T-Scroll: Visualizing Trends in a Time-series of Documents for Interactive User Exploration

Yoshiharu Ishikawa[1,2]
Mikine Hasegawa[3] *
[1]Information Technology Center, Nagoya University
[2]Nagoya University Library Study
[3]Department of Information Engineering, School of Engineering, Nagoya University
ishikawa@itc.nagoya-u.ac.jp

Abstract

On the Internet, a large number of documents such as news articles and online journals are delivered everyday. We often have to review major topics and topic transitions from a large time-series of documents, but it requires much time and effort to browse and analyze the target documents. We have therefore developed an information visualization system called *T-Scroll* (Trend/Topic-Scroll) to visualize the transition of topics extracted from those documents. The system takes periodical outputs of the underlying clustering system for a time-series of documents then visualizes the relationships between clusters as a scroll. Using its interaction facility, users can grasp the topic transitions and the details of topics for the target time period. This paper describes the idea, the functions, the implementation, and the evaluation of the T-Scroll system.

1 Introduction

Due to the evolution of information services on the Internet, various kinds of documents, such as news articles and online journals, are delivered everyday. Since a large amount of textual information is obtained continually, research aimed at summarizing huge text information and detecting trends becomes an important issue today [1, 2]. Although we can reduce the efforts of users by using clustering and information extraction techniques, users still have to make the effort to capture overall trends from the documents. For this purpose, the user needs an intuitive tool to see which kind of major topics appear and how topics change as time passes.

Based on this background, we have developed an information visualization interface called *T-Scroll* (Topic/Trend-Scroll) to visualize the overall trend of a time-series of documents based on their contents and timestamps. T-Scroll is constructed over a document clustering system and visualizes periodical clustering results. It organizes the clustering result for each time period along the time axis and displays links between clusters. Links are generated to represent related clusters and the system presents the topic flow in a scroll-like style. The user can browse the T-Scroll interface using Web browsers and can select and explore more detailed information if they need it.

The organization of the paper is as follows. Section 2 introduces related work. Section 3 describes the novelty-based clustering method for a time-series of documents, which is the basis of the T-Scroll system. Section 4 presents the features and functions of T-Scroll. Section 5 describes the implementation techniques then Section 6 shows the evaluation results. Finally, Section 7 concludes the paper and indicates future work.

*Current affiliation: Nihon Seifun Co. Ltd.

2 Related Work

2.1 Visualization of a time-series of documents

Müller et al. [3] provides a short survey of visualization techniques for time dependent data. There are few proposals of the visualization of a time-series of documents except for the following two systems.

ThemeRiver [4] is an information visualization system which visualizes topic streams like a *river*. The displayed image resembles a river that flows from left to right along the time axis. The river contains several streams in different colors and they correspond to the selected topics (themes). For each topic stream, phrases are displayed on the screen to help users' interpretation. The width of a stream changes depending on time and reflects the number of documents for each time period. ThemeRiver shares similar ideas with T-Scroll since they utilize a scroll-based interface, but it does not use clustering. ThemeRiver is a system that focuses on providing visual impact and cannot represent topic transitions. Although it may be useful to see an overview of the trend, the system is not a powerful tool for analyzing and browsing a time-series of documents. In contrast, T-Scroll provides facilities so that users can view document titles and contents if they need.

TimeMine [5] is a system that extracts topics from a time-series of documents then displays *timelines* to represent topics on the screen. It analyzes a time-series of documents over the specified time period using a statistics-based method and extracts topics which are represented by groups of documents. Based on the analysis, the system displays rectangular regions representing timelines on the screen in which time flows from left to right. In addition, the system displays keywords along the corresponding timelines. The main focus of TimeMine is to select major topics and their time periods. Although the proposed techniques are quite interesting, the system does not provide functionalities for more detailed analysis.

2.2 Analysis of time-dependent clusters

There are some proposals for tracking and analyzing clusters changing in time, but they do not aim for visualization. Mei and Zhai [6] propose a statistical approach for discovering major topics from a time-series of documents. In this scheme, a theme is represented as a probability distribution over a time period and can be seen as a cluster. Relationships between consecutive time instants are determined based on probabilistic criteria. The derived theme transition graph resembles the graph generated by the cluster relationships of T-Scroll. In [6], they also provide a global method for analyzing the whole graph to mine meaningful patterns.

