Temporal Multi-Page Summarization

Adam Jatowt[1]

National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, 619-0289 Kyoto, Japan
Phone: +81-774986828
Fax: +81-774986960
adam@nict.go.jp

Mitsuru Ishizuka

University of Tokyo
7-3-1 Hongo, Bunkyo-ku, 113-8656 Tokyo, Japan
Phone: +81-358416698
Fax: +81-358418570
ishizuka@miv.t.u-tokyo.ac.jp

Abstract. With the increasing popularity of the Web, efficient approaches to the information overload are becoming more necessary. Summarization of web pages aims at detecting the most important contents from pages so that a user can obtain a compact version of a web document or a group of pages. Traditionally, summaries are constructed on static snapshots of web pages. However, web pages are dynamic objects that can change their contents anytime. In this paper, we discuss the research on temporal multi-document summarization in the Web. We analyze the temporal contents of topically related collections of web pages monitored for certain time intervals. The contents derived from the temporal versions of web documents are summarized to provide information on hot topics and popular events in the collection. We propose two summarization methods that use changing and static contents of web pages downloaded at defined time intervals. The first uses a sliding window mechanism and the second is based on analyzing the time series of the document frequencies of terms. Additionally, we introduce a novel sentence selection algorithm designed for time-dependent scenarios such as temporal summarization.

Keywords: web document summarization, temporal web page analysis, change detection and relevance, web collection

1 Introduction

Multi-document summarization has recently attracted much interest in the text processing community (for example, see [35][5][31]). A multi-document summary contains the most salient concepts extracted from several sources and presents them in a condensed form. It is appealing to employ summarization techniques for web pages due to the growing popularity of the Web as a medium for publishing and exchanging information. However, web pages are unlike traditional document types. They are organized spatially in visual and structural units, such as frames and links, and are rich in multimedia and other non-textual contents. Additionally, the content and structure of a web document can change any number of times in a short period of time. This changeability is an important characteristic of web documents, and it differentiates them from other types of documents. It forces us to consider a new dimension of pages - time. We have investigated ways to summarize the contents of web documents as they change in time. This summarization approach is called a temporal summarization, since it is based on analyzing temporal versions of web documents focusing on their changing contents.

There are two main approaches to summarizing a web document: using textual contents extracted from the document [8][9] and using contextual data from the documents hyperlinked to it [4][16][23] (Figure 1). The first approach suffers from data scarcity; it is difficult to construct an accurate summary of a web document having little content. This led to the second approach, which exploits the structure of the Web. It is based

[1] The main part of this work was done when the author was at University of Tokyo

on analyzing anchor texts or complete linked documents. This approach also has limitations due to possible incompleteness of the contextual information, which can also be affected by the various kinds of relationships between linked web documents.
Temporal summarization of web pages can be regarded as an extension of content-based methods. A page is considered to be a dynamic entity that changes and evolves over time [27]. To summarize a single web document over a given time interval, first the web page is periodically downloaded with a certain frequency over that interval. The temporal versions of the page are then analyzed and any changes are identified (Figure 2). These changes are extracted and used to construct a summary [28].
In this paper we describe temporal multi-page summarization, that is, summarization of temporal collections of web documents. It is important to note the difference between traditional multi-document summarization and the summarization of temporal versions of a collection of related web documents, a "web collection". In the former, the summary is usually based on several, similar documents discussing the same event or narrow topic from different perspectives or with different levels of detail. In the latter, the documents may discuss the same event or topic only to a certain extent. Additionally, there may be time differences between the appearances of related information in different documents in the collection. Therefore, special content analysis methods must be used to detect distributed in time information that is common and related.
Web collections to be used for temporal summarization should be constructed with special care. The web documents not only need to be informative about a particular topic but also contain up-to-date information. Thus, the documents must provide fresh content related to a common topic, that is, they should be frequently updated. To determine whether a candidate document should be included in a collection, its topic relevance and the average frequency and size of the changes should be estimated.
The usefulness of temporal summarization can be seen when a user requires information about hot topics related to his area of interest. When a collection contains many "active" documents that are frequently updated, the main concepts and popular changes concerning a user-specified topic can be easily detected. If a lot of data needs to be analyzed, automatic processing techniques can be used. Temporal summaries could also be beneficial from a commercial point of view. For example, companies could summarize common information gleaned from their competitors' websites or abstract product and market-related, time-varying data.
The Web has become the biggest information repository. Unfortunately, information retrieval in the Web is still inefficient and users are often overloaded with unnecessary data. Besides its enormous size, the Web is also very dynamic - not only is information being continually added, it is also being continually removed and is often irrecoverably lost. The Internet Archive Project is the best-known initiative for preserving the past contents of web pages [25]. However, the data is too voluminous and volatile to be fully extracted, stored and utilized by existing systems. Thus, it is necessary to focus more on efficiently processing changing data so as to mitigate the effects of information overload and volatility.
The paper is organized as follows. In the next section we discuss related research. Section 3 presents methods for creating temporal web collections and discusses their characteristics. Section 4 introduces our two term scoring methods for summarizing temporal web collections, and Section 5 compares and discusses them. Section 6 describes an approach to selecting and ordering sentences. Section 7 discusses the results of our experiments. We conclude with a short summary in the last section.

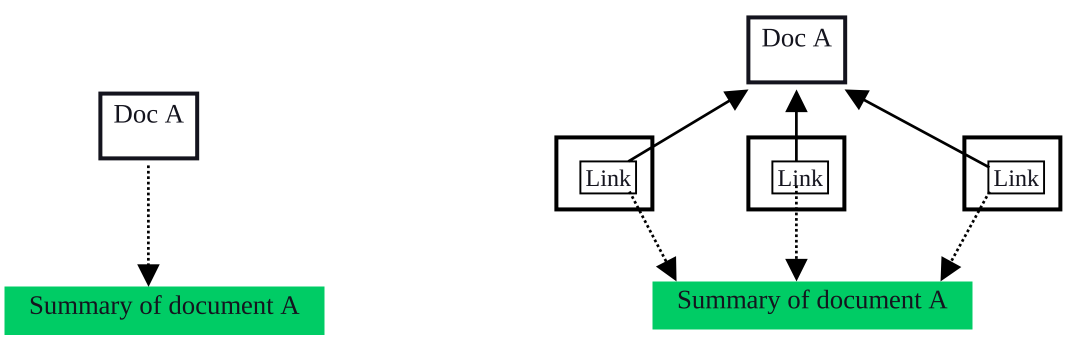

Fig. 1. Main web document summarization approaches

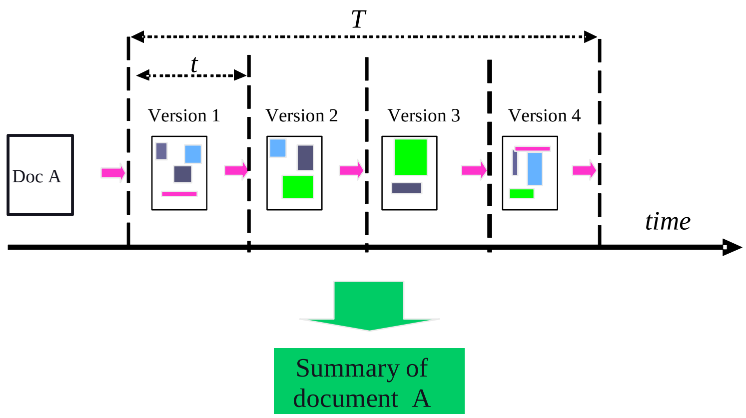

Fig. 2. Temporal summarization of a web document

2 Related Research

Studies of the characteristics of the Web [7][19][14] have revealed that the Web is a highly dynamic environment and attempted to estimate the frequencies of web page changes or to predict updating patterns. A growing number of web sites are adopting a push work mode in which updated information is sent directly to interested users. However, the majority of web sites still use the conventional pull mode. To alleviate the impact of changing web contents on information retrieval, the continual query concept was developed (for example, see [32][38][13]). Continual queries systems search for relevant web pages and maintain their freshness by continuously monitoring them with appropriate frequencies. A continual query application strives to optimize its web crawling schedule so that resource usage is minimized and web pages are kept as fresh as possible. Consequently, the responses to user queries are more up-to-date and generated at less cost.

Several change detection and monitoring systems have been proposed [18][26][33][6]. Most require the user to provide the URLs of target resources and specify a notification method and several tracking parameters (monitoring frequency, types or locations of changes of interest, etc.). The well-known AIDE [18] uses the HTMLdiff algorithm to present different types of changes from archives of web page versions in a graphical way. WebVigiL [26] is an information monitoring and notification system for detecting customized changes in semi-structured and unstructured documents. WebCQ [33] is a large scale web tracking application with interesting presentation capabilities. Finally, ChangeDetector [6] uses machine learning techniques for effective monitoring of entire web sites.

