Change Summarization in Web Collections

Adam Jatowt, Khoo Khyou Bun, and Mitsuru Ishizuka

University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, 113-8656 Tokyo, Japan
{jatowt, kbkhoo, ishizuka}@miv.t.u-tokyo.ac.jp

Abstract. World Wide Web is not only enormous but also dynamic information space. Every day large quantity of new information is published on web pages. Many times people want to know what are the major changes in their area of interest over a given time period. This paper addresses the problem of summarizing changes in web collections devoted to a common topic. We have created a system called ChangeSummarizer, which periodically monitors a web collection in search for new changes and generates their summary. Since many web pages can be quite static over long time or just unrelated to the query we employ the method to evaluate, which pages are dynamic and which provide valuable content. Basing on this evaluation ChangeSummarizer creates web page ranking list and updates it regularly in order to improve subsequent summaries. Additionally, the system searches for new, valuable web pages, which can be included into the collection to enhance its quality.

1 Introduction

Internet has become the biggest information repository in the world. The large quantity of available data and the dynamic nature of web pages make information retrieval a challenging task. User interested in a certain topic can exploit many information resources of various nature, content and characteristics. The low cost of publishing information in WWW pages results in an unpredictable and quick changes of document contents. User can be overwhelmed with the quantity of news sources and may not be aware, which of them contain valuable and up-to-date information. Our system assists users in searching for new relevant information by providing them with the summary of recent, important changes related to specified topic.

We would also like to introduce a new research area called "change summarization" or more specifically "multi-document change summarization". The idea is to collect textual changes in related documents over certain time interval and to produce their summary. Such summary would ideally display the most important, popular changes occurring in the whole collection of web documents. There are several situations when change summarization can be of some value. Users may want to know the most important changes occurring in some domains. They can be interested in popular topics discussed in their area of interest or for example in the changes in opinions of web page authors during a specified period. Web pages in contradistinction to standard documents can change their contents unlimited number of times. We can say that changes reflect the dynamic character of a document.

Generally, a web page should be considered as a dynamic document or as an information slot where a new content can be placed in undefined time. However, it is difficult to predict the scope and time of web page changes. Usually for newswire sources one can be quite sure that fresh news will be published on a daily or weekly basis. Nevertheless, the situation can be different in case of other types of pages. The changing text of the page can have various sizes. The most extreme case happens when the whole document is deleted or a new one is created. In other cases some new information is inserted or old text is deleted in addition to some unchanged, static context. The textual changes can have various meanings however we assume that, to high extent, they are topically related to the old versions of the page and to the entire collection.

Each WWW page from the collection is periodically checked for new textual data. After comparison of new and old versions of all pages from the set, the most important terms are extracted. The system calculates scores for each term according to the popularity of the term in static and dynamic parts of the collection. Basing on the ranking list of important terms occurring within the examined period, we select sentences with the highest overall scores and present them to user. Apart from these methods we decided also to exploit the knowledge hidden in the history of each web document activity. The notion of "up-to-date-ness" function for a given web page is introduced in order to evaluate how often textual changes appear on the page and how topically close they are in comparison to the other documents. This function is based on web page dynamic scores, which are calculated every time the WWW page is examined for new changes. They specify the significance of new textual data found at this page with regards to dynamic content of other pages from the collection. ChangeSummarizer maintains a ranking list of the web pages constituting collection, which is updated after each change monitoring. Therefore web pages, which are not changing frequently or whose changes do not contain sufficiently enough popular terms will not be taken into account during generation of next summaries. Furthermore, the top 15 pages from the ranked list are periodically exploited as the base for discovery of new documents. The system searches for other related or similar web documents. After inclusion into to the collection these new sites have dynamic scores assigned as in case of the rest of pages constituting the set. In this way we aim at creating and maintaining the collection of relevant to the topic and frequently updated web pages.

In the remainder of this paper, we describe our efforts towards summarizing changes in online resources. In the next section we review some related systems and solutions. Section 3 discusses the architecture of ChangeSummarizer. In Section 4 results of our experiment are presented. We conclude in the last section and outline future research directions.

