Visualizing Changes in Coordinate Terms over Time: An Example of Mining Repositories of Temporal Data through their Search Interfaces

Hiroaki Ohshima, Adam Jatowt, Satoshi Oyama and Katsumi Tanaka
Kyoto University
{ohshima, adam, oyama, tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract

Certain data repositories provide search functionality for temporally ordered data. News archive search or blog search are examples of search interfaces that allow issuing structured queries composed of arbitrary terms and selected time constraints for performing temporal search. However, extracting aggregated knowledge such as detecting the evolution of objects or their relationships through these interfaces is difficult for users. In this paper, we discuss the problem of knowledge extraction and agglomeration from repositories of temporal data. In particular, we propose a method for detecting and visualizing changes in coordinate terms over time based on a news archive.

1. Introduction

Knowledge extraction from the Web has been a popular research area over the past few years. The majority of the proposed approaches relied on collecting some portion of the Web and storing it locally for subsequent mining. For example, blogs could have been crawled over a certain portion of time to be later processed for extraction of interesting patterns or detection of topical communities. This approach, however, presents many obstacles such as the limited amount of stored data and extracted knowledge, high storage cost, and vulnerability to data obsoleteness. Other approaches to mining large repositories rely on utilizing public search interfaces for sampling the stored data [5,9,18]. For example, Bollegala [5] recently proposed several efficient measures for estimating the strengths of semantic similarities between arbitrary terms based on the WebCount values (number of search results on the Web) and several lexico-syntactical patterns extracted from returned snippets.

Certain search interfaces allow for constructing temporally-constrained queries. For example, many blog search engines let users enter text with an option for selecting a time scope over which the search is going to be performed. The returned results should be then composed of only those documents that were created (or were valid) within the specified time period. In this paper, we propose utilizing such search interfaces in order to detect various kinds of temporal knowledge, thus, extending the process of knowledge extraction into temporal dimension. This kind of a temporal mining of repositories through their search interfaces makes it possible to obtain different kinds of temporal knowledge, such as the evolution of certain relations or changes in the popularity of expressions over longer time periods. Obviously, it is difficult to extract this kind of knowledge manually, as it involves issuing series of unit queries directed to different time periods, as well as agglomerating the extracted results over time.

In [18] we have demonstrated an application that detects and visualizes coordinate terms to user-issued queries. Coordinate terms are terms that are bound by hypernym relationships. In many cases, such terms represent peer (or rival) relationships between real world objects. We propose an extension of the coordinate term detection for determining and visualizing the changes in coordinate terms over time. The application that we propose queries a news archive search public interface in order to extract partial data and agglomerate it for a more general knowledge of temporal character. We believe that the proposed system could be used for educational purposes and can also facilitate understanding of the state-of-the-art knowledge (i.e., current hypernym terms). For example, a user may enter term “Yahoo” to discover terms coordinate to it, such as, peer (or rival) companies Google[1] or Microsoft[2]. Next, by

[1] Google Search Engine: http://www.google.com
[2] Microsoft: http://www.microsoft.com

978-0-7695-3300-1/08 $25.00 © 2008 IEEE
DOI 10.1109/INGS.2008.13

analyzing the visualized past coordinate terms she or he may observe other entities that were valid in the past and may also find out the age of coordinate relationships that currently apply. In result, the user is provided with a kind of temporal support to better understand the current relations.

The remainder of this paper is as follows. In the next section, we discuss the related research. Section 3 provides the larger background of knowledge discovery through search interfaces. Section 4 describes our application for detecting and visualizing coordinate terms over time. Section 5 provides the discussion of some aspects related to the proposed approach. We conclude the paper in the last section and outline the directions for our next research.

2. Related research

2.1 Mining content of repositories via their search interfaces

Large text corpora have been successfully used for knowledge extraction since long time. For example, Hearst [12] proposed an approach for hyponymy relation discovery from unrestricted text collections. Since a free access to large content repositories is often impossible, several researchers began to seek effective ways for mining data collections through their search interfaces [5,9,18]. Bollegala et al. [5] described an approach for measuring similarity between arbitrary terms based on the number of search results returned from Web search engines and the content of snippets. Cilibrasi and Vitányi [9] proposed a semantic distance measure called Google Normalized Distance between query terms based on WebCount values. The authors have successfully used it for various knowledge-intensive tasks such as hierarchical clustering, classification or language translation.

