Bare Demo of IEEEtran.cls for IEEECS Conferences

Yan Wang

State Grid Shandong DongYing Electric Power Company
Jinan, China
Email: zdhkj2019@163.com

Mingyu Sun

State Grid Shandong DongYing Electric Power Company
Jinan, China
Email: zdhkj2019@163.com

Ning Hu

State Grid Shandong DongYing Electric Power Company
Jinan, China
Email: zdhkj2019@163.com

Sentao Liu

State Grid Shandong DongYing Electric Power Company
Jinan, China
Email: zdhkj2019@163.com

Juncheng Si

State Grid Shandong DongYing Electric Power Company
Jinan, China
Email: zdhkj2019@163.com

***Abstract*—Visual analytics play an important role in understanding complex datasets, e.g. the bibliographic data set. Word clouds, which use a compact visual form of words, have been used widely to provide the content overview of a set of documents, especially the hot topic trends. In this paper, we proposed a visualization method that embedding the temporal patterns along the word to visually illustrate the hot topic evolution.**

***Index Terms*—bibliographic dataset; word cloud; topic modeling; text visualization;**

I. Introduction

In recent years, word clouds (or tag clouds), which use a compact visual form of words, have been used widely to provide the content overview of a set of documents, e.g. the bibliographic data set. How to provide a pleasing summarization for a huge text data has thus become an important research topic in information visualization. Word clouds are text-based visual representations that display word significance in terms of popularity and importance by using different font sizes and colors. Existing efforts in producing effective tag clouds have achieved certain success especially in addressing many aesthetic issues. However, existing tag clouds are inadequate in portraying temporal content evolution of a set of bibliography documents. For example, to understand how a hot research topic have varied during the last decade could be a difficult task if we just visualize the bibliography 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.

To facilitate the understanding of temporal content evolution in a set of bibliography documents, we propose a visualization method that embedding the temporal patterns along the word to visually illustrate the hot topic evolution.

II. Related Work

We review previous work related to word cloud visualization and temporal word clouds.

A. Word Cloud Visualization

A word cloud, also known as a tag cloud, is a visual representation of text data that has been used on the web since 1997[1]. A word cloud encodes the frequency of words of a given text into font size and color[2], and spatially arranges the words on the canvas. Standard word cloud visualizations use a rectangular line-by-line layout, where the words may be sorted alphabetically or by their importance. To produce more compact and aesthetic visualizations, a large family of alternative layout methods have been proposed[3], [4], [5], [6], [7], [8]. Among them, the most well-known algorithm is EdWordle[8], a method for consistently editing word clouds, which allows users to move and edit words while preserving the neighborhoods of other words. However, such enhancements still do not capture the relationship between words, let alone the temporal coherence of time-varying text data. Therefore, a variety of temporal word cloud generation methods have been proposed in recent years.

B. Temporal Word Clouds

Given a time-varying set of words, temporal word clouds attempt to visualize temporal trends while preserving temporal

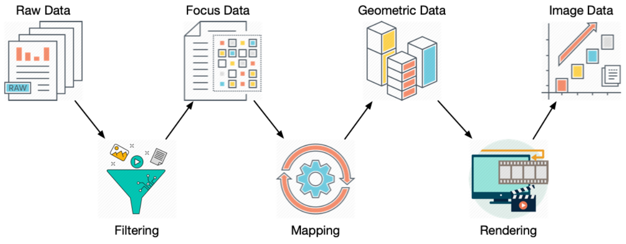

Fig. 1. Visualization Pipeline

coherence. Collins et al. [9] introduce Parallel TagClouds that combine parallel coordinates and traditional word clouds, where the words of each document are distributed along each coordinate axis. Lee et al. [10] present the Sparkclouds that visualize trends between multiple word clouds by integrating spark lines into a word cloud. Both methods perform well in the visualization of trends, with the Sparkclouds being the better one in terms of scalability. Cui et al. [11] combine a trend chart and multiple word clouds together to illustrate temporal changes of the underlying data. By combining multi-dimensional scaling and force-directed layout, this method can create semantic and stable word clouds over time. Recently, Chi et al. [12] propose a morphable word clouds, where a sequence of spatial shapes is specified as a boundary for a set of time-varying word clouds. By using rigid body dynamics, they arrange words within the given shape sequence so that temporal changes are encoded by both the shapes and the content of the word clouds. In this paper, we proposed an embedded temporal visualization method to present the temporal patterns in the text.