2 Related Work

In the last years several attempts have been made to design systems for developing automatic reports of important events [5], [3], [6]. Usually these applications search for popular and important topics basing on predetermined set of web pages. Google

News [3] tracks several thousands of news sources. User can issue a query and read related, recent articles. However Google News does not produce typical summaries but rather displays links to variety of sources discussing any given event. The other systems like Newsinessence [6] or Newsblaster [5] provide summaries of popular recent events, which are discussed in some chosen news sources. However, there is a need for an application that could summarize information from any types of web pages. In other words it should be a system that could produce summaries of collections of web pages, which are not limited to newswire extracts. WebInEssence [7] is an example of such application, attempting to generate summaries from dynamically constructed groups of web pages. ChangeSummarizer, on the other hand, summarizes textual changes in web collections. We focus on new, changed data in various kinds of web documents searching for common information. Additionally, our system expands web collection by searching for related web pages.

There are several systems designed for detecting and visualizing changes in WWW pages. Any user-specified features like links, text, pictures and etc. can be tracked. Usually change detection applications require a user to provide web page address to be monitored [4], [1]. The results can be sent by email as a list of changes or as a composition of different page versions for better visualization of changes. However, user is often overloaded with meaningless changes like, for example, modified syntax or color. In spite of this drawback, there was little research done on the extraction and summarization of meaningful changes from web pages.

3 ChangeSummarizer System

The conceptual architecture of our system is displayed in Figure 1. It shows the information flow in one cycle of ChangeSummarizer's performance, which takes place between downloading two consecutive versions of web collection. Each such phase is executed periodically within pre-defined time interval. Changes are examined throughout the whole tracking process, which contains some number of singular phases. The longer time the system is running, the more information can be gathered and later used as a collection history data.

The time interval between two consecutive downloads of web collection versions has an impact on overall "recall of changes" and on a single page influence on the summary. The shorter this period, the more efficient the system is in detecting short-life changes. However, usually such brief time is not sufficient enough for many web pages to change. Therefore only one or few of them will have any changes at all. In result it may happen that a single document change has relatively high impact on the final summary. The scope of this influence depends on the character of web pages constituting the collection. On the other hand, in case of the longer period one should obtain more changes, which in the end will diminish the influence of a singular web page on the summarization result. However there is a risk of losing some dynamic data during this extended time. Simply some documents may change their contents more than once during the period. In consequence, the "recall of changes" can be lower especially in the case of fast-changing resources.