Recently, Yamamoto et al. [22] demonstrated an application for visualizing evolution of the popularity of facts in the Web. First, their system detects counter expressions for a given factual phrase about which a user is unsure. Next, the changes over time in the popularity of all the counter expressions are analyzed to determine those ones that feature increasing or decreasing popularity. Users receive results in the form of interactive charts that portray the relative popularity distributions of the phrases over time. This supports the user judgment of fact trustworthiness.

Several research attempts [3,4] have been made in order to estimate various quality metrics of search engines through frequently querying their public interfaces. By statistically analyzing returned search results it becomes possible to evaluate search engine repositories from the viewpoints of their freshness levels, content coverage or the overlap with respect to other search engines. A similar kind of statistical analysis may be necessary to estimate the temporal characteristics and distribution of content in repositories containing temporally structured data.

Detecting co-ordinate terms from document collections has been already proposed before. For example, Shinzato and Torisawa [19] proposed a method to acquire coordinate terms from HTML documents focusing on HTML structure. Terms in the same level structure such as itemized terms in a list can become candidates for coordinate terms. Ghahramani and Heller [11] proposed using Bayesian Sets to acquire coordinate terms. Their method finds clusters of terms based on Bayesian inference. The algorithm is simple, fast and it needs external data sets.

2.2 Temporal text mining

Temporal text mining is a research devoted to analyzing and mining streams of text data [1,16,17,21]. Topic Detection and Tracking (TDT) [1] was a popular research challenge in this area aimed at detecting, classifying, and tracking events and coherent topics in news corpora. The five main challenges in TDT were story segmentation, topic detection, topic tracking, first story detection and story link detection.

Although TDT initiative appears to be no longer active, there are many recent research achievements along this line. For example, Wang and McCallum [21] identified topics persisting over dynamic collections of documents. Another work presented the development of topic patterns in news articles over time [17]. A probabilistic model for retrospective event detection in news corpora was introduced in [16].

From a more fine-grained viewpoint, several methods for temporal weighting of terms have been proposed for document collections of temporal character. A detailed survey of these can be found in [15]. The most popular weighting measure is the one based on a threshold level which classifies features as important ones if their frequencies are higher than pre-defined threshold levels [20]. On the other hand, Kleinberg [14] proposed a state-based approach to measure temporal term importance using transitions between two states, low and high frequency one.

Although the objective of our research is similar to temporal text mining, the main difference is the way in which collections are utilized. Traditional approaches benefited from the unrestricted access to data, while we focus on the knowledge generation using limited data retrieved from search interfaces of available data collections. This way of access presents several interesting research problems that need to be approached.

2.3 Visualizing network evolution

Visualizing network evolution is another related research area. Detecting and visualizing changes in the structure of networks over time presents significant challenges [7] such as the need for facilitating change detection or for preserving the overall context of changes. One way to portray the evolution of graph structure is to display the chronological sequence of graph snapshots in multiple, separate planes [6,10]. Another method relies on animation effects thanks to which users can passively or actively observe the consecutive stages of network evolution [8,7]. In the application described in this paper we have decided to utilize the latter visualization technique. This assures a fine granularity demonstration of graph changes over time and gives users several interaction possibilities. In order to decrease the cognitive load placed on the users we propose also several implementation solutions. In addition, a condensed overview of graph history is also provided.

3. Background

The current Web contains documents created at different time periods; hence, mining the current Web does not ensure that one can obtain the fresh (currently valid) results. Bar-Yossef et al. [2] discovered that the Web is actually polluted by numerous abandoned Web pages. Thus, by mining the Web using temporal constraints could result in higher probability that the returned results are up-to-date.