MONIC [7] proposes an approach for detecting various types of patterns from cluster transitions such as the splitting and merging of clusters, cluster size change, etc. MONIC discovers events based on historical snapshots of clusters. Its underlying idea is related with our approach.

3 Novelty-based Clustering Method for a Time-series of Documents

T-Scroll is based on the *novelty-based document clustering method* [8, 9, 10]. The target of the method is a *time-series of documents* such as news articles and online journals. Such documents have the general property that additional documents with new timestamps are continually delivered over the network.

The clustering method focuses on the clustering of a time-series of documents and has the following features:

1. To calculate similarity, it considers not only document contents but also the *novelty* of each document. It incorporates a similarity function that considers the novelty of documents then puts high weights on recent documents.

2. When a new document is delivered, clustering should be performed to acquire the new clustering result. To alleviate the processing cost, the method uses incremental processing as much as possible.

3. Since the method puts high weights on novel documents, old documents tend to have low effect on the clustering result and become outliers. Therefore, old documents are deleted from the clustering targets automatically so we can reduce the processing cost.

Based on this approach, the method clusters a time-series of documents in an online manner and provides clustering results focusing on current major topics.

We now introduce the similarity function used in the clustering method. In a time-series of documents, such as news articles and online journals, the value of a document generally decreases over time. The novelty-based clustering method for a time-series of documents [8, 9, 10] proposes the *document forgetting model* and derives the document similarity based on that.

The forgetting model assumes that the importance (weight) of a document declines in an exponential manner as time passes, and defines the weight of document d_i as follows:

$$dw_i = \lambda^{\tau - T_i} \qquad (0 < \lambda < 1), \tag{1}$$

where τ is the current time and T_i is the timestamp of d_i. The parameter λ represents how fast the weight declines. The model inherits the idea from *aging* or *obsolescence* in library information science and infometrics [11]. Now we define the total weight of a document set with n documents $d_1, \ldots, d_n$ as $tdw = \sum_{l=1}^{n} dw_l$ and define the occurrence probability of d_i within a document set as a subjective probability $\Pr(d_i) = dw_i / tdw$. Since old documents have small probabilities, this represents the idea of forgetting old documents.

Document similarity is defined based on a probabilistic approach [8, 9, 10]. Its general form is given by

$$sim(d_i, d_j) = \Pr(d_i)\Pr(d_j)\frac{\boldsymbol{d}_i \cdot \boldsymbol{d}_j}{len_i \times len_j}, \tag{2}$$

where "·" is the inner product of document vectors and len_i is the vector length of $\boldsymbol{d}_i$. Thus, document similarity considers not only how documents are similar, but also whether two documents are old or not. Very old documents tend to be dissimilar to other documents and become outliers. By using the similarity in the clustering procedure, we can achieve a novelty-based clustering that has a bias toward recent documents.

The clustering method actually used in [9, 10] is an extended version of the k-means method [12]. When new documents are obtained, we need to perform a new clustering to reflect them. Since clustering from scratch is quite costly, our approach utilizes k cluster representatives from the previous clustering result as initial cluster representatives. Based on this approach, the clustering procedure converges faster than the naive approach. Moreover, it can improve the clustering quality [10].

As described above, the novelty-based clustering method periodically performs incremental clustering for continually delivered documents then outputs the clustering results. Each clustering result represents major topics for the period when the clustering was performed. By storing such clusters permanently, we can use them for analyses to be performed later. T-Scroll is based on such an idea and can be used as a visual interface for analyzing retrospective document collections.

4 Overview of the T-Scroll System

4.1 System features

The main features of the *T-Scroll* system are summarized as follows:

1. It displays the clustering result for each time period along the time axis with topic labels so the user can grasp overall topics for the target time interval.

2. The user can select the cluster in which he or she is interested in, then the user can obtain more detailed information such as the keyword list or can refer to the original articles in an interactive way.

3. For clusters obtained in a period, it creates *links* from the clusters of the previous time period based on the cluster similarity; the user can observe the relationships between clusters.

4. The user can select an appropriate interval to visualize clusters on the screen then the user can perform analysis depending on his or her requirement with different veles of detail. The approach corresponds to *roll-up* and *drill-down* facilities in *OLAP* (*On-Line Analytical Processing*) [12].

Based on these features, the flow of topics and trends are represented as a scroll and we call the system *T-Scroll*.