Change detection can usually be done automatically in a straightforward way by comparing the contents of sequential versions of a web page; however, assessing the relevance of the changes is a more complicated task. One difficulty comes from the subjective question of whether a change is related to the topic of the web document or web collection. Walden Paths [21] is an example of an application utilizing relevance assessment of changes. It constructs knowledge paths of related web pages for educational purposes. The authors needed a method for assessing the relevance of changes in order to maintain the consistency of their topical collections. The path manager in Walden Paths quantifies the magnitude of the relevance of the overall changes based on the presentation, structure, and content of past versions of web documents. In another example, an agent called Do-I-Care [1] uses supervised learning based on feedback from the user to manage its profile and to track changes of interest to the user. In the case of changing web collections change relevance is a fundamental issue, which can be assessed by analyzing contents and temporal contexts of changes. It is achieved by comparing changes to the contents retrieved from other pages in the collection in limited time spans.

Automatic document summarization based on sentence extraction was first introduced by Luhn [34]. Extractive-type summarization systems usually use shallow, statistical information derived from the documents to determine the usefulness of sentences (for more, see [35]). The next step after single document summarization was investigating methods for summarizing several similar documents simultaneously (for example, see [35][31][5]). Multi-document summarization requires detection of similarities and differences between multiple documents in order to generate a single output. It becomes more difficult as the content diversity of the documents increases. Unfortunately, this is the case with temporal web collections of related documents due to their higher content diversity. Therefore, intuitively, it is important to ensure that the documents are topically close to each other.

Topic Detection and Tracking (TDT) (for example, see [3][43]) focuses on detecting and classifying events from streams of news articles or from retrospective corpora. Similarly to TDT, we utilize the temporal adjacency and content similarity of documents [43]. The main idea is that newsworthy events can be identified by searching for bursts of reporting activity in close time proximity. Although, like in TDT, our research is focused on temporal aspects of information, we combine summarization with temporal analysis of textual data. Additionally, we attempt to recognize not only popular news events but also more generally popular concepts in arbitrary topics. Thus, rather than analyzing different news articles, we analyze fixed group of related web documents during a certain time interval. Hence, summaries reflecting content changes of an arbitrary collection of web documents can be constructed.

Summarization of news articles for detection of newsworthy events is demonstrated in [24][2][36][39]. Google News [24] is an online application that tracks a large number of newswire sources and displays the latest news related to a query. Allan et al. [2] constructed temporal summaries of news streams by evaluating the novelty and usefulness of incoming sentences using recall and precision measures. Newsblaster [36] is an online news clustering and summarizing tool that produces automatic journal reports of the most important news events. In a similar way, NewsInEssence [39] clusters popular news from a list of news sites enabling a user to choose a cluster to summarize. However, since these systems use pre-selected and fixed resources for the newswires as input data, an application is needed that can summarize new information from any kind of resources specified by the user. WebInEssence [40] is an example of a Web-based multi-document summarization and recommendation system that can do this. However, our approach is different in the sense that we attempt to do temporal summarization of web documents, that is, the summarization of their changing contents, instead of considering web pages as momentary snapshots. For online web collections, multi-document summarization of common changes has been demonstrated with the ChangeSummarizer system [27], which uses temporal web page ranking based on contextual change information. Nevertheless, despite the popularity of the Web, there is still a lack of applications for retrospective summarization of changes in archives of web documents.

3 Temporal Web Collection

We assume that web documents have fixed structures during the chosen summarization period, so only content changes need to be considered. A study of the effect of changes in web pages on their perceived importance showed that content changes are more important to users then other types of changes, like presentation or structural ones [22]. Content changes in a single web page have following characteristics:

- timing and lifetime
- spatial position on page
- type
- content and amount
- context (local or distant)

While we may not know the exact timing of an occurrence of a change, we can approximate it by setting the sampling frequency sufficiently high. The lifetime of added content can be represented as the number of versions or intervals during which it remains on the page. The spatial position of the change on the page can be represented by the paragraph where it occurs or by the distance from some point on the page. The possible types of textual changes (words, sentences, paragraphs) are insertion, deletion, and modification. The content of a change is its semantic meaning, while the amount describes the size of the change. Finally, local context means all the neighboring elements that are on the page when a change occurs. Distant context, on the other hand, is the data found in the rest of the collection at the time of the change.

A web collection is defined as a set of web pages related to the same topic. A user may already possess a collection of related web documents about his area of interest that he likes and usually consults for information discovery purposes. Otherwise, there are two simple methods for constructing web collections about a given topic. Web collection can be downloaded from any existing web directory (for example, the Open Directory Project [37] or Yahoo! [42]) that clusters together similar documents; it can also be created using search engine results. In the first case, a user is limited to a fixed number of topics, hence, he may not find the collection of web pages about any specific topic that he likes. However, there is no need for content analyzing and filtering since web pages are already manually classified to given topics. In the second case, while any combination of words can be entered into the query box of a search engine, relevance filtering is often needed to obtain a set of topically close web pages. Moreover, there may be duplicate pages, which must be removed. Such duplicate documents would degrade results due to miscalculation of frequencies of terms in a collection. For automatic duplicate document detection, the contents of all the page pairs in the collection must be analyzed and compared. If two pages have content similarity higher than a given threshold, they are considered to be duplicates, and one is removed from the collection. This analysis can be done using the number of query terms, or the similarity of the content to the centroid TF*IDF vector of the collection [41]. Several other approaches for constructing collections of related web pages have been proposed in the past (for example, see [10][15][11]).

It is also possible to obtain web collections from web communities. In contrast to a web community, the pages in a web collection do not need to be connected by links to each other. Content relevance to the topic is the only requirement for being categorized in a certain collection. A web community is defined as a set of web pages that link to more pages inside the community than to ones outside the community. Consequently, web pages in the community may not be topically related to each other. A web community can be obtained using any of several web community distilling techniques ([12][20][29], for example). Next, it can be transformed into a web collection after the contents of its member pages are analyzed in relation to a certain topic.

To extend a web collection of N documents into the temporal dimension, we collect the samples of member pages with some fixed delay. Let t be the time interval between collecting successive samples of a single document. The document may change with a specific frequency pattern or a random pattern. Ideally, the sampling schedules should be set based on the expected patterns of modifications for each document. However, for simplicity, we assume one general t for all web pages in the collection. Consequently, after some interval T, which defines the summarization period, the collection will grow to the $N * T/t$. In another way we can build retrospective web collections by downloading past page versions of web pages from available web archives such as Internet Archive [25]. By comparing two consecutive versions of the same document, we can identify the changes that have been made during t.

The longer the t, the lower the recall of changes due to the possibility of occurrence of short-life content in web documents. This is often the case with newswire resources and popular, fast-changing pages. Some new content additions may pass undetected if they remain only a short time on the page, resulting in missing data. While higher frequency of page sampling usually results in increased recall, the consumption of resources is also increased. Let $C_a = \{C_1, C_{2,\ldots,} C_n\}$ be the set of all changes made on web page a in some time interval, and let $D_a = \{D_1, D_{2,\ldots,} D_u\}$ be the set of all discovered changes. Assuming t_a to be the average delay between changes made on the web page, the recall of changes can be approximated as

$$R_a = \frac{|D_a|}{|C_a|} \approx \frac{t_a}{t} \quad \text{if} \quad t_a \le t$$

$$R_a = \frac{|D_a|}{|C_a|} \approx 1 \quad \text{if} \quad t_a > t \tag{1}$$

Consequently, the change recall for the entire collection depends on the number of pages that have an average time gap between changes shorter than the granularity of the tracking process:

$$R_c = \frac{|D_c|}{|C_c|} = \frac{\sum_{a=1}^{N} |D_a|}{\sum_{a=1}^{N} |C_a|} \approx \frac{\sum_{a=1}^{N} t_a}{N * t} \quad \text{for a where} \; : \; t_a \le t \tag{2}$$

If summarization period T is short and contains only a few intervals t, there is usually a small number of pages that have any changes. Therefore, the effect of these web pages on the final summary is relatively high. Thus, there is a risk that the final summary has lower accuracy with regards to the actual changes in the topic, since only a few pages determine the output. On the other hand, if T is long and contains many

intervals *t*, more changes should be detected, resulting in any particular version of a web page having less effect on the final summary.
There can be a difference in the timing of changes made to two similar web pages in reaction to a given event occurring at a particular point in time. We assume that these pages report the most important events related to the user's area of interest. For a newswire source, the change to the page reflecting a particular event is normally made fairly quickly (in hours or days). For a more static web page, it may take longer. This difference in the response of web pages to the occurrence of some event is referred to here as "temporal diversity" to distinguish it from "content diversity".
To estimate the dynamic characteristics of a single web document, we introduce two temporal parameters: change volume and activity ratio. The first represents the average volume of changes in a page during a particular time period. The second is the ratio of the number of modified versions to the total number of versions of a web document. It shows how active a given web page is in terms of changes in its textual content. Both parameters are useful in estimating the temporal diversities of documents.
An ideal collection of web documents should contain documents closely related in terms of content and temporal characteristics. In other words, the documents should have low content and temporal diversities. The quality of input, that is, the characteristics of the collection, affects the quality of output. If the majority of documents are on-topic and have frequently updated contents, we can expect to obtain an informative and coherent summary. This does not mean the pages have to be strictly newswire type, but they should have semantically and temporarily close changes.
Although we concentrate here on analyzing single web pages, the proposed methods can be extended to summarize temporal collections of entire web sites. In the simplest approach to such an extension, one could combine together all the sub-pages belonging to a given web site and form a single super-page. Then, the collection of such super-pages can be summarized using the methods presented in this paper. However, since the content of a web site may not be completely related to the common topic of the collection, filtering or grading of the input documents could prove advantageous. For example, the grading scheme could be based on the distance between the constituent web pages and the main (or most related to the collection's topic) web page.