Certain Web search engines (e.g., Google) provide a kind of “date range” option. Users can issue a standard textual query together with the date range constraint in order to specify the temporal constraints of the search. However, in majority cases, these constraints define only time spans in when pages were crawled or indexed for the first time (or for the last time) by search engines rather than the exact origin time of their content. As a consequence, content may be considered older (or younger) than it actually is, as it could have been introduced into the pages some time later (or before) its first (or last) encounter by a search engine crawler. However, if search engines would store timestamps of any content changes their crawlers encountered, then this would prohibitively inflate the sizes of their indices.

Recently, Jatowt et al. [13] proposed a solution to content age detection problem in Web documents based on an online search performed on data of publicly available Web archives. This approach, however, suffers from relatively high time cost and, thus, cannot be used for a large scale purposes such as agglomerating information from multiple pages.

Thus, to the best of our knowledge, currently, there are no search engines that would allow for a reliable temporal search in the Web. Nevertheless, we believe that an additional filtering and aggregation of results from multiple Web search engines might push the quality of results to the level at which they could be utilized for temporal search and knowledge extraction. We leave it, however, for future consideration.

On the other hand, many smaller controlled repositories reliably provide content that was created or valid within arbitrary time frames. For example, news articles or blog posts usually have explicit timestamps which inform about their origin dates. Thus, many repositories containing news articles or blogs can be successfully mined in a longitudinal way. Often, the data stored in such repositories has been actually crawled from the Web, as it is in the case of Google News Archive[3] repository. Google News Archive collection stores news articles that were obtained from about 4500 online newswire sites. Since the stored content is highly reliable, and, at the same time, it is representative for the real world events and objects, thus, mining it should deliver trustfull and accurate data.

4. Application for visualizing coordinate terms over time

We demonstrate here an interactive application for visualizing the changes in terms that are coordinate to a given query term. The application allows users to drag the slide bar in order to view the coordinate terms in consecutive time periods. It works as follows: After receiving a query the system issues a series of converted query patterns to a search interface with a fixed temporal granularity. Each such query is performed over a unit time segment (fixed window). The time segments are mutually exclusive. The top *k* results for each time segment are then

[3] Google News Archive Search: http://news.google.com/archivesearch

analyzed for the occurrence of given lexico-syntactical patterns in order to detect coordinate terms. Lastly, the results from different time periods are agglomerated and visualized to users.

4.1 Detection of coordinate terms

In this section we describe the method of detecting coordinate terms in more detail. We have chosen Google News Archive as an underlying repository of temporal data. The whole time period of analysis was set from 2000 to 2007. The granularity of time segmentation is decided by a user. When the user issues a given query, *q*, the system converts it to a specially crafted "pattern query" that is sent to a search engine within the series of consecutive time units. The pattern query is composed of a phrase "*q* or" and "or *q*". For each unit time period, the snippets[4] of the top 100 returned results are analyzed for the occurrence of both patterns. The system searches for the terms that frequently appear before or after the pattern query depending on the order of conjunctive "or" (i.e., "*q* or *x*", "*x* or *q*"). Such terms are deemed to be coordinate terms (*x* in the previous example). Often they represent peer-level relationships, for example, the countries in Scandinavia – Norway, Sweden or Finland or the companies offering similar services – Microsoft, Yahoo![5] or Google. In addition, we have also implemented a pattern query "*q* vs." and "vs. *q*" in order to provide more refined results of rival-type relationships. The reason why we use bi-directed patterns is that this allows for determining the exact phrases which constitute coordinate terms [18]. Stop words are also eliminated from the candidate pool of coordinate terms.

We provide also an option for users to force a given context for the query term. For example, users can search for coordinate terms to Japan in the context of soccer. Such a query is created by adding the context word (e.g., soccer) to the pattern query.