4.2 System functionalities

Figure 1 shows a screenshot of T-Scroll. The figure represents news articles from the TDT2 Corpus [2]. The corpus contains news broadcasts on TV and radio in 1998. On the interface, the time flows from left to right. Using the slide bar on the screen, we can move to the previous time period. Ellipses shown on the same vertical line are clusters obtained in the same clustering process. In this figure, $k = 20$ clusters are generated for each time period. The interval between two consecutive clusterings is set to one day.

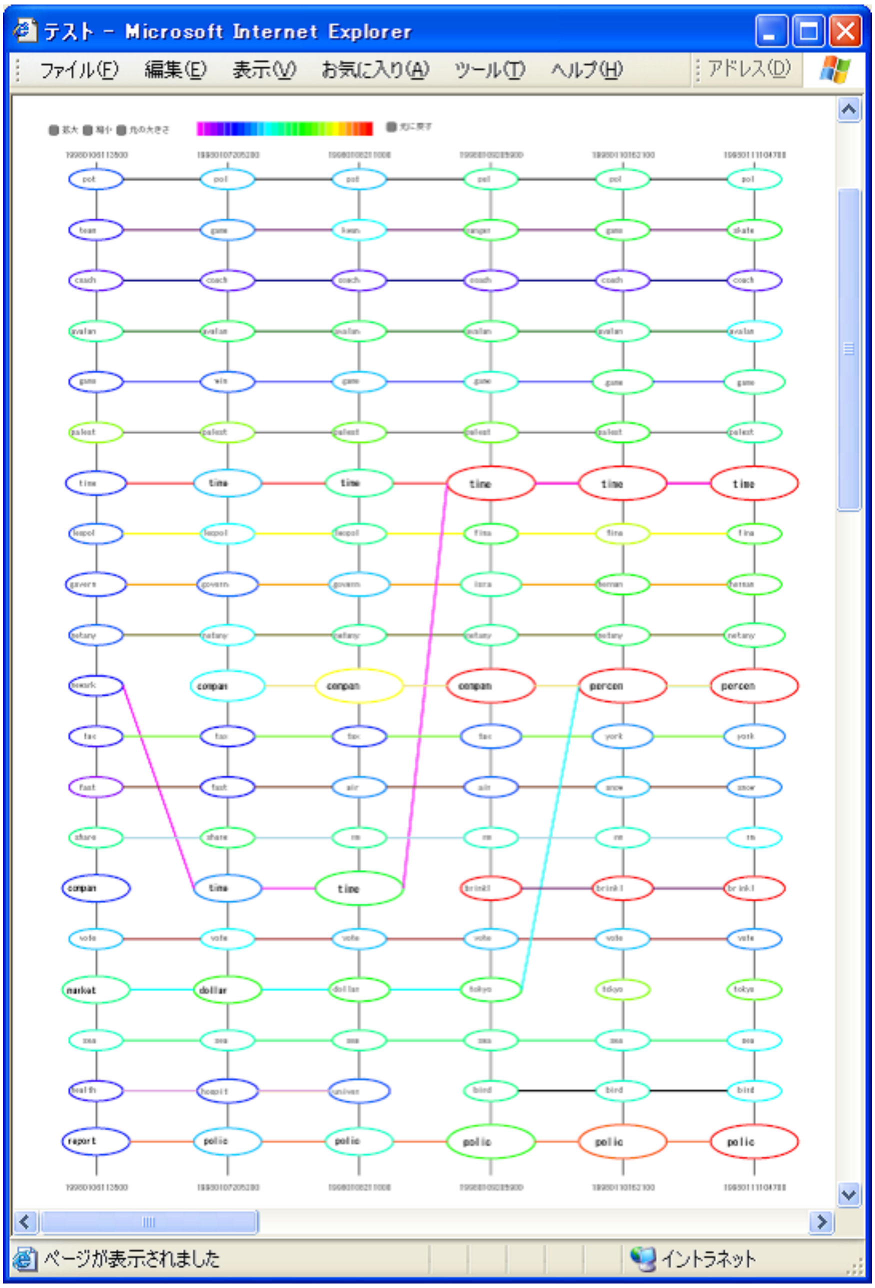

テスト - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
アドレス(D)
pol pol pol pol pol pol
team game team ranger game skate
coach coach coach coach coach coach
avalan avalan avalan avalan avalan avalan
game win game game game game
palest palest palest palest palest palest
time time time time time time
leopol leopol leopol fina fina fina
govern govern govern isra herman herman
netany netany netany netany netany netany
newark compan compan compan percen percen
tax tax tax tax york york
fast fast air air snow snow
share share rm rm rm rn
compan time time brinkl brinkl brinkl
vote vote vote vote vote vote
market dollar dollar tokyo tokyo tokyo
sea sea sea sea sea sea
health hospit univer bird bird bird
report polic polic polic polic polic
ページが表示されました
イントラネット