4 Term Scoring

We propose two methods for summarizing temporal web collections. Both are based on term scoring and sentence extraction. Terms in the web pages are assigned so-called "long-term" scores based on their macro-scale occurrences in the collection. The first method uses a sliding window to evaluate term importance, and the second one uses trend and variance analysis.

4.1 Sliding Window Method

This method assigns long-term scores to terms extracted from changed parts of web pages. First, the changed content in the web page versions is identified by comparing consecutive versions of each page at the sentence level. Here we focus only on the textual content of the documents; links and multimedia are neglected. Sentences from consecutive versions of every document in the collection are compared to identify the inserted and deleted sentences. If a particular sentence appears only in the later version, it is treated as an insertion change. If a sentence is found only in the earlier version, it is treated as a deletion change. To avoid detecting minor sentence modifications (a few changed words, grammatical or structural corrections, etc.), we use a similarity comparison algorithm, which examines the types of words and their order in the analyzed sentences. Analysis of the number and order of common words in every pair of sentences from neighboring versions produces a similarity value for each sentence pair. A sentence is considered to have been changed, if it does not pass the similarity test in each possible pair, that is, it has lower similarity values than some predefined threshold.
After the changed data is separated from the collection, words and bi-grams are extracted to form basic pool of features. We use bi-grams, that is, pairs of consecutive words, since they may have different semantics than their constituent words considered separately. Standard text preprocessing, such as

stemming and stop-words filtering, is then conducted. The result is a final working set of terms including stemmed and pre-filtered words and bi-grams extracted from the changes.

The basic approach for a calculation of long-term scores of terms is shown in Equation 3. The scoring scheme is based on the distribution of terms in the changed portions of the documents during interval *T*. Terms that appear in the changed parts of many different documents in the collection are assigned a higher "document frequency" value than the ones that are in the changed portions of only a few documents. Additionally, terms that appear more frequently inside the changed portions have a higher "term frequency" value assigned. The document frequency is more important than the term frequency because the former reflects the distribution of the term among different web documents in the collection. Consequently, the equation part that reflects the document frequency of a term is in the form of a natural exponent. In Equation 3, *DF* means the number of document versions containing a given term in their changes, whereas TF_j is the frequency of the term in the changed portions of some *j-th* document sample in the collection. The term frequency is divided by the number of terms in each change, that is, the size of change S_j. The first part of Equation 3 is the average term frequency over all web page samples *N*n*, where *N* is the number of different web documents and *n* is the number of versions of each web page.

$$S_{term} = \frac{\sum_{j=1}^{N*n} \frac{TF_j}{S_j}}{N*n} * \exp(DF) \tag{3}$$

As it was mentioned before, in this method, two basic types of changes are considered: insertions and deletions. Similar insertions in a large number of web pages in a collection in a proximate time indicate the onset or continuation of some important and newsworthy event. Thus, terms occurring frequently in such insertion-type changes are scored highly. The deletion-type changes might be considered as unnecessary, out-dated content that must be removed and replaced with new content. However, if many web pages contain similar terms in their deletion-type changes during a limited time span, it is likely that some important event or concept related to these terms has ended. In this case, such terms should be scored highly. Finally, a term's overall long-term score is the combination of the scores related to both types of changes.

The final version of the long-term scoring equation considers different types of changes. Let d_x denote a deletion and i_x an insertion change in a single web page, where *x* is the number of the document version (Figure 3). Thus, d_x indicates the whole content that was deleted from the *(x-1)-th* version of the web page, and i_x indicates the whole content that was added to the *x-th* version.

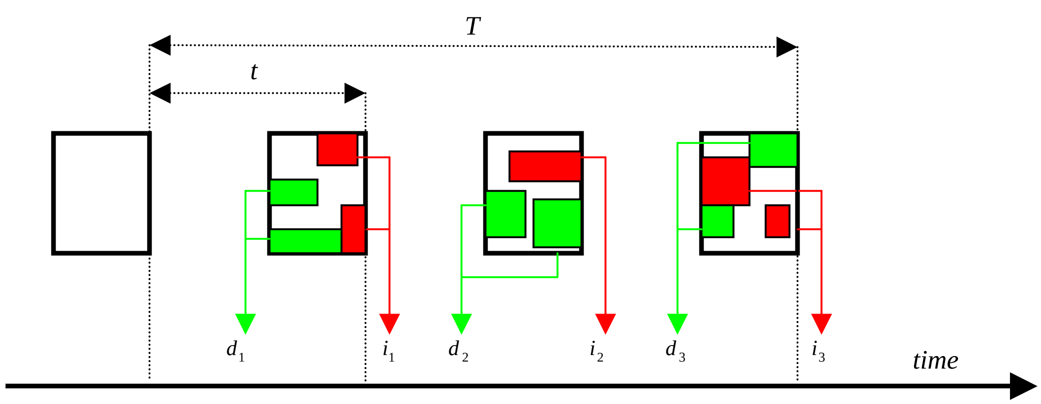

Fig. 3. Temporal representation of deletion- and insertion-type changes for a single web page

The importance of terms is calculated by analyzing their distributions for both types of changes in the temporal collection. High scores are assigned to terms that occur frequently in inserted or deleted parts of a substantial number of documents in a short time. This temporal closeness is used to detect bursts of terms and is estimated by a sliding window with adjustable length *L*. The window moves sequentially through the temporally ordered collection so that *L* versions of documents are considered in the same time (Figure 4). Terms found in changes inside the window are assigned partial scores for each window position. These scores are computed based on the general weighting scheme given by Equation 3. However, the differences in term and document frequencies for both kinds of changes are used here. This modification is based on the assumption that newsworthy, popular concepts are located in the same type of changes in numerous and temporally proximate web page versions. Thus, the partial score of a term in each window position is given by:

$$S_{term}^{win} = \frac{\sum_{j=1}^{N*L} \left| \frac{TF_j^I}{S_j^I} - \frac{TF_j^D}{S_j^D} \right| * \exp \left| DF^I - DF^D \right|}{N * L} \tag{4}$$

The superscripts *I* and *D* indicate insertion and deletion type changes inside a single window position. The term and document frequencies of each term are calculated only for the changes occurring in the area covered by the window. Thus, TF_j^I and TF_j^D denote frequencies of the term in the inserted and deleted changes of the *j-th* document sample inside the window, while DF^I and DF^D denote document frequencies of the term calculated over the whole pool of insertion and deletion changes inside the window. The overall long-term term score (Equation 5) is the average of partial term scores calculated for all window positions, *Nw*.

$$S_{term}^{overall} = \frac{\sum_{win=1}^{Nw} S_{term}^{win}}{Nw} \tag{5}$$

If a given term occurs mostly in only one type of change at many different window positions, its overall score will be quite high. On the other hand, if the term is equally distributed in insertions and deletions inside a majority of window positions, the overall long-term score will be low. In other words, terms that occur in bursts of deletions or in bursts of insertions in a substantial number of window positions are deemed important. Therefore, the absolute values of the differences in term and document frequencies for both types of changes are used (Equation 4). The length of the window is set depending on the type and characteristics of the analyzed collection. It should be longer for collections containing web documents with considerable temporal diversities. However, the longer the window, the greater is the risk of missing events that have lifetimes shorter than the window length.

The term calculating method and the effect of the window length on the final term scores can be shown in a simplified example where two instances of only one term constitute all the changes in the collection. The term occurs once as a deletion and once as an insertion during the period *T* in a hypothetical collection of *N* documents each having 18 versions. We consider several scenarios of different relative occurrences of these two instances of a change in time. Let us assume that there are no more other changes in the collection. Thus, in each scenario only two document versions contain any changes and the rest of the collection is assumed to be static. Also, both instances of a change cannot occur in the same document version. Let first instance - an insertion type - be fixed in middle of period *T*, that is, in the ninth version of a one document during all scenarios. For each single scenario, we will place the second (deletion type) instance subsequently in every consecutive version of another document and then compute term scores for each such scenario. Thus, starting from the beginning until the end of *T* we shift the deletion instance and calculate the score of the term for each position. In this way we can calculate the final term score for different temporal distances of both instances of the term. In Figure 5a we plot the relative term score calculated for different scenarios. The vertical axis represents the relative term score and the horizontal one represents the position of the second instance of the term as the number of document version where this instance occurs. The graph in Figure 5a shows also the relative score for three different lengths. So, in general, the graph displays the influence of the temporal closeness of change instances and the size of the sliding window on the long-term term score. It is evident that if both term instances occur close enough together to be covered by the same window, the long-term term score decreases. It reaches minimum value when the insertion- and deletion-type changes occur at the same time point, that is, in the versions of two different documents which were sampled in the same time. In conclusion, the term score decreases as the distance between both term instances becomes smaller. For shorter windows, this decrease begins later since the window embraces changes in only a few neighboring web page versions. Thus, we can see that for a short window length *L*, the algorithm works in a short-term mode. This means that high scores are assigned to terms whose change instances are clustered in short time spans during interval *T*. On the other hand, with a long *L*, the temporal granularity of event discovery is reduced. Therefore, for longer windows, some short-life events may not be detected. However, there is a trade-off here since, unlike in the former case, the negative effect of temporal diversity between different web pages is reduced. Figure 5b shows the case when both instances of the term's change are of the same type.