4.2 Detection of temporal context of co-ordinate relationships

In addition, the system detects the context of coordinate relationships. Real world objects can be peers (or rivals) within their different contexts, for example, countries like India and China can be listed as Asian countries, most populous countries or emerging economic powers. We determine the coordinate relationship context by detecting the most often co-occurring keywords inside snippets containing the same coordinate terms. For this purpose, Jaccard coefficient is used.

$$JC(a, p_b; t_i) = \frac{M(a, p_b; t_i)}{M(a; t_i) + M(p_b; t_i) - M(a, p_b; t_i)} \qquad (1)$$

Here $M(a,\ p_b;\ t_i)$ is the number of snippets that contain both the lexical pattern p_b with a given coordinate term *b* and a term *a* within a time period t_i. $M(a;\ t_i)$ and $M(p_b;\ t_i)$ denote the count of snippets that contain the term *a* and the count of snippets that contain p_b, respectively. The three terms that have the highest values of the Jaccard coefficient are chosen as the context of the co-ordinate relation between the terms *a* and *b* within t_i. The Jaccard coefficient ensures that the final result is not biased to the unusually high level of occurrence of any term. Thus, in other words, it captures non-casual association between two entities. Note that the Jaccard coefficient does not have to be used when searching for coordinate terms as there is already a semantically explicit relation between terms enforced by the lexical patterns "or" and "vs.".

4.3 Visualization

The results are visualized in the form of an animated graph whose changes over time can be interactively explored (Figure 1). In the graph, nodes represent terms and edges indicate coordinate relationships between the terms. Users can drag the horizontal slider in order to see the coordinate terms within different time periods. The granularity of the unit time spans can be chosen by the users (1 year is default). At any time point, users can click on arbitrary nodes in the graph in order to send their terms as queries and trigger the detection and visualization of terms that are coordinate to them. In this way, the graph can be arbitrarily expanded at any time.

Several implementation solutions have been made in order to support the change awareness and, thus, limit the cognitive burden posed on users. First, nodes representing query terms are coloured in orange. The nodes that represent coordinate terms are shown relatively near each other. We have used a spring type graph where relative positions of nodes are determined by two forces acting between the nodes, attraction and repulsion. The attraction force occurs only between nodes that represent coordinate terms (nodes with an edge), while the repulsion force

[4] We have chosen snippets rather than the whole documents for efficiency.

[5] Yahoo!: http://www.yahoo.com

acts between every node in the graph. The strengths of the both forces depend on the distance between the nodes. When there is a coordinate relationship between any two nodes, then both the nodes converge to the equilibrium position. This position is determined by the situation in which the total repulsion and attraction forces of both nodes have equal values[6]. On the other hand, when there is no edge between the nodes, the nodes are pushed apart to certain distance after which there is no longer any force applied. Users can interact with the network at any time by changing positions of any selected nodes.

To facilitate the awareness of changes, the nodes that are absent from period t_i while present at t_{i-1} are shown in gray colour (Figure 3). On the other hand, the newly added nodes are coloured in red. This helps users to become more aware of changes in relationships over time. In addition, the nodes that become coordinate in a given time period move towards each other until they are stopped in their equilibrium positions in which they are bound by edges. Thus, users can more easily spot the changes in graph structure between the consecutive time segments. In contrast, the nodes that cease to represent coordinate terms are pushed away from each other and do not have any edges.

In addition, the width of a given edge indicates the strength of the coordinate relationship between terms represented by the end nodes of the edge. It is calculated as the frequency of expressions containing both terms within selected lexical patterns. Lastly, the detected context of each relationship is shown as an edge annotation containing top three context words of that relationship.

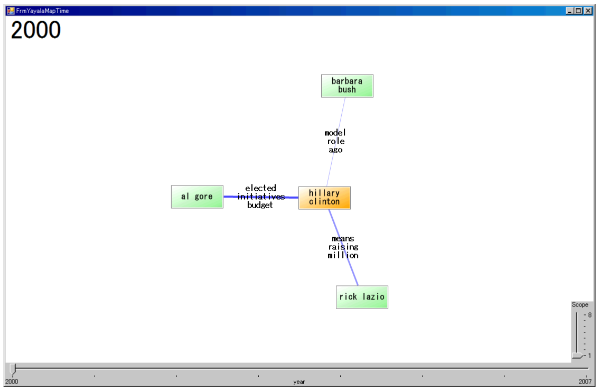

Figure 1 Visualization of the coordinate terms relationships (query “hillary clinton”, year 2000).