Figure 1: Screenshot of T-Scroll (one day basis)

The vertical order of clusters is not very meaningful: clusters are displayed in order of their increasing cluster ID. However, the novelty-based clustering method [10] is able to generate the “regular” graph structure as shown in the figure. When a new clustering is performed, the method reuses the previous cluster representatives then performs a clustering process based on the k-means method. Therefore, the previous cluster IDs are retained for the new clustering result. Such situations occur quite often, especially when we utilize a short period for the display interval such as "one day" for Fig. 1.

4.2.1 Cluster labels

For each cluster, we select a feature term that has the highest score within the terms contained in its documents as a *cluster label*. After trials of several scoring methods, we have decided to assign the score for term t_j in cluster C_p as

$$score(t_j) = \sum_{d_i \in C_p} \Pr(d_i) \cdot tf_{ij}. \tag{3}$$

Namely, for each document d_i in the cluster, the term frequency tf_{ij} for term t_j is multiplied with the document weight $\Pr(d_i)$ then the summation is taken. Although we have tried to display multiple terms on a cluster, our impression was that it is too complicated. So we select only one term in the current implementation.

4.2.2 Cluster sizes

The area of a cluster ellipse corresponds to the number of documents in the cluster. To represent the cluster size, we select an appropriate size from several size levels. Thus, the user can be made aware of topic sizes.

4.2.3 Cluster links

As shown in Fig. 1, links are generated between selected clusters. Each link means that those clusters are related. The cluster relationship score is defined as a probability:

$$score(C_i \rightarrow C_j) = \Pr(C_j | C_i) = \frac{|C_i \cap C_j|}{|C_i|}. \tag{4}$$

The formula measures the degree to which the documents in cluster C_i are contained within cluster C_j. We allow zero or multiple links to emerge from one cluster so that the system can represent topic expiration (represented by zero link) and topic separation (represented by multiple links).

4.2.4 Cluster qualities

To help the user to capture the quality of a cluster, T-Scroll visualizes cluster quality using different colors for the cluster border lines. Red represents the highest cluster quality and purple means the lowest cluster quality. The quality score of cluster C is defined as follows [10]:

$$quality(C) = |C| \cdot avg\_sim(C), \tag{5}$$

where $|C|$ means the number of documents in C, and $avg\_sim(C)$ represents the average similarity of all the document pairs in the cluster which is defined as follows:

$$avg\_sim(C) = \frac{1}{|C|(|C|-1)} \sum_{d_i, d_j \in C, d_i \neq d_j} sim(d_i, d_j). \tag{6}$$

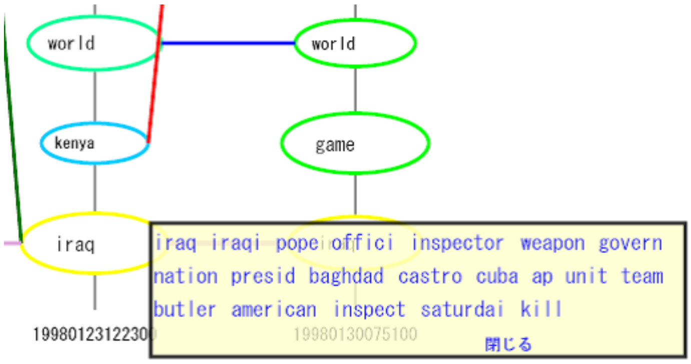
Note that $quality(C)$ takes a large score not only when the cluster size is large but also documents in a cluster are similar each other.

4.2.5 Drill-down/roll-up functions

T-Scroll supports drill-down and roll-up functions. For example, Fig. 1 is an example with a one day basis, but we can utilize more coarse values (e.g., three days, one week). In the actual implementation, we periodically perform clustering then determine whether to create links between clusters based on Eq. (4). Then we store the result as a graph structure. When a visualization request is issued to T-Scroll, the system extracts the required subgraph from the stored graph.