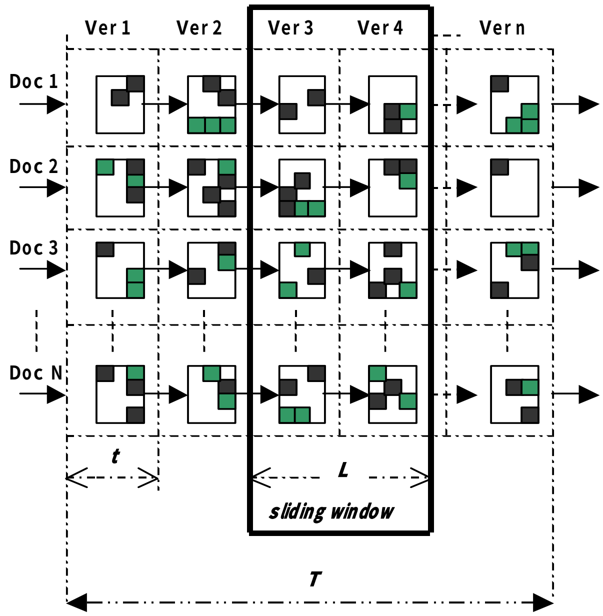

Fig. 4. Sliding window of length L in web collection of N documents where each document has n versions. Colored areas represent insertions and deletions in web page versions

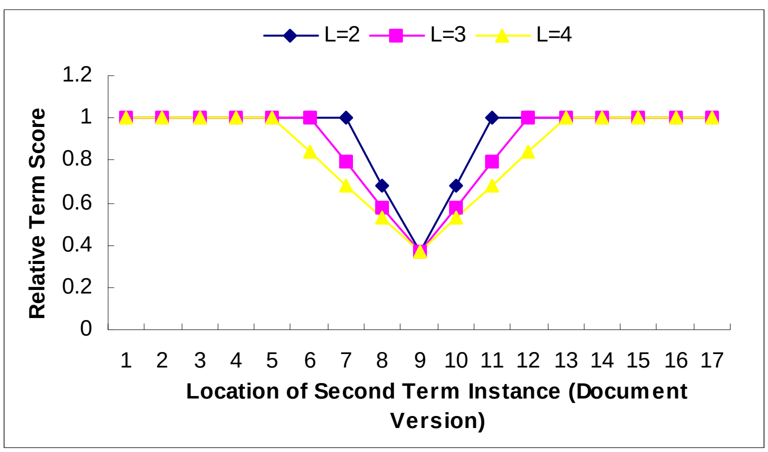

Fig. 5a. Relative long-term score for two opposite instances of same term for different window lengths

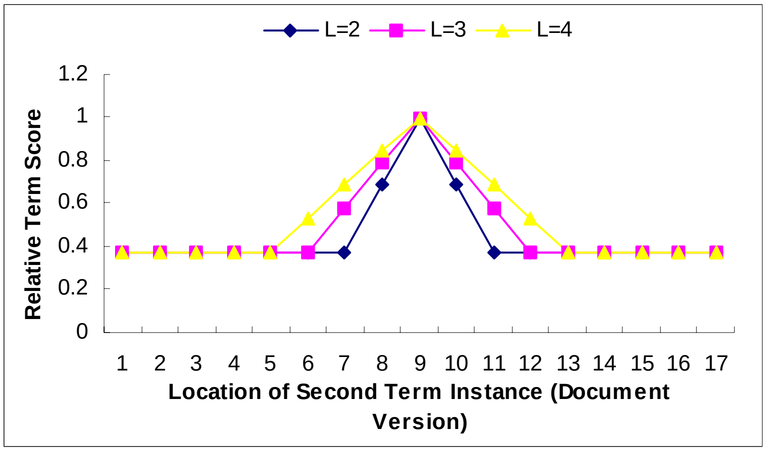

Fig. 5b. Relative long-term score for two same instances of term for different window lengths

4.2 Statistical Analysis Method

The second method for calculating the long-term scores is based on analyzing the statistical parameters of time series of term occurrences. It uses the entire content of the collection, unlike the sliding window method, which uses only the changed parts of web documents. The text preprocessing steps, stemming and stop-word filtering, are the same, and single words and bi-grams are similarly considered as the working set of features.

The document frequency is calculated for each term at every time point during the summarization interval. If we visualize the collection in the same way as illustrated in Figure 4 for the sliding window method, the document frequency of a given term is the number of document versions from a single column that contain the term. In other words, the document frequencies of terms are calculated for each group of document versions downloaded at the same time. A term's document frequency reflects to some extent the popularity of the topic described by the term at a given point in time. From the time series of document frequency values we can identify the trend and major changes in the usage of the term (Figure 6).

We search for terms in the collection that have temporal characteristics indicating their high importance. To represent the trend in the changes in a term's document frequency during interval *T*, we apply simple regression analysis between time and the document frequency. Trend detection of terms by simple regression technique was used before, for example, to increase the efficiency of information retrieval by considering changes in meanings of words over long time spans [30]. In our method, every term has slope *S* and intercept *I*, both calculated so that the plot of its document frequency can be fitted with a regression line (Figure 6). The slope shows the general trend in document frequencies during period *T*. The intercept point shows the term's frequency of usage at the beginning of the collection tracking. Additionally, variance *V* is calculated for each term to represent the average magnitude of the changes in the term's importance for interval *T*.

To determine which terms are important from the point of view of a temporal summary, we need to compare their statistical parameters. A slope that differs from zero indicates an emerging or disappearing event or concept. A term with a slope much higher than zero is assigned a high score. If it also has a low intercept point, there is a high probability that the term represents a new, emerging topic. A term with a negative slope may be associated with an already completed event or with a concept that is no longer of much interest. The term's score should be low if the user is interested only in emerging or developing events or concepts. A term with a nearly zero slope is probably not "active"; that is, the average level of interest in the topic described by the term remains the same throughout the summarization period. To verify this, the term's variance should also be examined. If it is high, there is a large fluctuation in the number of document versions containing the term during interval *T*, which means that it is probably an active type of a term. Finally, a high intercept point means that the term probably does not add any new information or discuss any new concept. Thus, intercept plays a role of a novelty parameter in the temporal summarization. It enables identification of terms that are novel during tracking process. We define the score of a term by the following formula:

$$S_{term} = \alpha * \frac{\left|2 * S^{R} - N_{S}^{R}\right|}{N_{S}^{R}} + \beta * \frac{N_{V}^{R} - V^{R}}{N_{V}^{R}} + \gamma * \frac{I^{R}}{N_{I}^{R}}$$
$$\alpha + \beta + \gamma = 1 \qquad (6)$$

The score is expressed as a linear combination of the ranks of a term, S^R, V^R, and I^R, in the slope, variance, and intercept ranking lists, respectively. Terms are ordered in ascending order based on their slopes, variances, and intercepts. N_S^R, N_V^R, and N_I^R denote the numbers of different ranks for the three lists. Weights α, β, and γ specify the strengths of each parameter's effect on the final score of a term. The choice of weights depends on the kinds of terms that should be favored most. Equation 6 is composed of three parts. The first part is the contribution to the final score of the term's overall trend. It gives a preference to terms with high absolute values of their slopes. The middle part is related to the term variance and rises along with the increasing variance rank of terms. The last part is the contribution of the term's intercept. It is positive when the intercept rank is low, that is, when the term has a low intercept value so that it is novel to the collection. The contribution of each part can be controlled by the user by adjusting the weights. Generally, the combination of weights depends on the required type of output summary.

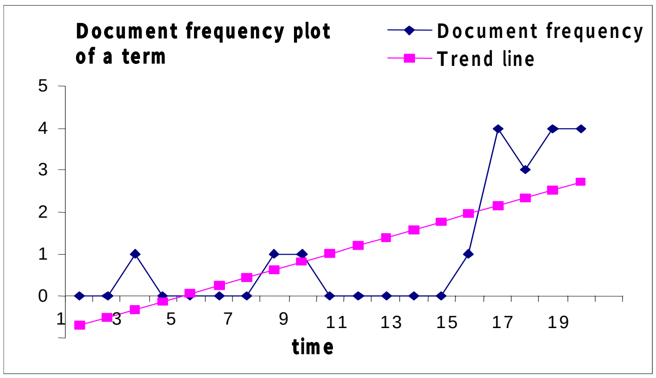

Fig. 6. Document frequency plot of a term and its trend

5 Comparison and Discussion of Term Scoring Methods

The two term-scoring methods described above are the core of our summarization approach. Their choice determines the type and characteristics of the summary that will be constructed. Here we discuss the similarities and differences of the both methods.