4.4 Summarized history view

The above visualization portrays the fine grained changes in the coordinate terms over time to the scope determined by granularity levels selected by users. In addition, we also display a summarized historical graph of coordinate terms (Figures 6 and 7). This kind of a static visualization provides a user with a quick overview of the main coordinate terms to the query term inside the whole specified time period. The width of edges indicates the strength of relationships over time. It is calculated as the average strength of the relationship values over the *N* number of unit time segments.

$$S_a(b;T) = \frac{1}{N}\sum_{i=1}^{i=N} M(a,b;t_i) \qquad (2)$$

Each edge is also annotated with its main relationship context. This relationship context is the top average context over the whole time period of analysis.

The thickness of the frames of nodes indicates the age of coordinate relationships. We count the age of coordinate term from its oldest appearance in the graph. The thick frame denotes terms that have been coordinate to a given term for a relatively long time, while thin frame denotes terms that recently have become coordinate.

The summarized view of coordinate terms over time is useful for comparing the historical coordinate terms with the ones that are detected from the recent data. Users could better understand the coordinate relationships that currently apply and discover the ones that are no longer valid.

5. Discussion

5.1 Time segmentation

The size of a unit time segment determines the precision of the extracted data. The shorter the temporal distance between upper and lower temporal constraints, the more accurate the result is. Naturally, there is a trade-off between the cost and the precision, since the high number of issued queries increases the burden on the repository, resulting often in time delays. On the other hand, when the unit time segment is long, the number of queries necessary to be submitted is low. However, in the latter case, some relatively short-lived events may remain undetected due to the smoothing effect of search results from longer time periods.

To alleviate the problem of granularity-precision, an adaptive kind of segmentation could be used, in which the time window changes its size according to

[6] The length of the distance between the nodes in the equilibrium state is predefined by a system designer.

the character of data or returned obtained results. For example, the length of the unit time segment could be decreased for those time scopes for which there is a high expectation of the occurrence of interesting events, while staying unchanged for the remaining time spans. Naturally, in order to agglomerate the results of such a querying strategy, one has to normalize them considering the different sizes of the time segments. In our example application, we have decided to use a fixed size time segment for the benefit of simplicity. Note that the optimal length of a unit time segment usually depends on the character of the mining task and user expectations, and thus is difficult to be determined automatically. Therefore, we have decided to let users choose the required granularity levels. Our future research will go into direction of an automatic detection of optimal time segments and a more passive viewing style of evolution similar to a slideshow mode.

5.2 Normalization of search results from different time periods

In general, if the results of mining or search over different time periods are to be agglomerated together, then, often some sort of normalization may have to be performed. This is because the amount of data assigned to different time segments can differ and users may not know the exact distribution of data in repositories. For example, it is obvious that the Web has been constantly growing since its origin, thus, search engine repositories contain uneven amounts of data collected from the Web in different time periods. However, the exact growth pattern of the Web remains unknown. In addition, the crawling rate of data may have been changing in the past being subject to certain fluctuations.

A simple solution to the above problem might be to use a pool of time-invariant and common words for estimating data distribution in collections for the normalization purpose. Such terms should be generally common inside the content of the repository and their occurrence frequencies should be independent of the time flow. Stop words usually fulfil these requirements; although, some stop words are actually temporal expressions, such as May, 2008, and, thus, need to be filtered out.

Note that this kind of normalization also depends on the mining task and the character of the data stored in repositories. Since in our application we do not use the raw number of search results but just analyze the top 100 returned snippets, hence, the normalization by the data size within different time periods is of lesser importance here. Also, in the case of the context terms' detection there is no need for the normalization as the Jaccard coefficient provides already a normalization effect.

Lastly, we need to note that our approach is also biased towards top-ranked documents. This is still fine as long as the ranking function of a search engine works in a consistent way over all chosen time periods.