4.2.6 Displaying a keyword list

As shown in Fig. 1, T-Scroll displays one keyword as a label for each cluster, but the user often needs more keywords to understand the cluster content. Therefore, T-Scroll also provides a facility for seeing the cluster contents. When the user moves the mouse cursor over a cluster ellipse, the system displays a keyword list for the cluster on the screen as shown in Fig. 2. The figure shows the situation when we move the mouse cursor over the "iraq" cluster. The system displays top-ranked keywords for the cluster.

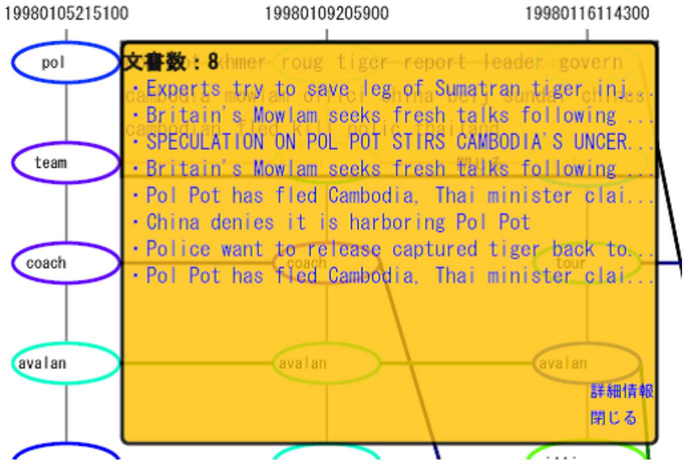


iraq iraqi pope offici inspector weapon govern nation presid baghdad castro cuba ap unit team butler american inspect saturdai kill

Figure 2: Displaying keyword list

4.2.7 Access to original documents

Although we can see the overall contents of clusters using the above functions, it is difficult to know which documents are contained in the cluster. Therefore, the system provides a facility for displaying documents in a cluster by clicking the mouse on the cluster ellipse. Figure 3 shows this situation. The system displays the titles of recent documents in the cluster. In addition, if the user clicks on a document title, its content appears on the display (the figure is omitted here).



文書数：8

・Experts try to save leg of Sumatran tiger inj...
・Britain's Mowlam seeks fresh talks following ...
・SPECULATION ON POL POT STIRS CAMBODIA'S UNCER...
・Britain's Mowlam seeks fresh talks following ...
・Pol Pot has fled Cambodia, Thai minister clai...
・China denies it is harboring Pol Pot
・Police want to release captured tiger back to...
・Pol Pot has fled Cambodia, Thai minister clai...

Figure 3: Displaying document titles

4.2.8 Keyword-based emphasis

We can perform keyword-based queries by entering keywords in the keyword query field on the T-Scroll interface. The system emphasizes the clusters if their top-20 keywords contain the given keyword. Figure 4 shows the situtaion. The figure shows the result when we provide the keyword "olympic" (a Winter

Olympic was held in the display period). The emphasized clusters are matched ones. In addition, the system can receive multiple keywords. In this case, if either of the given keywords is contained in the keyword list, the cluster is matched. Based on this functionality, the user can see topic transition more easily.

