.1 Experiments

The system was tested on an iMac desktop computer(3.06GHz Intel Core Duo CPU, 4GB RAM, GeForce GT 130 with 512MB RAM). All time-consuming operations, such as the significance estimation and the initial word placement, can be performed at a preprocessing stage. Thus, all results including the significance curve and the word clouds can be obtained interactively. For the largest testing data set which consists of several thousands of documents each of which has several hundreds of words, our preprocessing tasks of the data set were all completed in a couple of minutes.

We did the first experiment to test the correctness of our layout generation method. The testing data set contains only eight capital names with their positions reflecting the geographical locations. The font sizes were set randomly. Figs. 5(a) and 5(b) show the initial word layout and its mesh generated by Delaunay Triangulation, respectively. Figs. 5(c) and 5(d) show a sparse word cloud and the mesh captured at a step during the layout adjustment process. Figs. 5(e) and 5(f) are the final word cloud layout and its mesh. The initial layout has severe clutter (see Fig. 5(a)). From the figures we can see that the repulsive force first dominated the adjustment and separate those overlapped words from one another (see Fig. 5(c)). However, this made some triangles (drawn in red in Fig. 5(d)) flipped in the mesh . Then the attractive force became effective and made them flip back. Meanwhile, the spring force was also exerted on the edges between separated words to pack the words. Fig. 5(e) is the resulting word layout where its mesh became planar finally (see Fig. 5(f)), respectively. All the words in Fig. 5(e) are still semantically coherent to those in Fig. 5(a).

We tested our system in the second experiment to show the effectiveness of the created semantically coherent layouts. This experiment was conducted on 13,828 news articles related to the AIG company spanning over one year (from Jan. 14, 2008 to Apr. 5, 2009). We generated a sequence of word clouds and chose two

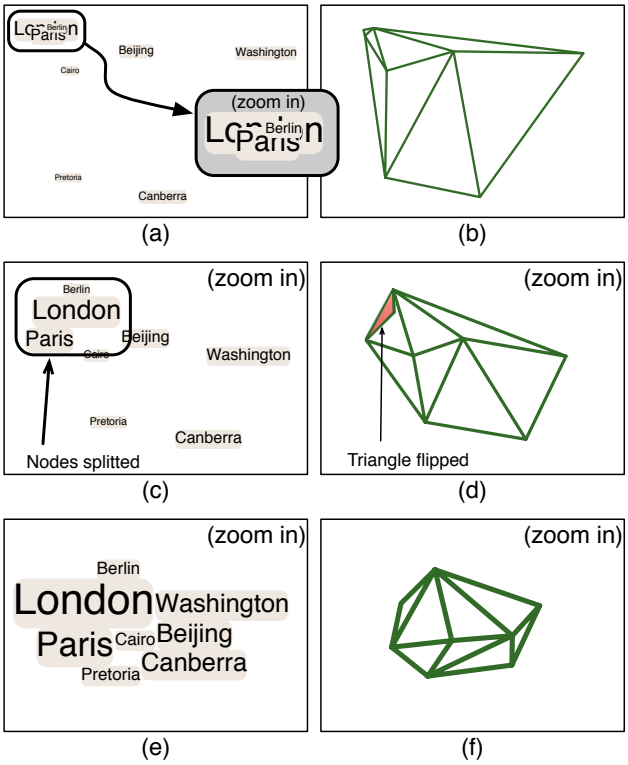

Figure 5: Word cloud layouts and their corresponding meshes generated in the process of our adapted force-directed algorithm.

neighboring word clouds for demonstration. For every word cloud, we generated two different layouts using our method (see Figs. 6(a) and 6(b)) and Wordle [9] (see Figs. 6(c) and 6(d)), respectively. Our layouts are comparable to those created by Wordle with respect to the layout compactness. More important, our layouts have two unique advantages over those of Wordle. First, our layouts can present more semantically meaningful information. For example, by just looking at the cluster (marked in blue) in 6(a), we can easily tell that the underlying documents talk about the economy and president election together, since a group of economy words appear together with "Obama" and "Mccain". On the other hand, users may be able to see "Obama" or "Mccain" separately (marked in blue in Fig. 6(b)). However, since they are far from each other, users may have less idea about the content topic (i.e., U.S. president election). The complete information would be broken into pieces if all the related words are randomly placed. Second, our method is very efficient to help users compare and track different word clouds over time because the semantically clustered words can greatly narrow down the visual search space. For example, users can easily track variation of keywords (e.g., "economy" and "Obama") between Figs. 6(a) and 6(b) and figure out that "economy" becomes smaller and "Obama" disappears. In contrast, it may take them much longer time to do so in Figs. 6(c) and 6(d) because of the much larger visual search space. We further measured the average offset of all the shared words between Figs. 6(a) and 6(b), and then between Figs. 6(c) and 6(d). The average offset by using our method is 141 pixels, while it is 313 pixels by using Wordle. The results indicate that users need to visually search a larger space using Wordle than using our method to find the common words between two word clouds in this example.