III. Proposed Technique

A. Motivation

Models of bibliographic data need to consider many kinds of information. Articles are usually accompanied by metadata such as authors, publication data, categories and time[13]. Bibliographic analysis considers the author's research areas, the citation network and the paper content among other things, which can be combined into a topic model that produces a bibliographic model of authors, topics and documents. In this paper, We propose a novel and efficient inference visualization algorithm to explore the constantly change of the hot topics.

B. System Overview

To facilitate generation of visual output at all three levels, a flexible mapping strategy is required. Such a strategy has been manifested as the so-called visualization pipeline. The visualization pipeline leveraged in this paper is proposed by Santos and Brodlie[14], which consists of the three steps: filtering, mapping and rendering (see Figure 1).

The filtering step prepares the raw input data for processing through the remaining steps of the pipeline. This is done with respect to the given analysis task and includes not only selection of relevant data but also operations for data enrichment or data reduction, interpolation, data cleansing, grouping, dimension reduction, and others.

Literally, the mapping step maps the prepared data to appropriate visual variables. This is the most crucial step as it largely influences the expressiveness and effectiveness of the resulting visual representation.

Finally, the rendering step generates actual images from the previously computed geometry and visual attributes. This general pipeline model is the basis for many visualization systems.

IV. Temporal Display Design

The bibliography raw data usually is collected in plain text files, and organized by a standard rule, such as BibTex, EndNote etc. In this paper, we use the IEEE VIS conference series data[15] from 1990 to 2016 to illustrate the usage of our method. The data used in our experiments contains IEEE VIS publications, i.e., InfoVis, SciVis, VAST, or Vis. Each record consists of 11 fields, including conference type, publication year, paper title, DOI, abstract, author names, and references (inside this dataset only). In the purpose of exploring research topic, only some fields are involved in the proposed method, which contains the topic features and temporal features. Specifically, the topic features will be extracted from these fields, which are paper title, abstract, and full text if available. The temporal features will be aggregated by the time granularity, which is the 'year' data field in this case.

A. Data Processing

The raw data contains topic features and temporal features will be filtered into the focus data by the data processing approach. The raw data is composed by records, which could be distinct from each other by some specific notations or punctuation, such as a semicolon or a line break. Let's define the raw data set is $X := \{X^{(1)}, X^{(2)}, ..., X^{(n)}\}$, where the n is the number of records contained in the raw data, and the $X^{(i)}$ means the ith record in X with $1 \leq i \leq n$.

To show the temporal visualization of the bibliography data set, both the words variable and the times variable should be considered in the data processing approach. To achieve a better visualization presentation with clarity and concision, the method proposed in this paper gives a general pipeline to declare the details of the data processing.

An ignored word list and a merged word list should be initialized before the data processing approach, and will be continuous updated during the data exploring activity.

The ignored word list contains frequently used words (such as and, the, because, and so on) or words that you do not want indexed, which will be excluded from the result. The data processing program comes with a default set of ignored words that you can modify as needed. The ignored word list could be defined as $P := \{P_1, P_2, ..., P_m\}$, where P_i with $1 \leq i \leq m$ means a single word will be ignored.

The merged word list is composed by several word groups, each group contains the words which is a synonym for each other. The merged word list is defined as $M := \{M^{(1)}, M^{(2)}, ..., M^{(n)}\}$, where $M^{(i)}$ with $1 \leq i \leq n$ means the ith merged word group. The merged word group is an order-sensitive sequence, which is defined as $M^{(i)} := \{M_1^{(i)}, M_2^{(i)}, ..., M_m^{(i)}\}$, and all the words in the same merged word group will be merged into the first word in the sequence. The merged word list can be initialized by several approaches. One of the most leveraged method is the word distance algorithm, such as the Levenshtein algorithm, which is a string metric for measuring difference between two sequences. Informally, the Levenshtein distance between two words is the minimum number of single-character edits (i.e. insertions, deletions or substitutions) required to change one word into the other. But the word distance algorithm should be applied under supervision. For example, the words "universe" and "university" are quit similar but with very different meaning.

Given the ignored word list and the merged word list, the raw data processing program could be leveraged to produce the focus data set, which is defined as $R := \{R_1, R_2, ..., R_n\}$, where $R_i := \{word, weight, seq\}$ and $seq := [\{t_1, v_1\}, ..., \{t_n, v_n\}]$. Considering the visual dimension, the focus data set only contains the n records of the whole data set, following the rule of sorting the records in descending order by the weight variable of each record. The raw data processing program is briefly described in a pseudo code block, as is shown in algorithm 1.