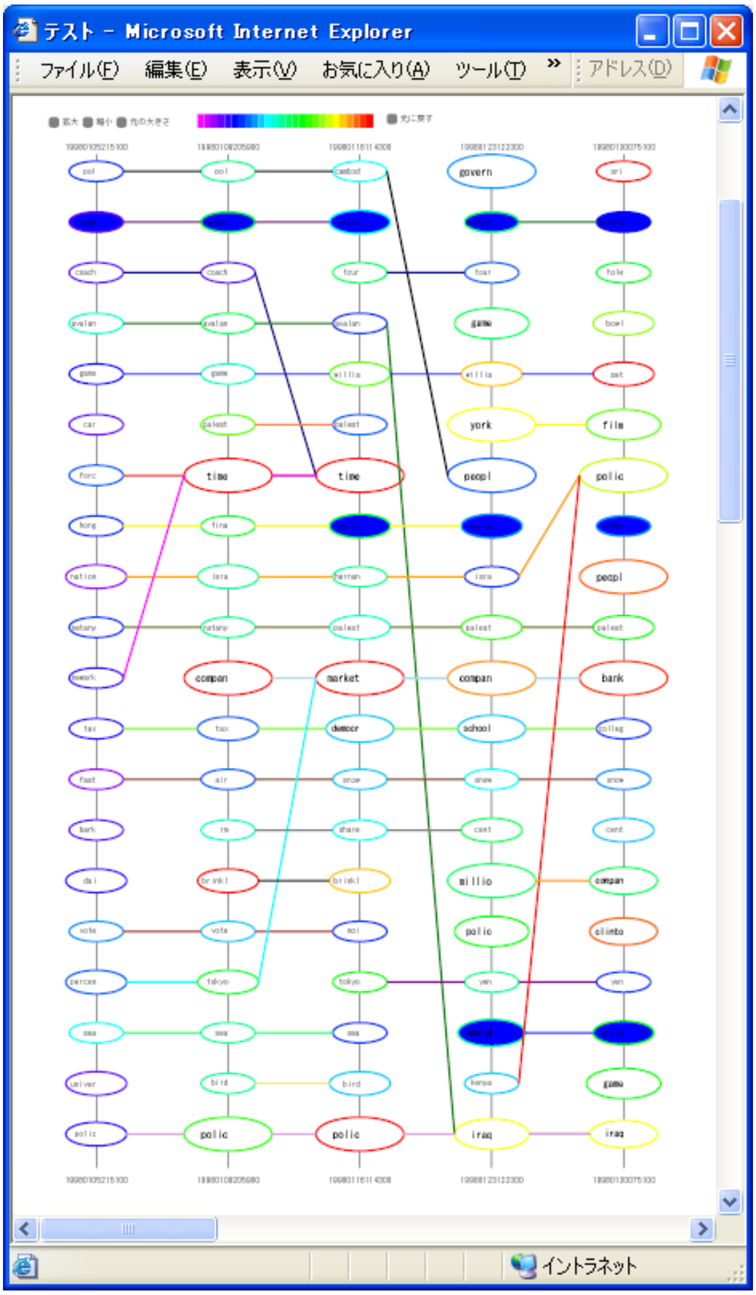


Figure 4: Emphasized cluster display based on keywords

5 System Implementation

This section provides an overview of the implementation of the system. Figure 5 shows the organization of the T-Scroll system. We describe each system component. A solid line represents a data flow and a dotted line means a control flow (procedure call).

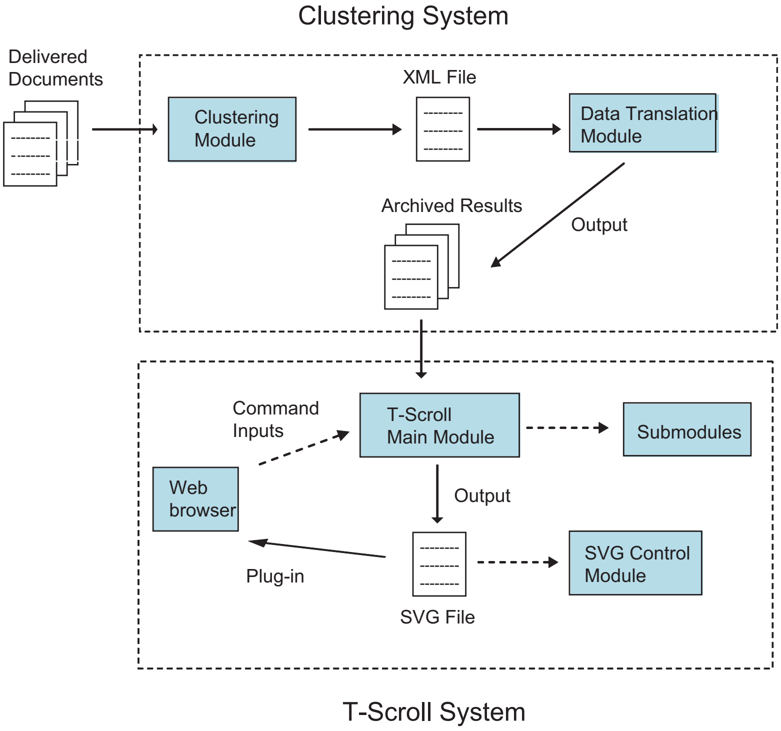


Figure 5: System organization of T-Scroll system

The system cooperates with the novelty-based document clustering system [9, 10] and uses its outputs. A new clustering result is obtained by giving a newly acquired document set to the clustering program. The clustering result is output as an XML file.