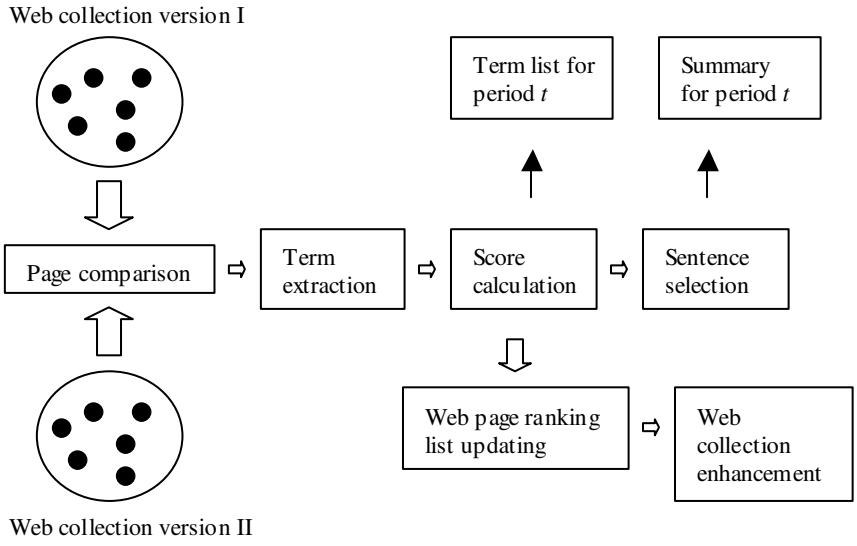

Fig. 1. Information flow in the system

3.1 Change Detection and Term Scoring

We can collect web pages relevant to user's interest in two ways: by issuing query to search engine or by using some popular web directories like, for example, Yahoo! In case of the first way, the decision of the right query words is very important since tracking results will naturally depend on the content of a web collection. Elementary base set of web pages is created by fetching first 200 hits generated by search engine in response to the user query. ChangeSummarizer checks also for duplicate pages, which must be discarded. On the other hand, the second method is a straightforward one since it utilizes already clustered and human-edited set of web documents.

Periodically or according to an arbitrarily specified time schedule, ChangeSummarizer extracts changes from the web collection by comparing old and new versions of each web page. User can specify the threshold or percentage of web pages, which will be considered for summary. We have decided to process 75% of web pages with the highest scores in the ranking list of the collection.

Next, textual data must be extracted from downloaded HTML files and converted into plain text format. Retrieval of new information (changes) is conducted by the comparison of sentences from the consecutive document versions. If a given sentence from the latest version does not appear in the previous version of the web page, then it is regarded as a new one.

In the following step, ChangeSummarizer separates words from the changed sentences and subjects them to stemming. To eliminate semantically poor words the

system conducts basic stop-list filtering. Regarding the selection of features for summary creation, we have decided to use *n*-grams, where *n* is from *1* to *3*. *N*-grams are combinations of consecutive *n* words. ChangeSummarizer calculates *n*-grams for changed parts during each cycle and also eliminates low frequency terms to reduce the feature dimension. *N*-grams are treated as separated entities in an equal way as single words.

Consequently, a weight S_i, denoting a score of "popularity" for a term *i*, is computed using the following weighting scheme.

$$S_i = \left(1 + \frac{\sum_{j=1}^{N_{doc}} \left[\frac{n_{jc}}{N_{jc}+1} - \alpha * \frac{n_{js}}{N_{js}+1} \right]}{N_{doc}} \right) * \exp\left(\frac{n_{icp}}{N_{cdoc}+1} - \alpha * \frac{n_{isp}}{N_{sdoc}+1} \right). \quad (\mathbf{1})$$

Table 1. Explanation of symbols used in Equation 1

S_i - *score for term i*	n_{isp} - *number of pages where static parts contain term i*
N_{doc} - *number of pages in the collection*	n_{icp} - *number of pages where changed parts have term i*
N_{js} - *number of static terms in page j*	N_{jc} - *number of changed terms in page j*
N_{sdoc} - *number of static documents*	n_{jc} - *number of term i in changed part of page j*
N_{cdoc} - *number of changed pages*	n_{js} - *number of term i in static part of page j*

Table 1 explains the meanings of individual symbols used in Equation 1. This equation defines the score of a term as its "popularity" in changes in the web collection. However, additionally, the score is also influenced by term's "unpopularity" throughout static parts of current web collection snapshot. Consequently, terms with high scores should appear often in changed parts of many web pages but rarely in static parts of documents. In this way, user may find not only popular terms in changes but also terms, which are unexpected since they do not occur frequently in static parts of documents. Such unexpected terms can be interesting to a user who is an expert in his area. Another motivation for this approach is that terms appearing frequently in changes may have low semantic values for a given topic. Since we want the system to be domain independent, we employ only one general stop-list. However each topic has different terms that are considered as semantically poor in given domain. Therefore we should assign higher scores to terms that do not occur frequently or are not typical words of specific domain chosen by user. Parameter α is used to specify the relative weight of such "unexpected" terms. Its range is from *0* to *1*, where the value equal to *0* indicates that there is no influence of the static part of collection on the term selection. Common terms in changed (dynamic) parts of documents are represented as the exponent of frequency of a term in those parts of web pages. For better explanation we can conceptually divide the Equation 1 into two parts. The first one describes how often a given term occurs inside each document on average. This part can have values between *0* and *2*

depending on the term distribution in every single document and on parameter α. The second part is the exponent of the term distribution among all documents in the collection. It has major impact on overall score since terms popular in many changed parts of documents should reflect common changes.