Algorithm 1 Data Processing Algorithm

```
Input
    X    raw data set
    P    ignored word list
    M    merged word list
    n    return the top $n$ records
Output
    R    $R := \{R_1, R_2, ..., R_n\}$
                        ▷ where $R_i := \{word, weight, seq\}$
                        ▷ $seq := [\{t_1, v_1\}, ..., \{t_n, v_n\}]$
procedure DataProcess($X, P, M, n$)
    $t_{min} \leftarrow$ MinYear($X$)
    $t_{max} \leftarrow$ MaxYear($X$)
    $ry \leftarrow t_{max} - t_{min} + 1$
                        ▷ ry: the time range in years
    $R = \emptyset$
    for $\forall X^{(i)} \in X$ do
        for $\forall w, t \in X^{(i)}$ do
                        ▷ w is the word in text mode
                        ▷ t is the time variable in year
            if $w \in P$ then
                continue
            end if
            for $\forall M^{(i)} \in M$ do
                if $w \in M^{(i)}$ then
                    $w \leftarrow M_0^{(i)}$
                        ▷ $M_0^{(i)}$ is the replacement in the group $M^{(i)}$
                    $found \leftarrow false$
                    for $\forall R_i \in R$ do
                        if $\exists R_i[word] = w$ then
                            $found \leftarrow true$
                            $R_i[weight] \leftarrow R_i[weight] + 1$
                            $j \leftarrow t - t_{min}$
                            $R_i[seq][j] \leftarrow R_i[seq][j] + 1$
                        end if
                    end for
                    if $notfound$ then
                        $word \leftarrow w$
                        $weight \leftarrow 1$
                        $seq \leftarrow$ Array($ry$)
                        $seq[t - t_{min}] \leftarrow \{t, 1\}$
                        $R \leftarrow R \cup \{word, weight, seq\}$
                    end if
                end if
            end for
        end for
    end for
    Sort(R)
                        ▷ Sort R in descending order by weight
    Return $R[1, ..., n]$
                        ▷ Only the top $n$ records are needed
end procedure
```

B. Data Mapping

The mapping from focus data to geometric data is the core of our method. Compared to traditional word cloud visualization algorithm, e.g. the Wordle layout algorithm, our method has considered the relevant temporal patterns additionally.

The word layout algorithm is revealed in the pseudo code as shown in algorithm 2. The whole process of the algorithm is similar to the Wordle layout algorithm, but with one bit improvement, which is the placement of the underline pattern bar under the word area before the detection of the intersections. For each word in the focus data, the algorithm gives the dimension of the word and the relevance font size when there is no intersections with any previously placed words.

The temporal pattern related to each word is encoded into the underline pattern bar, which is split joint by a rectangle sequence filled with a correlated color. There is a one-to-one relationship between the temporal pattern and the rectangle sequence, and the numeric value in the temporal pattern is encoded to a color.

C. Visualization Result

To demonstrate the proposed visualization method in this paper, an evaluation experiment is developed base on the IEEE VIS conference series data (as mentioned in section IV). The experiment is implemented by JavaScript program language, and exhibited in the chrome browser. The data mapping part of the algorithm leverages the "Canvas" technique to perform the

Algorithm 2 Word Layout Algorithm

```
Input
    R    $R := \{R_1, R_2, ..., R_n\}$
                                        $\triangleright\ R_i := \{word, weight, seq\}$
                                        $\triangleright\ seq := [\{t_1, v_1\}, ..., \{t_n, v_n\}]$
Output
    R    $R := \{R_1, R_2, ..., R_n\}$
                                        $\triangleright\ R_i := \{word, weight, seq, size, d, colors\}$
                                        $\triangleright\ d := \{x, y, w, h\}$
procedure WORDLAYOUT($R$)
    for $\forall R_i \in R$ do
        $R_i[size] \leftarrow$ FONTSIZESCALE($R_i[weight]$)
        $R_i[colors] \leftarrow$ COLORSCALE($R_i[seq]$)
        repeat
            MOVEALONGSPIRALPATH($R_i[word]$)
            PLACEMENTSTRATEGY($R_i[word]$)
            PLACEUNDERLINEPATTERNBAR($R_i[word]$)
        until no intersection detected
      return $R$
    end for
end procedure
```

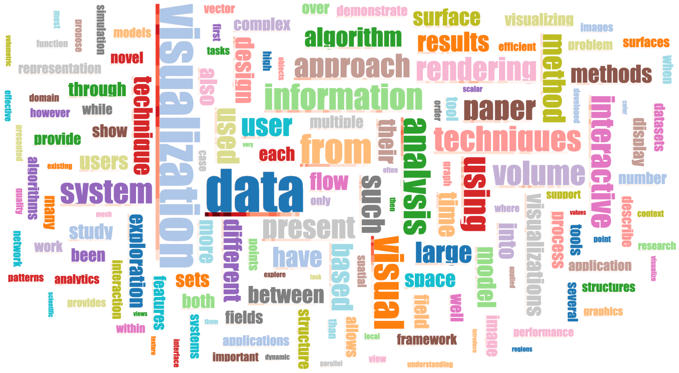

Fig. 2. Temporal Word Cloud

intersection detection during the word layout phase. The final visualization effect is achieved by the ”SVG” specification provided by a third-party library named D^3[16]. A final visual snapshot of the system is shown in the figure 2. By utilizing the system, a researcher could have a good understanding what aspects to consider when selecting a topic for the research paper.