The T-Scroll system reads the XML files output from the clustering system. Selected XML files are incorporated and used according to the period and the parameters specified by the user. The main module of T-Scroll is written in JavaScript and runs within a Web browser. Some part of processing concerning the user interface is implemented by JavaScript and AJAX.

Given the target period and other parameters, T-Scroll displays the interface on a Web browser. For this purpose, a submodule written in Perl is called from the main module. This submodule analyzes the XML files and generates an SVG file to be displayed on the interface. The SVG file is read in the Web browser then the interface appears as shown in Fig. 1. JavaScript codes are embedded in the SVG file and modules written in Perl are called on as necessary.

6 System Evaluation

This section summarizes the results of the evaluation experiments. The evaluation experiments were conducted by 10 users. The data set used was articles collected from Japanese news web sites from September 2006 to February 2007. On average, 100 articles were collected and clustering was performed at six-hour intervals.

6.0.9 Overall impressions

First, we asked for a general impression of the T-scroll user interface. The scores were selected from five levels: 1 (very bad) to 5 (very good). Four evaluation categories (usability, understandability, usefulness, and design) were used.

Figure 6 shows the evaluation result. The averaged scores with standard errors are plotted. Although the usefulness score is 3.7 on average, the usability score is 2.5, which means that further system improvement is necessary. Scores for understandability and design are 3.1 and 2.8, respectively, but the design score has large variance. We conducted user interviews and collected valuable comments. The major problems are 1) the response time of the system and 2) the label for a cluster is not necessarily an intuitive one. The first problem can be solved by revised implementation. The second problem is more important and we will consider it later.

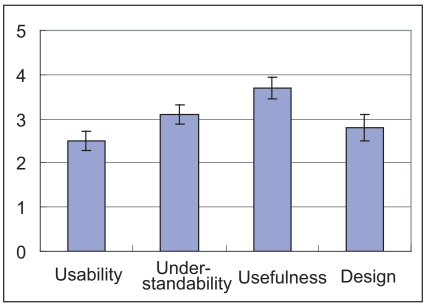


Figure 6: Overall evaluation scores

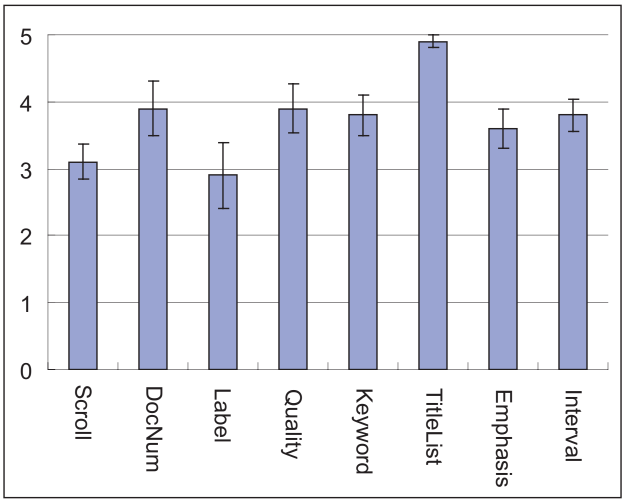


Figure 7: Evaluation score for each function

6.0.10 Evaluation of each function

The evaluation results for the individual system functions are shown in Fig. 7. The evaluation criteria are as follows:

- Scroll: The scroll-like visualization approach
- DocNum: The appropriateness of associating the number of documents of a cluster with the size of its corresponding ellipsoid
- Label: The label display facility
- Quality: The method of showing the quality of clusters by colors
- Keyword: The keyword list display function (Fig. 2)
- TitleList: The function for displaying document titles (Fig. 3)
- Emphasis: The keyword-based emphasis function (Figure 4)
- Interval: The function for allowing multiple interval settings

All the average scores are over three except for the "Label" facility. The reasons for the "Label" problem are as follows: 1) We have used Japanese morphological analysis tool to extract feature terms (e.g., noun terms). Since we did not tune the dictionary of the tool for the experiment, the quality of the extracted terms is not satisfactory. Expansion of dictionary would improve the quality of terms. 2) The automatic label selection method based on Eq. (3) does not necessarily select comprehensible terms. It may be better to use a controlled vocabulary for labels. Next, we consider the "Score" criteria. One of the reasons that it has relatively low score 3.1 would be that there is no candidate to compare with T-Scroll in this experiment. Finally, consider the "TitleList" facility. The experimental result says displaying document titles is quite useful for the users.