6. Experiments

In this section, we demonstrate some examples of the results that the proposed system can deliver. Figure 2 shows the coordinate terms to a query term "hillary clinton" for the year 2004, while Figure 3 displays the results for the same query for year 2007. One can notice the strong coordinate relation between "hillary clinton" and "barack obama" for the year 2007. This relation is because both the politicians are candidates for Democratic Party nominations in 2008 U.S. Presidential Elections. Rudy Guliani ran for Republican Party nomination for the election but withdrew in January 2008. John Edwards resign from candidating at the same time. In Figure 1 we can also see the coordinate terms of "hillary clinton" in 2000. At this time, Hillary Clinton was the first lady of USA to candidate to U.S. Senate Elections against Rick Lazio. She was preceded by Barbara Bush as the first lady.

In the second example, we have input queries "yahoo" and "google" for years 2000 (Figure 4) and 2006 (Figure 5). We can see that Altavista[7] was a rival company to both Google and Yahoo! while Hotbot[8] was a rival to Google in 2000. On the other hand, in 2006, we can observe Apple[9] and Youtube[10] as new rivals to Google and Microsoft as a strong rival to both Google and Yahoo.

Figures 6 demonstrates historical summary view for a query "hillary clinton" and Figure 7 shows same view for queries "yahoo" and "google".

[7] Altavista: http://www.altavista.com

[8] Hotbot: http://www.hotbot.com

[9] Apple: http://www.apple.com

[10] Youtube: http://www.youtube.com

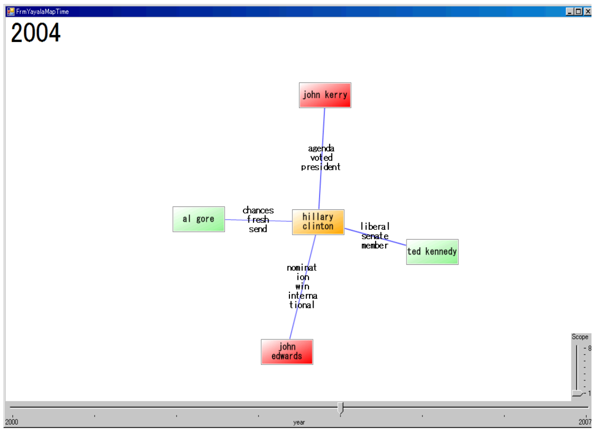

Figure 2 Coordinate terms to the query "hillary clinton" for 2004.

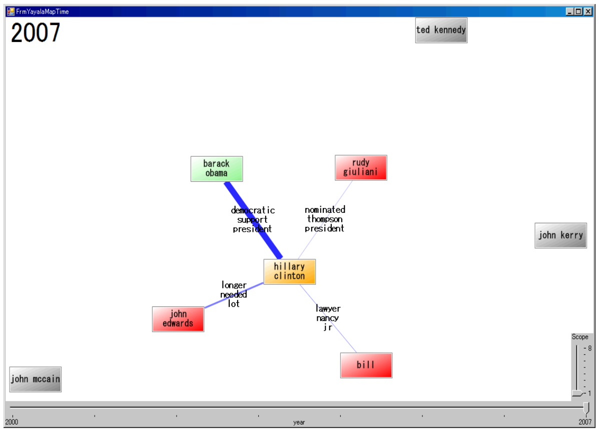

Figure 3 Coordinate terms to the query "hillary clinton" for 2007.

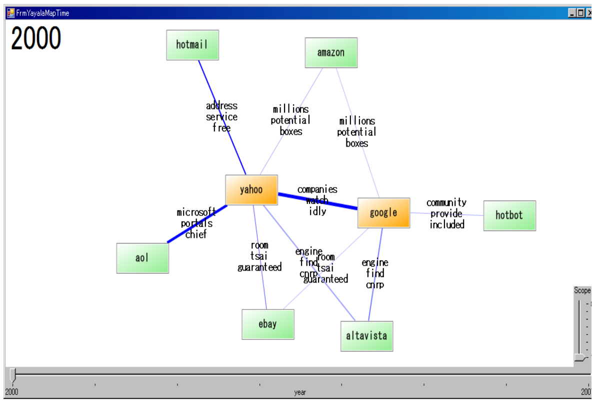

Figure 4 Coordinate terms to the queries "yahoo" and "google" for 2000.