Words and bi-grams are inputs to both the sliding window and statistical analysis method. Their output is in the form of a ranking list of terms to be used for sentence selection. Both methods utilize a measure of document frequency variance. In the statistical analysis method, the variance measure is directly used as part of the scoring equation. In the sliding window method, it is expressed as the sum of the absolute values of the differences between the document frequencies of terms in the deleted and inserted types of changes. In the statistical analysis method, except for the variance measure, the trend and novelty of terms are also considered. This makes it possible to construct different types of summaries by adjusting the combination of weights used in Equation 6.

Both methods use the temporal proximity of terms as a basic criterion for computing the scores. This temporal closeness is determined by the window length in the sliding window method. In the statistical analysis method, only fixed time intervals are examined. In other words, we analyze only the *x-th* document versions at the same time. Consequently, in the statistical analysis method, there is no means for eliminating inaccuracies due to the temporal diversity of documents. With the sliding window method, one can adjust the temporal area to be analyzed by changing the length of the window so as to detect similar contents in temporally diverse documents. Additionally, both types of changes are considered in the sliding

window method. The distributions of insertions and deletions are analyzed in a temporal collection, and their differences are used to compute the term scores. In short, the variance calculation in the sliding window method, apart from considering both types of changes, can also have different time scales due to the use of an adjustable window.
The next difference between the methods is the type of data that they use. Sliding window method uses only the changed contents of the document versions, whereas the statistical analysis uses all the contents. It considers all the text on the web page versions, including the static text. The output of the sliding window method can thus be called a "dynamic summary" versus the "temporal summary" obtained by using statistical analysis method, which is a more general approach from the viewpoint of temporal web page content. In many cases, static text also has important meaning that can explain why it remained unchanged for a relatively long time on a web page. However, the reason for the "long life" of static data may not always be related to its information purposes. Such content can sometimes have a structural or navigational meaning; for example, it may be the title of a topical section or anchor text for links to other pages in the web site. Unfortunately, it is difficult to accurately detect the purpose of page content and to determine its usefulness for a temporal summary. It would be helpful to have methods for recognizing the functions of content in web documents.
Another difference between the two methods is that, in the statistical analysis method, only document frequencies are used whereas, in the sliding window method, term frequencies inside the document versions are also analyzed. The term frequencies show how often terms occur in the document versions on average whereas the document frequencies give an estimate of the frequency of usage of terms in different pages throughout the collection. The document frequency has an exponential form in the sliding window method and a linear one in the statistical analysis one.

6 Sentence Selection and Ordering

In the last step, sentences about popular concepts are extracted for a given period and presented to the user as a temporal summary. We calculate scores of sentences in the changed parts of the collection for sliding window method or in the all collected textual contents for statistical analysis method.
The simplest method for finding the significance of a sentence is to calculate its average term score. Those sentences that have scores higher than a given threshold are selected for the summary. Let us call it the "average importance method".
Another approach to sentence selection presumes identification of time points, when particular terms have their most salient scores. In the previous sections we described two methods for calculating the "long-term" scores of terms based on their distributions and statistical parameters over the entire summarization interval. However, terms can also have "momentary scores", i.e. scores computed for each time point during *T*. These scores can be used for sentence selection. They are calculated based on the local and neighboring frequencies of the term. Every term has a momentary score computed for the set of page versions from the same time point (the single column of page versions in Figure 4) depending on the local and neighboring values of its document frequencies. The peak momentary score of a term indicates the point where the frequency increase was the largest and the frequency remained later for some time on the high level. This point corresponds to a high probability of the onset of an event or a concept associated with the term (Figure 7). Therefore, sentences that have terms with the highest local scores should be selected. The simplest version of the momentary weighting function considers only the two nearest neighbors of each time point:

$$M_i^{\,j} = \left(1 + \frac{\left(DF_i^{\,j} - DF_i^{\,j-1}\right)}{DF_i^{\max}}\right) * \frac{DF_i^{\,j}}{DF_i^{\max}} * \left(1 - \frac{\left|DF_i^{\,j} - DF_i^{\,j+1}\right|}{DF_i^{\max}}\right) \tag{7}$$

Terms are scored highly at time point *j*, where the document frequency of a term *i*, expressed as DF_i^j, has increased significantly compared to the document frequency at the previous time point, DF_i^{j-1}. The momentary term score, M_i^j, also increases when the document frequency has a high value in the collection snapshot obtained at moment *j* compared to the maximum document frequency DF_i^{max} for the whole summarization interval. Lastly, if the term at the next time point, *j+1*, also has a high document frequency, it is likely to be in its peak importance state. The more general version of the momentary scoring function takes into consideration an arbitrary number, λ, of local points from each side of the time point for which the score is to be computed (Equation 8). The effect of document frequencies at more distant time points depends on their distances from the time point for which the momentary score is calculated. A given term has a peak momentary score at a certain time point if its document frequencies are for a relatively long time low before and high after this time point.

$$M_i^j = \frac{\sum_{v=j-\lambda}^{j-1} \frac{1}{j-v}\left(1+\frac{\left(DF_i^j - DF_i^v\right)}{DF_i^{\max}}\right) * \frac{DF_i^j}{DF_i^{\max}} * \sum_{v=j+1}^{j+\lambda} \frac{1}{v-j}\left(1-\frac{\left|DF_i^j - DF_i^v\right|}{DF_i^{\max}}\right)}{\left(\sum_{v=1}^{\lambda} \frac{1}{v}\right)^2} \quad (8)$$

For calculating momentary scores in the time points at the beginning and the end parts of interval T it is necessary to know document frequencies of terms in time moments before and after the period T. If this data is not available then we can extend T with “virtual time points” with such document frequency values so as there is a minimal bias created for the time points near boundaries of T. One possible option is to introduce virtual points with document frequency values equal to the document frequency values at the boundary points of interval T.
The momentary term score can be applied to sentence selection in two ways. First, the importance of a sentence can be estimated simply by multiplying the momentary scores of its terms by their long-term scores. The momentary score can thus be regarded as an indicator of the relative strength of a term at a particular time point during interval T. Thus, when estimating scores of sentences, we would consider not only the scores and numbers of their terms but also at which time points these particular sentences appear.

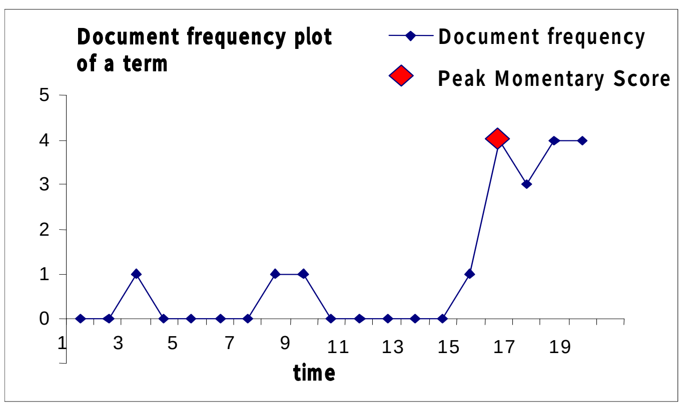

Fig. 7. Peak momentary score of a term

The second way is to take some u number of terms that have the highest long-term scores. Let us call these terms “leading terms”. Let TP_i denote a time point where the leading term i has its peak momentary score. From all the sentences containing i in TP_i the one with the highest average long-term score is chosen for the summary. Thus, for every candidate term we search for related sentences simply by applying the average importance method in the time moment when this candidate term has its highest momentary score. The same procedure is followed for the next leading term unless it is in any already selected sentence, denoted as SEN_J, where J is a set of different time points where this sentence appears in the temporal collection. If the next leading term is included in some SEN_J, then the algorithm checks whether the difference in maximal momentary score M_i^{max} of the term calculated for the whole T and its momentary scores at the time points of set J is at least once lower than some predefined threshold. If it is, the term is considered to be already represented in the summary, and the value of u is increased by one. Then, the next leading term is analyzed. The sentence selection algorithm is summarized below.

1. Define an empty set Z of sentences to be included into summary
2. For each term i from (1 to u):
    a. For each sentence SEN_J from Z:
        i. If term i exists in sentence SEN_J then
            1. For each time point j from J
                a. If (M_i^{max} - M_i^j < Threshold) then increase u by one and go to (2.)

b. Find time point TP_i of term i
c. Find the sentence from all document versions collected at TP_i that contains term i and has the highest average long-term score. Then include it in Z

3. Return Z

This algorithm ensures that the certain, desired number of top-scored terms will be represented in the summary. It is useful if some top-scored terms have scores much higher than the remaining terms. If only the average importance method is used, some of these terms may not be included in the final summary.
The length of the summary is set by the user as the number of sentences. Additionally, the user can limit the number of sentences selected from a single document version. This prevents one or only a few documents from dominating the output summary. To improve the coherence of the summary, we include the immediately preceding and following sentences of the sentences selected by the algorithm. Moreover, the cosine similarities of sentences are compared to identify redundancies to be removed. Lastly, sentences are arranged in a combination of temporal and content orderings (Figure 8). Temporal ordering positions sentences based on the sequence of web page versions in which they occur. It is necessary to know when a sentence first appears in the collection. Temporal ordering corresponds to the order of events or concepts appearing in the collection during interval T. In contrast, content ordering arranges sentences based on their relative positions in the documents. In other words, the combined ordering algorithm sorts sentences by two keys. The first key is the timestamp of the document version when the sentence first appeared, and the second one is its spatial position with relation to the beginning of the document. Finally, the sentences may be provided with links to their original page versions to enable users easy access to their contents.