The next experiment was conducted to demonstrate the different uses of our three semantic coherence criteria. We used the same data set in the second experiment, but with different semantic crite-

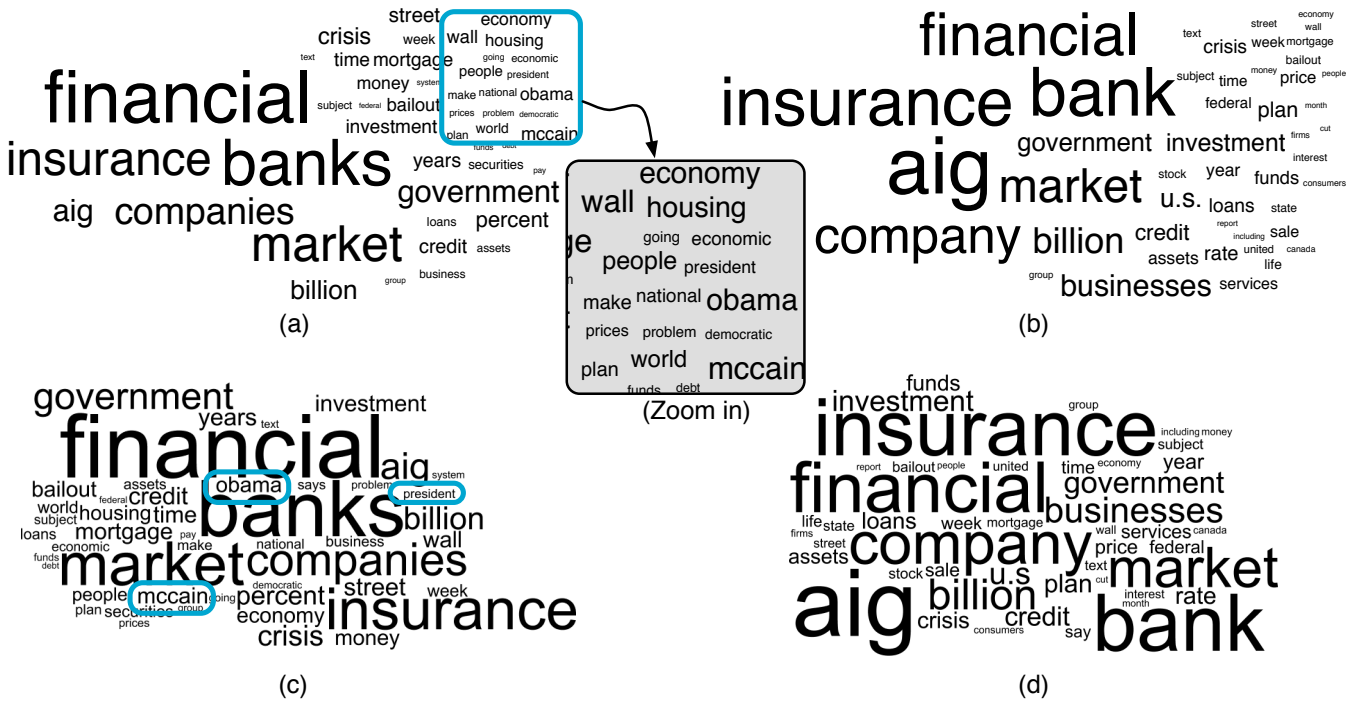

Figure 6: Comparison of the word cloud layouts created by our method (word clouds in (a) and (b)) and by Wordle (word clouds in (c) and (d)). Word clouds (a) and (c) as well as those in (b) and (d) are generated for the documents at the same time point.