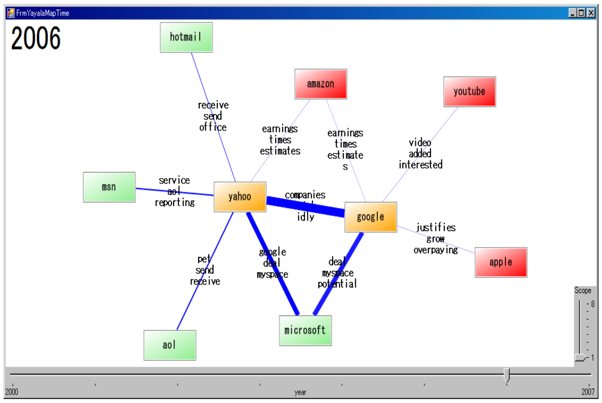

Figure 5 Coordinate terms to the queries "yahoo" and "google" for 2006.

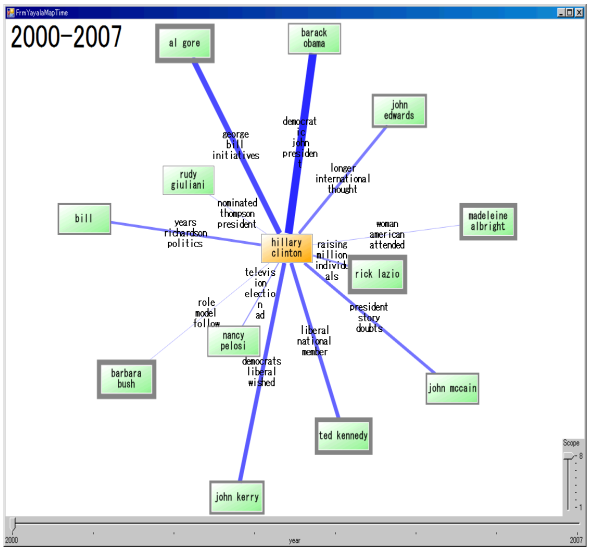

Figure 6 Historical summary view of coordinate terms to the query "hillary clinton".

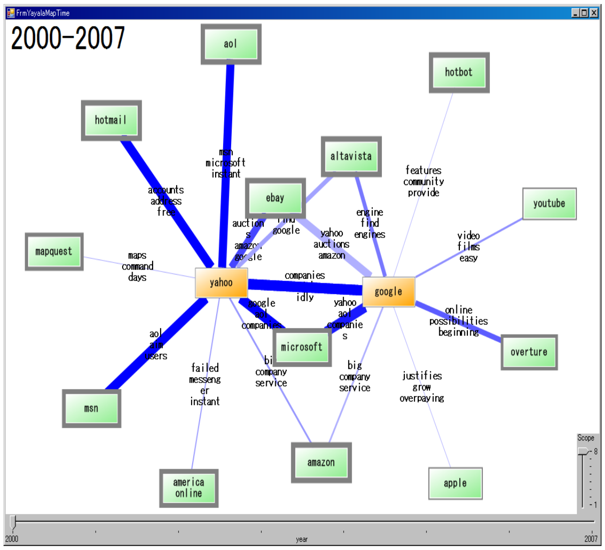

Figure 7 Historical summary view of coordinate terms to the queries "yahoo" and "google".

7. Conclusions and future work

Recently, mining the Web or other large repositories via their public interfaces has become an attractive research area. In this paper, we have proposed mining search engine interfaces that allow for temporally structured queries in order to extract knowledge of temporal character. We have demonstrated an example application for interactively tracking the changes in coordinate terms over time. Using the proposed system users can discover terms that have high probability to represent rival objects for a given query and a specified time span. This kind of historical knowledge can serve educational purposes and can support understanding of the present relations between terms.

In the future, we plan to conduct large scale experiments and to provide a more general framework for mining temporal data behind search interfaces. We also would like to implement other lexico-syntactical patterns in order to portray the evolution of different kinds of relationships. Lastly, the proposed application could be improved by adding functionality for passive watching style, proposing solutions for a more precise detection of salient relationships over time and providing efficient means to contrast the historical relations with the ones that currently apply.