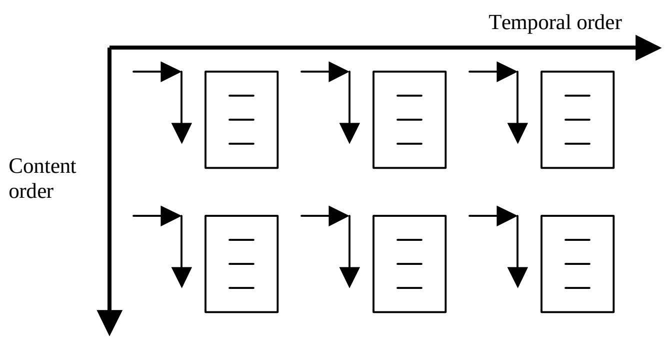

Fig. 8. Temporal and content orderings

7 Experiments

We have evaluated both summarization methods by conducting two experiments: automatic evaluation and human-based evaluation. In the first experiment, summaries of three collections were constructed for different sets of parameters. Each collection was composed of 200 pages obtained from search engine results after filtering. Every page had 18 temporal versions collected from March to June 2004. The queries that were used to construct these collections were following: “EU enlargement”, “chicken flu” and “passenger airlines”.
Creating manual summaries for evaluation purposes is difficult due to typically large sizes of input collections that should be used for the temporal summarization (more than 3000 documents). Unfortunately, we do not know other similar temporal summarization systems for retrospective web page collections that could serve as comparison. Thus, in order to roughly assess the quality of our summarization techniques we have used a modification of a content-based evaluation measure proposed by Donaway et al. [17]. This measure is based on computing the cosine similarity of the vector representations of the summary and the input collection. The assumption behind this measure is that the higher the similarity is, the better is a summary.

The graphs in Figures 9a, b and c show the effects of different window lengths and different lengths of interval *T* on qualities of summaries generated by the sliding window method. The quality is represented as a summary-collection similarity. From these graphs we can notice that summary-collection similarities have roughly similar values for short or long lengths of the interval *T*. An exception is the "chicken flu" collection for which summary for *T* equal to 18 time units has higher similarity score than for lower summarization periods (Figure 9b). The highest similarity to the source collection is attained for "EU enlargement collection" (Figure 9a), followed by "passenger airlines" collection (Figure 9c). Regarding the influence of window lengths on summary quality, we can observe that very small window lengths provide slightly worse results than larger ones. An interesting observation is that window length equal to 4 or 5 time units is a stabilization point so that the lengths longer than 4 provide highest and similar results. One exception here is the "EU enlargement" collection for *T* equal to 5 and 18 where there is no significant change in the summary–collection similarity for different window lengths. Additionally, in case of "passenger airlines" collection, this stabilization occurs later at more higher window lengths.

The graphs in Figures 10a, b, and c show the effects of different values of the weights from Equation 6 in the statistical analysis method on the similarity between the resulting summaries and collections. The similarities are plotted for each weight separately. For every plot two weights have fixed values of 0.1, and the third weight is treated as an independent variable. Thus, it is possible to observe the independent effect of each weight on the summary quality. Generally, slope and variance weights have positive effects on the quality of temporal summaries. Especially, an increase in the variance weight causes much improvement of the summary-collection similarity. However, the increase of the slope weight value does not improve much the quality of the summary. It is probably due to difficulty in choosing suitable topic and collection that would feature a clear trend. The increase in the novelty weight, on the other hand, significantly reduces the summary-collection similarity. This reduction is most likely caused by selecting sentences that contain new, original concepts, which have not been yet widely dispersed in the collection. The value of 0.4 is a stabilization point, after which the changes in the summary qualities are very low for higher weight values. Also, like in the sliding window method, summaries of "EU enlargement" and "passenger airlines" collections have higher similarities than the summary of "chicken flue" collection. We have also checked the effect of the mutually dependent slope and variance weights (denoted as "sl-va" in graphs) on the summary-collection similarity. The sum of both weights was fixed to be one. The value of the variance weight was used as an independent variable. The third weight - novelty parameter - was set to zero. The best results were achieved for such combination of weights when slope had higher values than variance. This confirms the positive impact of slope parameter on the summary-collection similarity.

Finally, the influence of the length of time interval *T* and different weight values on summary quality is shown in Figures 11a, b, and c for the collection "passenger airlines". Similarly to the sliding window method there is no significant relation between the length of *T* and summary-collection similarity.

The second experiment was carried out on a small collection of web documents so that a human created summary could be generated and used for the evaluation. The collection of five web pages, each having five temporal versions collected from 15th September to 23rd November 2003, was constructed from the ODP web directory [37] from the category "soccer". Each participant was asked to select seven most important sentences from the contents of the web documents. These sentences were then used to create a "majority agreement summary", which was compared with the automatic summaries constructed by our proposed methods. The results from sliding window and statistical analysis methods were also compared with random, random-lead, and term-frequency summaries. We constructed ten random summaries by randomly choosing sentences from the contents of the web documents. Then, for each random summary we have calculated its agreement with the human-created summary. The entry in the table is the average agreement for all 10 random summaries. Random-lead summary was obtained by taking only leading sentences from the changes of web page versions. Then, summary sentences were randomly chosen from leading sentences in the same way as for random summaries. Term-frequency summary was constructed by merging all changes of page versions together into a single document, calculating frequencies of terms and finally selecting top sentences that contained the most popular terms.

Table 1 shows the results of comparing the automatic summaries against the human-created ones. The random, random-lead, and term-frequency summaries were worse than those created by the proposed retrospective temporal summarization methods. Especially, random and random-lead summaries provided much worse results than the summaries created by our proposed methods.

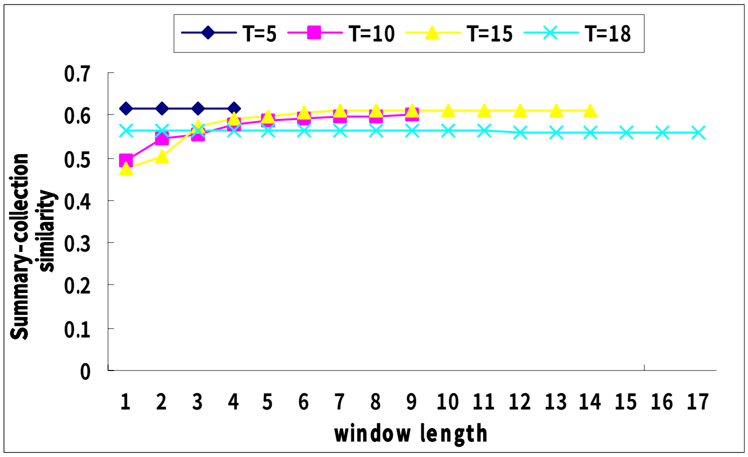

Fig. 9a. Summary-collection similarity for different window lengths in sliding window method for "EU enlargement" collection

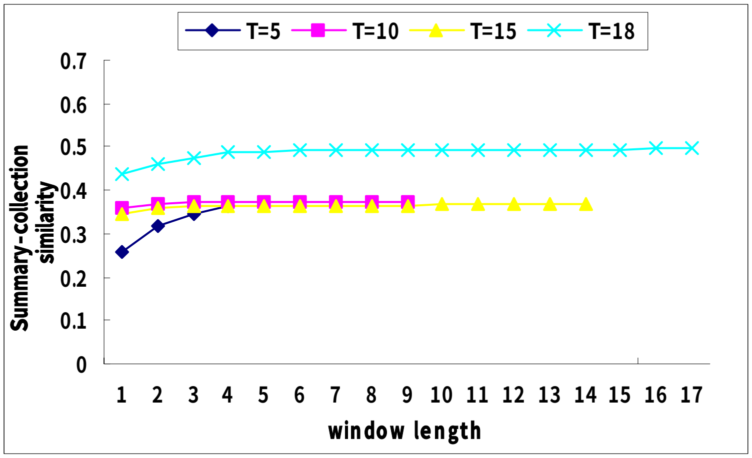

Fig. 9b. Summary-collection similarity for different window lengths in sliding window method for "chicken flu" collection