V. Conclusion

In this paper, we proposed an embedded visualization method to facilitate the understanding of temporal content evolution in a set of bibliography documents by visually illustrating the hot topic evolution. A system is developed and demonstrated to evaluate the method, which given a very positive feedback. But the current version of the system is lacking of an interactive user interface, which causes a bad performance in parameter adjustment. In the future work, an interactive control panel will be mounted on the interface, which will improve the user experience.

References

[1] F. B. Viégas and M. Wattenberg, “TIMELINESTag clouds and the case for vernacular visualization,” interactions, vol. 15, no. 4, p. 49, Jul. 2008.

[2] A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen, “Getting our head in the clouds,” presented at the the SIGCHI Conference, New York, New York, USA, 2007, p. 995.

[3] H. Locarek-Junge and C. Weihs, Eds., Classification as a Tool for Research. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[4] O. Kaser and D. Lemire, “Tag-cloud drawing: Algorithms for cloud visualization,” arXiv preprint cs/0703109, 2007.

[5] C. Seifert, B. Kump, W. Kienreich, G. Granitzer, and M. Granitzer, “On the Beauty and Usability of Tag Clouds,” presented at the 2008 12th International Conference Information Visualisation (IV), pp. 17–25.

[6] H. Strobelt, M. Spicker, A. Stoffel, D. Keim, and O. Deussen, “Rolled-out Wordles: A Heuristic Method for Overlap Removal of 2D Data Representatives,” Computer Graphics Forum, vol. 31, no. 3, pp. 1135–1144, Jun. 2012.

[7] F. B. Viegas, M. Wattenberg, and J. Feinberg, “Participatory Visualization with Wordle,” IEEE Transactions on Visualization and Computer Graphics, vol. 15, no. 6, pp. 1137–1144.

[8] Y. Wang, X. Chu, C. Bao, L. Zhu, O. Deussen, B. Chen, and M. Sedlmair, “EdWordle - Consistency-Preserving Word Cloud Editing.,” IEEE Trans. Vis. Comput. Graph., vol. 24, no. 1, pp. 647–656, 2018.

[9] C. Collins, F. B. Viégas, and M. Wattenberg, “Parallel Tag Clouds to explore and analyze faceted text corpora,” presented at the 2009 IEEE Symposium on Visual Analytics Science and Technology, pp. 91–98.

[10] Bongshin Lee, N. H. Riche, A. K. Karlson, and S. Carpendale, “SparkClouds: Visualizing Trends in Tag Clouds,” IEEE Transactions on Visualization and Computer Graphics, vol. 16, no. 6, pp. 1182–1189.

[11] W. Cui, Y. Wu, S. Liu, F. Wei, M. X. Zhou, and H. Qu, “Context preserving dynamic word cloud visualization,” presented at the 2010 IEEE Pacific Visualization Symposium (PacificVis), pp. 121–128.

[12] M.-T. Chi, S.-S. Lin, S.-Y. Chen, C.-H. Lin, and T.-Y. Lee, “Morphable Word Clouds for Time-Varying Text Data Visualization,” IEEE Transactions on Visualization and Computer Graphics, vol. 21, no. 12, pp. 1415–1426.

[13] K. W. Lim and W. Buntine, “Bibliographic analysis on research publications using authors, categorical labels and the citation network,” Machine Learning, vol. 103, no. 2, pp. 185–213, May 2016.

[14] S. dos Santos and K. Brodlie, “Gaining understanding of multivariate and multidimensional data through visualization,” Computers & Graphics, vol. 28, no. 3, pp. 311–325, Jun. 2004.

[15] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. D. Stolper, M. Sedlmair, J. Chen, T. Moller, and J. Stasko, “Vispubdata.org: A Metadata Collection About IEEE Visualization (VIS) Publications,” IEEE Transactions on Visualization and Computer Graphics, vol. 23, no. 9, pp. 2199-2206, Oct. 2016.

[16] M. Bostock, V. Ogievetsky, and J. Heer, “D^3 Data-Driven Documents,” IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 12, pp. 2301–2309.