3.2 Page Up-to-date-ness and Web Collection Enhancement

Some web pages change within high frequency and have meaningful contents related to the query. Including such web documents into the collection could improve the quality of subsequent summaries. Thus it would be beneficial to select and utilize them to greater extent. In order to do so, the system creates web page ranking list. The rank of a page is based on singular dynamic scores S_t, which are assigned to the web page for every period t of consecutive cycles of the system. As was said before, periodically, we obtain a list of common weighted terms taken from all changes. Thus for a given page it is possible to describe the value of "commonness" of its dynamic content simply by summing weights of all terms and dividing the acquired sum by the number of all terms in this piece of text N_d (Equation 2).

$$D = \sum_{t=1}^{T} \frac{S_t}{(T-t+1)} \quad ; \quad S_t = \frac{\sum_{j=1}^{N_d} S_{jd}}{N_d} . \tag{2}$$

We also consider the frequency of web page changes as another type of measure of web document value. Thus we need to take into consideration the number of times that the web page changed during the whole monitoring process. Our idea is to sum the dynamic scores of the web page with respect to their dates of occurrence. The latest additions of new information indicate higher usefulness of the web page in the present moment. Therefore, the latest changes should be scored higher then changes, which occurred somewhere in the beginning or in the middle of page tracking. We are proposing an up-to-date-ness function D of a web page expressed as a sum of singular dynamic scores, whose value is decreasing along with time (Equation 2). Variable T denotes the number of time units, which passed from the beginning of the change tracking process. Web pages are sorted in the ranking list according to the value of function D. Figure 2 shows an example of the up-to-date-ness function (thick line) where the singular dynamic scores S_t of the page are represented as the starting points of thin lines.

Except for change extraction and summarization, new resources should be constantly discovered and included into the collection to enhance its quality. After the inclusion of new web pages, the scores would be computed to calculate their positions in the ranking list. Basing on this approach the consecutive summaries should be more correct and up-to-date.