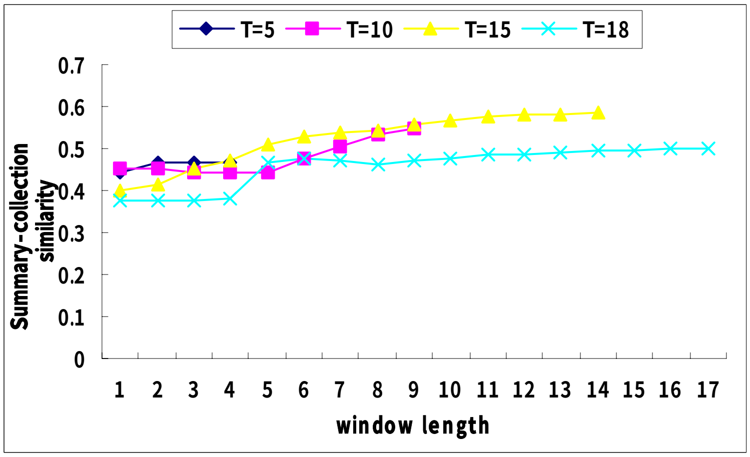

Fig. 9c. Summary-collection similarity for different window lengths in sliding window method for "passenger airlines" collection

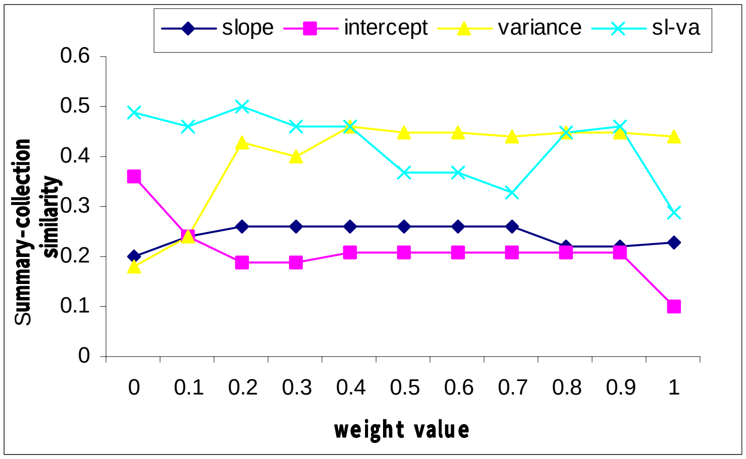

Fig. 10a. Summary-collection similarity for different weights in statistical analysis method for "EU enlargement" collection

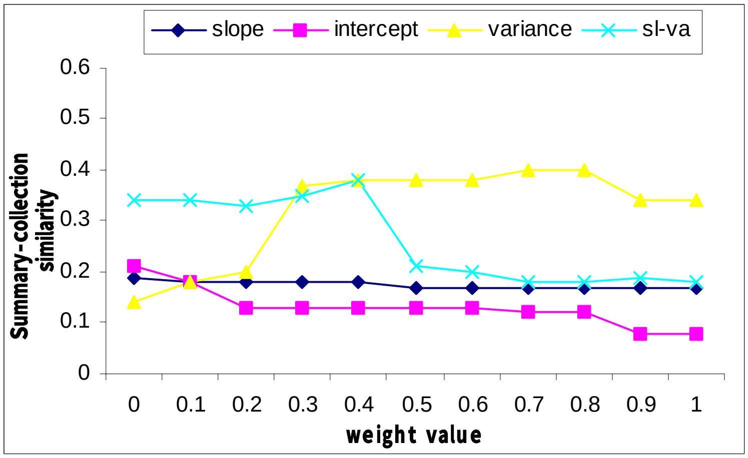

Fig. 10b. Summary-collection similarity for different weights in statistical analysis method for "chicken flu" collection

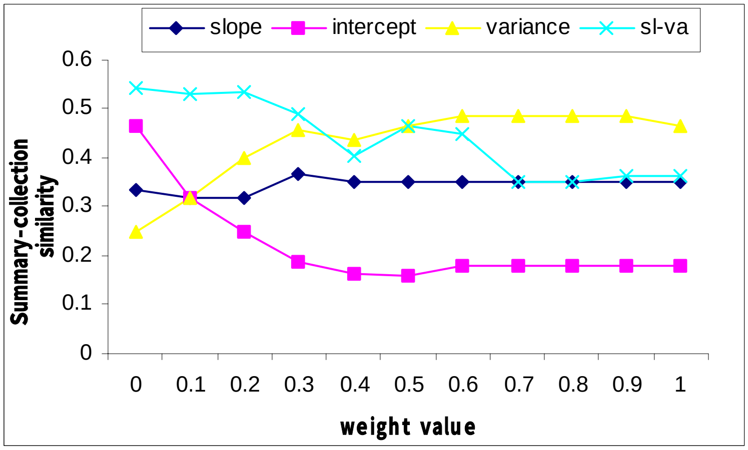

Fig. 10c. Summary-collection similarity for different weights in statistical analysis method for "passenger airlines" collection

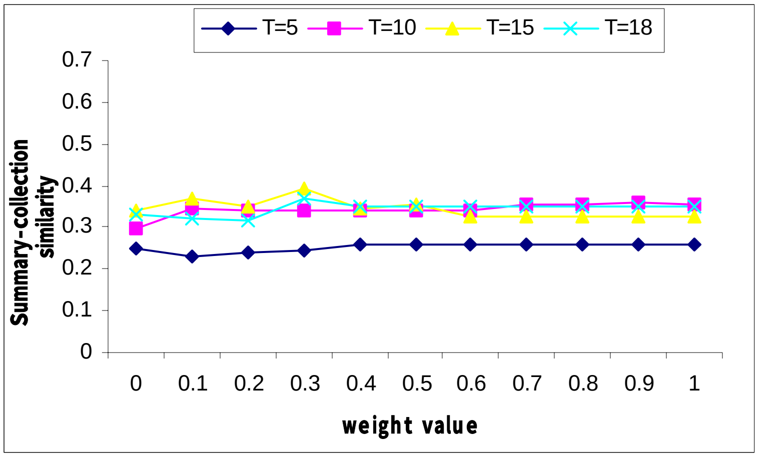

Fig. 11a. Summary-collection similarity for different slope weight values in statistical analysis method for different lengths of interval *T* for "passenger airlines" collection

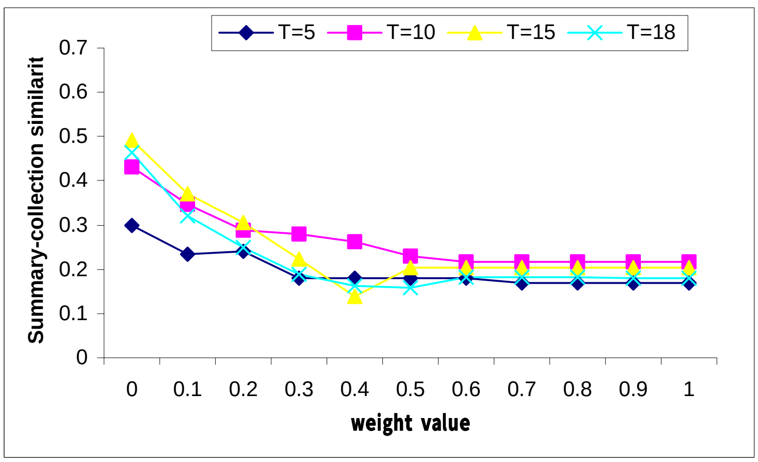

Fig. 11b. Summary-collection similarity for different intercept weight values in statistical analysis method for different lengths of interval *T* for "passenger airlines" collection

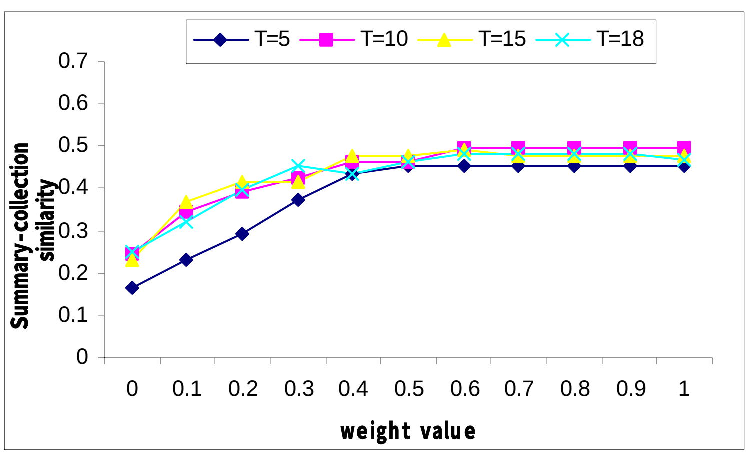

Fig. 11c. Summary-collection similarity for different variance weight values in statistical analysis method for different lengths of interval *T* for "passenger airlines" collection

Table 1. Comparison of automatic summarization with human judgment

Method	Percentage of Majority Agreement
Sliding Window	43%
Statistical Analysis	57%
Random Summary	11%
Random-Lead	12%
Term Frequency	24%

8 Conclusions

The growth of information available online and the increasing information overload have made summarization techniques more attractive. New approaches to summarization are needed for the Web due to its temporal dimension. The Web is a dynamic environment where web pages are changing frequently often in an unpredictable way. By tracking the changes in content of a topical group of web pages, we can detect popular topics or identify main changes. In this paper, we have proposed solutions for temporal multi-page summarization and discussed characteristics of temporal collections of topically related web documents. We have introduced two different methods for extractive type summarization of temporal content in retrospective web collections: sliding window and statistical analysis. In the first method we utilize spatial and temporal distributions of insertion and deletion types of changes in the collection. In the second method, statistical parameters of time series of terms are used for detecting important terms. Finally, we have presented a novel sentence selection algorithm based on local and neighboring distributions of document frequencies of terms.