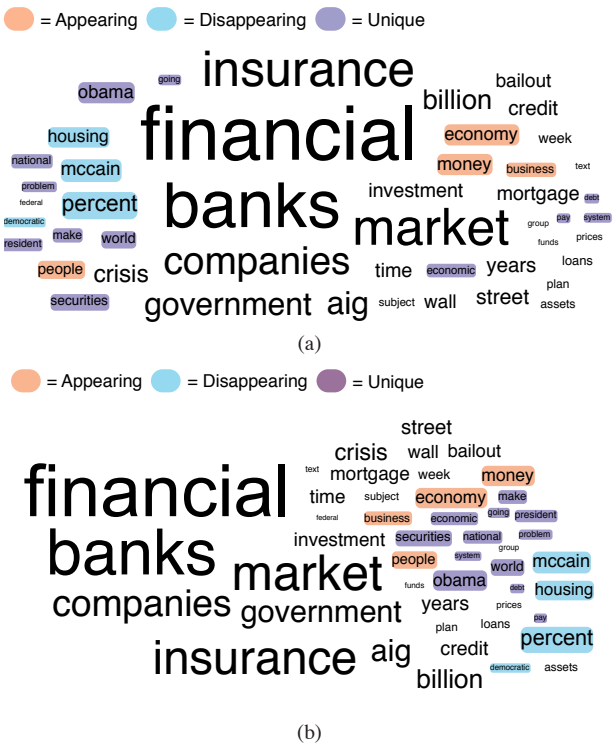

Figure 7: Two word cloud layouts (a) and (b) generated by the importance criterion and the co-occurrence criterion, respectively.

ria for showing different styles of word clouds. Fig.6(a) is the word cloud layout generated by the similarity criterion. Figs. 7(a) and 7(b) present another two layouts of the same data using the importance criterion and the co-occurrence criterion, respectively. The words are highlighted by different colors according to their appearing behavior. We can see that the words with the same color are roughly clustered together in Fig. 7(b). In contrast, most words are grouped together according to their font sizes in Fig. 7(a). With the experiment, we know that our technique successfully grouped words according to the semantic coherence criteria in this example.

The last experiment was a case study for demonstrating the usefulness of our system. This experiment was carried out on a collection of 1,933 news articles related to Apple Inc. from August 1989 to August 2009. The top center of Fig. 1 shows the computed significance curve for presenting an overall picture about the content evolution of the news articles. From the chart, we can see that the part of the significance curve starting from August 1998 appears generally higher than that before August 1998. This is in accordance with the event that Steve Jobs returned to Apple in late 1996. After his return, Apple kept staying in the spotlight and attracting more eyeballs by creating various hot topics. Although the curve is roughly higher on the right hand side, we can still observe several peaks at October 1998, August 2000, September 2005, and June 2007 (see the top center of Fig. 1). The last three peaks roughly match Apple's major product announcement or release dates in that year (May 2000, September 2005, and June 2007). We extracted all four word clouds at these peaks as well as a word cloud for May 1995, the time before Steve Jobs returned, for comparison. Figs. 1(a)-1(e) show a clear visual summary of Apple's several key steps during these years, from computer to iPod and finally to iPhone. There are several interesting patterns in the word clouds. For example, we can track the font size of "computer". With words being semantically clustered, we can find its positions in each word cloud efficiently, even though its font size is no longer big enough to attract users' attention in some figures, such as in Fig. 1(e). It is also

very clear that its font size decreases monotonically because computer became less important in Apple's major product line. iPod, released in 2000, and iPhone, released in 2007, became more popular. Therefore, the media moves its attention to these two products.

Another interesting pattern is the varying trend of the keywords about Microsoft. We can see that the words related to Microsoft disappear from Figs. 1(d) and 1(e). It may indicate that Apple has broaden its market so successfully that the media has no longer treated it as a competitor of Microsoft. In contrast, the keyword "Microsoft" in Fig. 1(b) appears abnormally big, which indicates that the two companies were strongly related. After investigation, we found an interesting event in an article of October 28, 1998, i.e., "Microsoft said that Apple agreed to adopt for its browser as part of broad agreement that included a $150 million investment by Microsoft ...". This event may be one of the reasons that got Microsoft into trouble with its antitrust lawsuit, which is a hot topic in news articles at that time. We thought these events (the investment and the lawsuit) might drove the relations of the two companies so close. Our semantic coherence layouts can also help us identify some interesting patterns surrounding a certain keywords. For example, we compared the words around "ipod" in Fig. 1(d) and words around "iphone" in Fig. 1(e). We could find out what features people were interested in for the two products, i.e., "memory" and "portable" for iPod, and "keyboard" and "battery" for iPhone. The results demonstrate that our word cloud layouts have presented an effective summary for this text corpus.