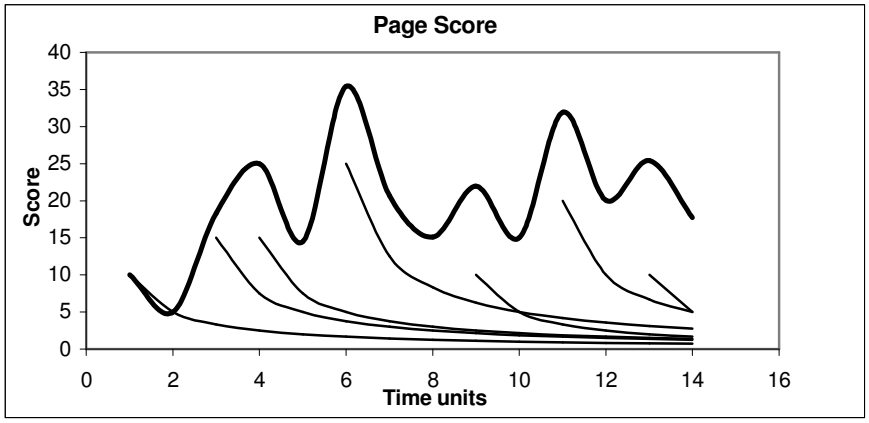

Fig. 2. Example of up-to-date-ness function

In order to automatically discover new informative web pages, ChangeSummarizer searches for web sites related to the few high-scored documents from the ranking list. Our method is a modification of Co-citation algorithm [2] in the sense that we search for resources related to several web pages simultaneously rather than only to one web page. Usually 15 web pages with highest ranks in the list are examined for their in-links. It means that the system looks for web pages which link to this selected group of documents. Then common in-links are arranged according to their frequency and 200 of the top parent pages are downloaded and stored in a separate folder. These are the documents that have the highest number of links leading to different pages from the group of 15 best resources. As a next step, the system investigates out-links from these retrieved 200 web pages searching for the frequent ones. Finally, the most common 10 links are selected. Then the web pages that are pointed to by these links are fetched and included into the collection as new resources. Consecutive change tracking phases will process these pages and use for summarization. It is important to state that only original pages should be inserted into the collection. Duplicate ones would not only influence the results but also would cause the miscounting of links during next collection updating phases. Therefore we filter duplicate pages, as well as, pages from the same sites, that means, web documents, which have identical domain names. This is due to the fact that pages within the same site may contain identical link set, which influences link counting.

3.3 Sentence Selection

In the result of change tracking, the system stores the ranked list of the most common terms together with their responding weights. To produce the final summary one needs to select representative sentences from the content of changes. ChangeSummarizer calculates the overall weight for each new sentence to pick up the ones that convey the meaning of main changes in a given period. Consequently the user is presented with the list of few most highly weighted sentences for each cycle of

the system. To increase readability we add also preceding and following sentences surrounding selected top sentences. Additionally, such extracts contain links to their host web pages to enable the user to read the whole document if necessary. Sentence scoring formula is illustrated in Equation 3.

$$S_{sen} = \frac{\sum_{i=1}^{N_{sen}} S_i}{N_{sen}} * \left(1 + \beta * \frac{N_{doc} - S_{page}}{N_{doc}}\right). \quad (3)$$

Score for a sentence is basically the sum of the scores of its all terms divided by the number of these terms. Furthermore, we make use of the historical data, which has been acquired during earlier change tracking phases. Thus we modify the sentence score by the page weight of the document, where a given sentence was published. The symbol S_{page} indicates the web page position in the ranking list of the collection. Parameter β corresponds to the strength of the "historical data" influence and has a range from *0* to *1*.

As was mentioned before, 75% of the highest scored web pages are utilized for summarization. Nevertheless, ChangeSummarizer continually computes scores for the remaining pages in the lower positions of the list. Thanks to that, there is a chance to include them for forthcoming summaries provided that their characteristics will improve.

4 Results

We present the results from the experiment conducted for the query "latest movies". The collection was obtained after issuing the query to a search engine and fetching the top 200 web pages. Change tracking was performed several times during period from 18th February to 12th May. Table 2 displays top-scored terms obtained during 4-days interval from 26th to 29th April 2003. Due to limited space we present only the top-scored terms for one time interval.

On the 25th April two popular movies were released such as "It runs in the family" and "Identity". The first one is about the life of three generations of New York's family with Kirk and Michael Douglas starring. The later one shows the sequence of mysterious murders committed in motel during one night. The main actor in this movie is John Cusack.

We show three sets of highest-scored terms for different values of parameter α. Along with the increasing value of α terms having rather general meaning such as: "movie", "film" or "make" are descending to lower ranks in the list. On the other hand rare or specific to the above movies terms like: "douglas", "motel" and "identity" have higher relative scores.

Table 3 displays top sentences for different time intervals for parameter α value equal to *1*. Basically they provide information or comments on new movies, which have been or are going to be released in the nearest time. Given the diversity of types of pages and topics related to the query, it should not be surprising that the final results may not constitute coherent summary. Intuitively, it is very important to

construct appropriate web collection with closely related web pages. We have noticed that the system produces better results for narrow topics, where documents tend to be more topically related. In the collection except for several newswire resources or web sites solely devoted to topic, there have been found also some message, opinion boards and "blog" type pages. Although, such pages can blur the final results, they provide information about opinions or hot topics in contradistinction to information about rather pure facts presented by newswire sources.