References

[1] M.S. Ackerman, B. Starr and M. Pazzani, The Do-I-Care Agent: effective social discovery and filtering on the Web, in: RIAO '97: Computer-Assisted Information Searching on the Internet, Montreal, CA, 1997, pp. 17-31.

[2] J. Allan, R. Gupta and V. Khandelwal, Temporal summaries of news topics, in: Proceedings of the 24th Annual ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, USA, 2001, pp. 10-18.

[3] J. Allan (ed.), Topic detection and tracking: event-based information organization, Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[4] E. Amitay and C. Paris, Automatically summarizing web sites: is there any way around it? In: Proceedings of the 9th International Conference on Information and Knowledge Management, McLean, Virginia, USA, 2000, pp. 173-179.

[5] R. Barzilay, K. McKeown and M. Elhadad, Information fusion in the context of multi-document summarization, in: Proc. of Annual Meeting of the Association of Computational Linguistics, College Park, MD, USA, 1999, pp. 550-557.

[6] V. Boyapati, K. Chevrier, A. Finkel, N. Glance, T. Pierce, R. Stokton and C. Whitmer, ChangeDetector™: A site level monitoring tool for WWW, in: Proceedings of 11th International WWW Conference, Honolulu, Hawaii, USA, 2002, pp. 570-579.

[7] E. B. Brewington and G. Cybenko, How dynamic is the Web? in: Proceedings of the 9th International World Wide Web Conference, Amsterdam, The Netherlands, 2000.

[8] A. L. Berger and V.O. Mittal, OCELOT: a system for summarizing web pages, in: Proceedings of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval, Athens, Greece, 2000, pp. 144-151.

[9] O. Buyukkokten, H. Garcia-Molina and A. Paepcke, Seeing the whole in parts: text summarization for web browsing on handheld devices, in: Proceedings of the 10th International WWW Conference, Hong Kong, SAR, China, 2001, pp. 652-662.

[10] J. Carriere and R. Kazman, WebQuery: searching and visualizing the Web through connectivity, in: Proceedings of 6th International WWW Conference, Santa Clara, CA, USA, 1997, pp. 701-711.
[11] S. Chakrabarti, M. van den Berg and B. Dom, Focused crawling: a new approach to topic-specific web resource discovery, in: Proceedings of 8th International World Wide Web Conference, Amsterdam, Netherlands, pp. 1999, pp. 1623-1640.
[12] S. Chakrabarti, Mining the Web: discovering knowledge from hypertext data, Morgan Kaufmann Publishers, 2002.
[13] J. Cho and A. Ntoulas, Effective change detection using sampling, in: Proceedings of the 28th VLDB Conference, Hong Kong, SAR, China, 2002, pp. 514-525.
[14] J. Cho and H. Garcia-Molina, Estimating frequency of change, ACM Transactions on Internet Technology (TOIT), Vol. 3 (3), 2003, pp. 256-290.
[15] J. Dean and M. Henzinger, Finding related pages in the World Wide Web, in: Proceedings of 8th International WWW Conference, Toronto, Canada, 1999, pp. 1467-1479.
[16] J.-Y. Delort, B. Bouchon-Meunier and M. Rifqi, Enhanced web document summarization using hyperlinks, in: Proceedings of the 14th ACM Conference on Hypertext and Hypermedia, Nottingham, UK, 2003 pp. 208-215.
[17] R.L. Donaway, K.W. Drummey and L.A. Mather, A comparison of rankings produced by summarization evaluation measures, in: Proceedings of the Workshop on Automatic Summarization, Seattle, USA, 2000, pp. 69-78.
[18] F. Douglis, T. Ball, Y.-F. Chen and E. Koutsofios, AT&T Internet difference engine: tracking and viewing changes on the Web, World Wide Web, vol. 1 (1), Kluwer Academic Publishers, 1998, pp. 27- 44.
[19] D. Fetterly, M. Manasse, M. Najork and J. Wiener, A large-scale study of the evolution of web pages, in: Proceedings of the 12th International World Wide Web Conference, Budapest, Hungary, 2003, pp. 669-678.
[20] G.W. Flake, S. Lawrence and C.L. Giles, Efficient identification of web communities. Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD-2000), Boston, MA, ACM Press, 2000, pp. 150-160.
[21] L. Francisco-Revilla, F.M. III Shipman, R. Furuta, U. Karadkar and A. Arora, Managing change on the Web, in: Proceedings of ACM/IEEE Joint Conference on Digital Libraries (JCDL 2001), Roanoke, Virginia, USA, 2001, pp. 67-76.
[22] L. Francisco-Revilla, F.M. III Shipman, R. Furuta, U. Karadkar and A. Arora, Perception of content, structure, and presentation changes in Web-based hypertext, in: Proceedings of the 12th ACM Conference on Hypertext and Hypermedia, (Hypertext '01), Aarhus, Denmark, ACM Press, 2001, pp. 205-214.
[23] E.J. Glover, K. Tsioutsiouliklis, S. Lawrance, D.M. Pennock and G.W. Flake, Using web structure for classifying and describing web pages, in: Proceedings of 11th International WWW Conference, Honolulu, Hawaii, 2002, pp. 562-569.
[24] Google News, http://news.google.com
[25] The Internet Archive, http://www.archive.org
[26] J. Jacob, N. Pandrangi, A. Sanka and S. Chakravarthy, WebVigiL: An approach to just-in-time information propagation in large network-centric environments, Web Dynamics Book, Springer-Verlag, 2003.
[27] A. Jatowt, K.B. Khoo and M. Ishizuka, Change summarization in web collections, in: Proceedings of the 17th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Ottawa, Canada, 2004, pp. 653-662.
[28] A. Jatowt and M. Ishizuka, Web page summarization using dynamic content, in: Proceedings of the 13th International World Wide Web Conference, New York, USA, 2004, pp. 344-345.
[29] J.M. Kleinberg, Authoritative sources in a hyperlinked environment, Journal of the ACM, vol. 46 (5), 1999, pp. 604-632.
[30] R. Liebscher and R. Belew, Lexical dynamics and conceptual change: analyses and implications for information retrieval, Cognitive Science Online, vol. 1, http://cogsci-online.ucsd.edu 2003, pp. 46-57.
[31] C.-Y. Lin and E.H. Hovy, From single to multi-document summarization: a prototype system and its evaluation, in: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics, Philadelphia, PA, 2002, pp. 457-464.
[32] L. Liu, C. Pu and W. Tang, Continual queries for Internet scale event-driven information delivery, IEEE Knowledge and Data Engineering vol. 11 (4), Special Issue on Web Technology, 1999, pp. 610-628.
[33] L. Liu, C. Pu and W. Tang, WebCQ: detecting and delivering information changes on the Web, in: Proceedings of the 9th International Conference on Information and Knowledge Management, Washington, DC, USA, 2000, pp. 512-519.
[34] H.P. Luhn, The automatic creation of literature abstracts, IBM Journal of Research & Development, vol. 2 (2), 1958, pp. 159-165.
[35] I. Mani and M.T. Maybury (eds.), Advances in automatic text summarization, MIT Press, Cambridge MA, USA, 1999.
[36] K. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, J.L. Klavans, A. Nenkova, C. Sable, B. Schiffman and S. Sigelman, Tracking and summarizing news on a daily basis with Columbia's Newsblaster, in: Proceedings of Human Language Technology Conference, San Diego, USA, 2002.

[37] Open Directory Project (ODP), http://dmoz.org
[38] S. Pandey, K. Ramamritham and S. Chakrabarti, Monitoring the dynamic Web to respond to continuous queries, in: Proceedings of the 12th International World Wide Web Conference, Budapest, Hungary, 2003, pp. 659-668.
[39] D. Radev, S. Blair-Goldensohn, Z. Zhang and S.R. Raghavan, NewsInEssence: a system for domain-independent, real-time news clustering and multi-document summarization, in: Proceedings of Human Language Technology Conference, San Diego, USA, 2001.
[40] D. Radev, W. Fan and Z. Zhang, WebInEssence: a personalized web-based multi-document summarization and recommendation system, in: NAACL 2001 Workshop on Automatic Summarization, Pittsburgh, USA, 2001, pp. 79-88.
[41] G. Salton and M. McGill, Introduction to modern information retrieval, McGraw-Hill, 1983.
[42] Yahoo!, http://www.yahoo.com
[43] Y. Yang, T. Pierce and J. Carbonell, A study on retrospective and on-line event detection, in: Proceedings of ACM SIGIR 1998 Conference on Research and Development in Information Retrieval, Melbourne, Australia, 1998, pp. 28-36.