8. Acknowledgments

This work was supported in part by Grants-in-Aid for Scientific Research (Nos. 18049041, 18049073, 18700111, 19700091) from MEXT of Japan, a MEXT project entitled “Development of Fundamental Software Technologies for Digital Archives,” a Kyoto University GCOE Program entitled “Informatics Education and Research for Knowledge-Circulating Society,” and a Microsoft IJARC CORE4 project entitled “Toward Spatio-Temporal Object Search from the Web.”

9. References

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

[2] Bar-Yossef, Z., Broder, A. Z., Kumar, R., and Tomkins, A. Sic transit gloria telae: towards an understanding of the web's decay. WWW 2004, 328-337.

[3] Bar-Yossef, Z., and Gurevich, M. Efficient search engine measurements. WWW 2007, 401-410.

[4] Bar-Yossef, Z., and Gurevich, M. Random sampling from a search engine's index. WWW 2006, 367-376.

[5] Bollegala, D., Matsuo, Y., and Ishizuka, M. Measuring semantic similarity between words using web search engines. WWW 2007, 757-766.

[6] Brandes, U., and Corman, S. R. Visual unrolling of network evolution and the analysis of dynamic discourse? Information Visualization 2(1), 40-50 (2003).

[7] Chen C. Information Visualization. Springer, 2006.

[8] Chen, C., and Morris, S. Visualizing Evolving Networks: Minimum Spanning Trees versus Pathfinder Networks. INFOVIS 2003.

[9] Cilibrasi, R., and Vitányi, P. M. B. The Google Similarity Distance. IEEE Trans. Knowl. Data Eng. 19(3): 370-383 (2007).

[10] Erten, C., Harding, P. J., Kobourov, S. G., Wampler, K. and Yee, G. Exploring the computing literature using temporal graph visualization. Proceedings of SPIE, Volume 5295, Visualization and Data Analysis, 2004.

[11] Ghahramani, Z. and Heller, K. Bayesian Sets. NIPS 2005.

[12] Hearst, M. A. Automatic Acquisition of Hyponyms from Large Text Corpora. COLING 1992, 539-545.

[13] Jatowt, A., Kawai, Y. and Tanaka, K. Detecting Age of Page Content. Proceedings of the 9th ACM International Workshop on Web Information and Data Management, Lisbon, Portugal, 2007, 137-144.

[14] Kleinberg, J. Bursty and hierarchical structure in streams. Data Mining Knowledge Discovery, 7(4), 2003, 373-397.

[15] Kleinberg, J. Temporal dynamics of on-line information streams. In Data Stream Management: Processing High-Speed Data Streams, (Garofalakis, M., Gehrke, J., Rastogi, R., eds.), Springer, 2005.

[16] Li, Z., Wang, B., Li, M. and Ma, W.-Y. A probabilistic model for retrospective news event detection. In Proceedings of the 28th Annual International Conference on Research and Development in Information Retrieval, Salvador, Brazil, 2005, 106-113.

[17] Mei, Q. and Zhai, C-X. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In Proceedings of the 11th International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 198-207.

[18] Ohshima, H., Oyama, S., and Tanaka, K. Searching Coordinate Terms with Their Context from the Web. WISE 2006, 40-47.

[19] Shinzato, K. and Torisawa, K. A Simple WWW-based Method for Semantic Word Class Acquisition. RANLP05 2005, 493-500.

[20] Swan, R. and Allan, J. Automatic generation of overview timelines. In Proceedings of the 23rd Conference on Research and Development in Information Retrieval, Athens, Greece, 2000, 49-56.

[21] Wang, X. and McCallum, A. Topics over time: a non-Markov continuous-time model of topical trends. In Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, 2006, 424-433.

[22] Yamamoto, Y., Tezuka, T., Jatowt, A., and Tanaka, K. Honto? Search: Estimating Trustworthiness of Web Information by Search Results Aggregation and Temporal Analysis. APWeb/WAIM 2007, 253-264.