Table 2. Top terms for different parameters α

Parameter α = 0		**Parameter α = 0.5**		**Parameter α = 1**	
movi	1.649009	star	1.395895	star	1.306643
new	1.564911	new	1.354146	man	1.247069
star	1.485302	movie	1.320918	family	1.245389
film	1.435972	like	1.315999	douglas	1.23617
like	1.412125	man	1.293055	like	1.21999
just	1.362831	family	1.280962	girl	1.215398
time	1.339155	girl	1.265839	run	1.207945
man	1.339055	just	1.25945	murder	1.172834
family	1.316553	film	1.255065	college	1.166521
want	1.316437	douglas	1.243223	dark	1.165882
girl	1.316373	run	1.239737	start	1.165546
make	1.295729	year	1.225889	motel	1.160303
look	1.294302	want	1.209694	year	1.158099
year	1.293781	start	1.208024	just	1.156253
run	1.271534	make	1.20319	cusack	1.14825
start	1.250516	murder	1.190517	John cusack	1.148215
play	1.250359	dark	1.187093	identity	1.14806

Table 3. Final sentences

Top sentences with following and preceding sentences	**Period**
"It was the first time three generations of one family have been in a picture," says the elder Douglas. The father and son star in the movie, along with Michael's son Cameron, 24, and Michael's mother, Diana Douglas, long divorced from Kirk. Sitting in the elder Douglas' elegantly appointed one-story home, Kirk, 86, and Michael, 59, clearly have a warm relationship.	26/4-29/4
The result is "Ghosts of the Abyss," an hour-long triumph of documentary filmmaking and a new high-mark in the director's already fabled career. Rock star Rob Zombie makes his debut as a feature film writer and director with "House of 1000 Corpses," opening this weekend at Cinema World in West Melbourne. Zombie's only previous feature experience was an animated sequence in "Beavis & Butt-head Do America," though he has directed several music videos.	4/4 - 12/4

James Cameron mixes CG and 3D. Dark Horizons reports that the Titanic director's next film will be in the same vein as Avatar, which was going to be the first film with total CG actors in it. He said it wouldn't be as big scale as that, but would have some CG characters in it.	16/4-26/4

5 Conclusions and Future Work

We have introduced a new research area of summarizing textual changes in web page collections and have presented a complete, cyclical system called ChangeSummarizer. The system uses novel methodology for extracting and summarizing textual changes in web collections. Summarizing changes is based on searching for common and semantically rich content terms. ChangeSummarizer maintains ranking list of web pages, where each page is scored according to the frequency and the contents of its changes. In this way historical data can be gathered and later exploited for the summarization purposes. The most valuable web pages, according to this measure, are utilized for consecutive summaries. Additionally they form a base for finding new web pages to be included into the collection.

Our system has several limitations, which we want to focus on in the future. One problem concerns changes in the form of old and new links found in web documents. If web pages linked by a certain document are found on the same web site as this document then we should also consider their changes. Moreover, since new sentences can be placed in different semantic contexts on a web page, one should take into consideration not only the content of a changed sentence but also its relation to the surrounding text.

References

1. Changedetect: http://www.changedetect.com/
2. Dean, J., Henzinger, M.: Finding Related Pages in the World Wide Web. In Proceedings of The Eighth International World Wide Web Conference. Toronto Canada (1999) 1467-1479
3. Google News: http://news.google.com
4. Liu, L., Pu, C., Wang, T.: WebCQ: Detecting and Delivering Information Changes on the Web. In Proceedings of International Conference on Information and Knowledge Management. Washington, DC, USA (2000) 512-519
5. McKeown, K., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J.L., Nenkova, A., Sable, C., Schiffman, B., Sigelman, S.: Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster. In Proceedings of Human Language Technology Conference. San Diego, USA (2002)
6. Radev, D., Blair-Goldensohn, S., Zhang, Z., Raghavan, S.R.: NewsInEssence: A System for Domain-Independent, Real-Time News Clustering and Multi-Document Summarization. In Human Language Technology Conference. San Diego, USA (2001a)
7. Radev, D., Fan, W., Zhang, Z.: WebInEssence: A Personalized Web-Based Multi-Document Summarization and Recommendation System. In NAACL 2001 Workshop on Automatic summarization. Pittsburgh, USA (2001b) 79-88