6.2 Discussion

Our experiments have shown the effectiveness and advantages of our system. Compared with existing layouts, our layouts work better for visualizing documents with time stamps, as semantically coherent words are grouped together to ensure spatial stability over time. Our significance estimation of word clouds is novel and useful for providing users with a visual summary of semantic variation of document content. Nevertheless, our methods still have a few limitations. Our method can create layouts as compact as those generated by existing methods in most time, but it may fail to deliver a packed layout when the initial layouts are very irregular (e.g., most words are placed on a straight line). This can be alleviated by setting higher priority for the spring force and lower priority for the attractive force. The initial layouts only depend on semantic information, thus they cannot be adjusted manually. We plan to improve this by enabling user interaction and integrating user knowledge into the initial layout generation. Our system allows users to create a storyboard from the documents. However, simply selecting the word clouds from the peaks on the significance curve may not be sufficient for telling the whole story. Thus, we would like to study the effective selection of word clouds for a story presentation from the trend chart.

7 Conclusions

In this paper, we have presented a context-preserving visualization for users to visually analyze a collection of documents.The visualization has two major components. The first component is a trend chart that depicts the varied semantic significance of document content, represented by a set of representative words, over time. The second component is a set of word clouds distributed over time to provide an overview of document content at different time points. To highlight the content changes, we visually depict the differences of word clouds at different time points while maintaining the semantic coherence and spatial stability of the word clouds. In the future, we plan to conduct a formal user study for thoroughly evaluating the usefulness of our system. With feedback from the user study, we will further improve our system.

Acknowledgements

Most of this work was done when the first two authors were on summer internships at IBM China Research Laboratory. This work was supported in part by grant HK RGC 618706, GRF 619309, and an IBM Faculty Award.

References

[1] K. Bielenberg. Groups in social software: Utilizing tagging to integrate individual contexts for social navigation. Master's thesis, Universitat Bremen, 2005.

[2] I. Borg and P. J. F. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2nd edition, 2005.

[3] L. Byron and M. Wattenberg. Stacked graphs: Geometry & aesthetics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1077–2626, 2008.

[4] J. Clark. Neoformix blog. `http://neoformix.com/`, Aug 2009.

[5] C. Collins, F. B. Viégas, and M. Wattenberg. Parallel tag clouds to explore and analyze faceted text corpora. In *IEEE Symposium on Visual Analytics Science and Technology*, 2009.

[6] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

[7] M. de Berg, M. van Krefeld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer, 2nd edition, 2000.

[8] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins. Visualizing tags over time. *ACM Transactions on The Web*, 1(2):1–22, 2007.

[9] J. Feinberg. Wordle. `http://www.wordle.net/`, Aug 2009.

[10] M. J. Halvey and M. T. Keane. An assessment of tag presentation techniques. In *Proceedings of the 16th international conference on World Wide Web*, 2007. Poster.

[11] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.

[12] O. Kaser and D. Lemire. Tag-cloud drawing: Algorithms for cloud visualization. In *Proceedings of the World Wide Web Workshop on Tagging and Metadata for Social Information Organization*, 2007.

[13] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[14] A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen. Getting our head in the clouds: toward evaluation studies of tagclouds. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 995–998, 2007.

[15] T. Russell. cloudalicious: folksonomy over time. In *JCDL*, pages 364–364, 2006.

[16] H. Schütze. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123, 1998.

[17] C. Seifert, B. Kump, W. Kienreich, G. Granitzer, and M. Granitzer. On the beauty and usability of tag clouds. In *International Conference Information Visualisation*, pages 17–25, 2008.

[18] B. Shaw. Utilizing folksonomy: Similarity metadata from the del.icio.us system. Project Proposal, December 2005.

[19] M. Stefaner. Visual tools for the socio-semantic web. Master's thesis, University of Applied Sciences Potsdam, 2007.

[20] F. B. Viégas and M. Wattenberg. Tag clouds and the case for vernacular visualization. *TIMELINES*, 15(2):49–52, 2008.

[21] F. B. Viégas, M. Wattenberg, and J. Feinberg. Participatory visualization with wordle. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1137–1144, 2009.

[22] C. Wang, H. Yu, and K.-L. Ma. Importance-driven time-varying data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1077–2626, 2008.