6.0.11 Observability of topics

We conducted another experiment to see whether the users could observe the major five topics in Japan in November 2006. The five topics include big events and accidents such as "damage by big tornado on November 7th", but their details are omitted. Figure 8 shows the evaluation result by ten users. We have asked the users whether they can actually observe each topic. We can say that all the average scores are successful. The reason for the relatively low score on Topic 3 is that the topic was a big issue for this period and it contains subtopics. Since subtopics have appeared in some periods as multiple topic threads, the users might find it difficult to make a judgment.

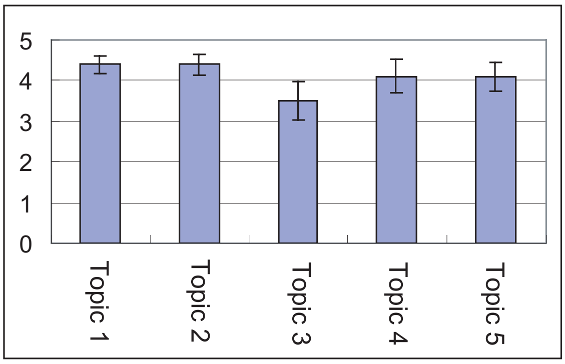


Figure 8: Observability of topics

7 Conclusions and Future Work

In this paper, we have described the idea, the functions, the implementation, and the evaluation of the T-Scroll system, a visual interface for analyzing the transition of topics from a time-series of documents. This system is based on the novelty-based clustering method for time-series of documents and uses the clustering results for the visualization. T-Scroll supports interactive processing facilities and has several functions such as keyword list display, document title display, and so on.

Based on evaluation by users, it was shown that they can observe major topics that actually happened using the system. We can say that the objective of capturing the trends in a time-series of documents is achieved by the system. However, further improvement of the system is necessary. As shown in the evaluation by the users, we should improve the usefulness and understandability of the interface.

Acknowledgments

This research is partly supported by a Grant-in-Aid for Scientific Research (19300027) from the Japan Society for the Promotion of Science (JSPS), Japan. In addition, this research was supported by grants from the Hoso Bunka Foundation.

References

[1] Kontostathis, A., Galitsky, L.M., Pottenger, W.M., Roy, S., Phelps, D.J.: A survey of emerging trend detection in textual data mining. In Berry, M.W., ed.: Survey of Text Mining: Clustering, Classification, and Retrieval. Springer-Verlag (2003) 185–224

[2] Allan, J., ed.: Topic Detection and Tracking: Event-based Information Organization. Kluwer (2002)

[3] Müller, W., Schumann, H.: Visualization methods for time-dependent data: An overview. In: Proc. of 2003 Winter Simulation Conf. (2003) 737–745

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

[5] Swan, R., Allan, J.: Automatic generation of overview timelines. In: Proc. of ACM SIGIR. (2000) 49–56

[6] Mei, Q., Zhai, C.: Discovering evolutionary theme patterns from text: An exploration of temporal text mining. In: Proc. of ACM KDD. (2005) 198–207

[7] Spiliopoulou, M., Ntoutsi, I., Theodoridis, Y., Schult, R.: MONIC: Modeling and monitoring cluster transitions. In: Proc. of ACM KDD. (2006) 706–711

[8] Ishikawa, Y., Chen, Y., Kitagawa, H.: An on-line document clustering method based on forgetting factors. In: Proc. of the 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2001). Volume 2163 of LNCS. (2001) 332–339

[9] Khy, S., Ishikawa, Y., Kitagawa, H.: Novelty-based incremental document clustering for on-line documents. In: Proc. of International Workshop on Challenges in Web Information Retrieval and Integration (WIRI 2006). (2006)

[10] Khy, S., Ishikawa, Y., Kitagawa, H.: A novelty-based clustering method for on-line documents. World Wide Web Journal (2007) (in press).

[11] Egghe, L., Rousseau, R.: Introduction to Informetrics: Quantitative Methods in Library, Documentation and Information Science. Elsevier, Amsterdam (1990)

[12] Han, J., Kamber, M.: Data Mining: Concepts and Techniques. 2nd edn. Morgan Kaufmann (2005)