Seeing Past Rivals: Visualizing Evolution of Coordinate Terms over Time

Hiroaki Ohshima, Adam Jatowt, Satoshi Oyama, and Katsumi Tanaka

Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, 606-8501
Kyoto, Japan
{ohshima,adam,oyama,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract. In this paper, we describe an approach for detection and visualization of coordinate term relationships over time and their evolution using temporal data available on the Web. Coordinate terms are terms with the same hypernym and they often represent rival or peer relationships of underlying objects. We have built a system that portrays the evolution of coordinate terms in an easy and intuitive way based on data in an online news archive collection spanning more than 100 years. With the proposed method, it is possible to see the changes in the peer relationships between objects over time together with the context of these relationships. The experimental results proved quite high precision of our method and indicated high usefulness for particular knowledge discovery tasks.

1 Introduction

Automatically extracting knowledge from large textual data collections has been a popular research area for quite a long time. Researchers captured different kinds of knowledge such as semantic relationships between terms, news or any temporal patterns in text. The usual approach was to scan the stored data in search for evidences supporting particular knowledge patterns. However, the access to many collections is limited through online search interfaces due to their large size or proprietary character. Consequently, the type of knowledge that could be obtained from such repositories is somewhat limited. Therefore some researchers have recently proposed knowledge extraction from online repositories by multiply querying them through their search interfaces. For example, Bollegala et al. [2] demonstrated semantic similarity estimation between arbitrary terms that takes into account the number of search results and the content of snippets returned from Web search engines.

This kind of approach, however, can be directly applied to the collections of temporally-invariant data for which there is a simple textual interface. In contrast, in this research we attempt to focus on repositories of temporal data such as news archives. Such digital collections often allow issuing structured queries composed of search terms and a selected time period, which specifies the temporal constraints for returned data. Due to the temporal character of the data different kinds of longitudinal type knowledge can be captured.

G. Vossen, D.D.E. Long, and J.X. Yu (Eds.): WISE 2009, LNCS 5802, pp. 167–180, 2009.

In this paper, we propose a method for detecting coordinate terms to a user-issued query together with their context and for visualizing their changes over time [10]. Our approach is based on data queried from online collections of news articles. Coordinate terms are bound by the fact of having the same direct hypernym term. Hypernym relation of terms is the relation represented by the pattern "is a kind of". In other words, these are terms that are on the same (semantic) hierarchy level and that often indicate peer or even rival relationships between real-world objects represented by terms. For example, Toyota, Mitsubishi and Honda are coordinate terms representing similar automobile companies in Japan, or Norway, Sweden and Finland are coordinate terms indicating countries in Scandinavia. The application that we demonstrate detects coordinate terms from an online repository of news articles over arbitrary time periods and visualizes them in form of an interactive, dynamic network. We believe that the proposed method could be applied for educational purposes at schools or other teaching facilities. Through exploratory search students could serendipitously discover relations between requested objects over time. In addition, the detected knowledge could be used as an input for subsequent processes in more complex application frameworks; for example, in automatically building histories of real-world objects such as companies, persons or countries and their inter-relationships over time.

In summary, in this paper, we (a) propose a method of detecting and visualizing coordinate terms as well as their contexts over time, (b) demonstrate an interactive application for visualization and exploration of the retrieved data and (c) evaluate it on example queries and through user studies.

The remainder of this paper is as follows. In the next section, we discuss the background and the related research. Section 3 describes our method and application for detecting and visualizing coordinate terms over time. Section 4 provides the experimental evaluation. We conclude the paper in the last section.

2 Background and Related Research

2.1 Knowledge Search from Search Engines

According to an online survey that we conducted in Japan in February 2008 on the group of 1000 respondents, 50.2% of users perform some kind of knowledge search using conventional search engine interfaces. By this, we mean the activity that is different from a usual search for arbitrary Web documents. According to the results of the survey, at least once in a month, users try to a) learn the correct spelling of input phrases (22.4% respondents), b) search for exact information such as location names returned for query "the origin of tulips is" (26.7%), c) determine the meaning of acronyms (26.7%), d) search for related terms (21.4%) or e) investigate the popularity of real-world objects (23.3%). In general, 89.8% users perform an object-level search that is a search for the information about concrete instances of persons, institutions or objects using the interfaces of standard search engines. Regarding the temporal aspect of information on the Web, 86.2% of the respondents declared that they would like to know the age of information they encounter. In the view of these results we can assume that there is a need for applications that would facilitate knowledge extraction from search engine repositories. Also, users should benefit from applications that help to estimate the age and temporal characteristics of certain information.

2.2 Related Research

Several techniques have been proposed for the purpose of discovering relations between terms. For example, in the context of coordinate term detection problem, Hearst [8] proposed discovering hypernyms through pattern matching in news articles, while Shinzato and Torisawa [12] acquired coordinate terms from HTML documents. In the latter work, coordinate terms were the ones appearing in the same levels of DOM structure such as itemized lists.

However, currently, it is difficult to freely scan the whole content of large data repositories such as collections of news articles, digitalized books or Web search engine indices. This is usually due to their huge size, proprietary character or access restrictions. Consequently, effective ways for mining data collections through their search interfaces have been proposed [2,6,10]. Bollegala et al. [2] measured inter-term similarity by analyzing Web search results. Cilibrasi and Vitányi [6] proposed Google Normalized Distance based on WebCount values in order to use it for such tasks as hierarchical clustering, classification or language translation. None of these works, however, focused on mining data collections of temporal character through their search interfaces.

Several proposals have been also made for analyzing and mining temporal document collections [1,9,13]. Topic Detection and Tracking (TDT) [1] initiative is probably best-known effort of detecting, classifying, and tracking events and story topics in historical news corpora. In addition, several methods for temporal weighting of terms have been proposed [9]. Nevertheless, the above approaches worked by scanning whole collections and required unrestricted access to every document, while our method relies on accessing the collections only through their search interfaces.

In a preliminary report [10] we have outlined a basic part of the coordinate term extraction method and shown initial system snapshots. In this paper, we a) enhance the term extraction and agglomeration method by adding additional linguistic patterns and by applying smoothing of extracted results, b) extend the context detection method and c) perform the experimental evaluation of the coordinate term extraction accuracy as well as conduct user studies on the overall system.

Our approach to visualizing the evolution of coordinate terms is related to topics of visualizing network evolution. Displaying evolution of network structure over time is quite a challenging task involving such issues as efficient indication of changes or preservation of their context [4]. There are basically two ways for visualizing graph changes over time. The first one relies on displaying sequence of graph snapshots chronologically in separate planes [3,7], while the second one uses various animation effects [4,5]. We use the latter visualization technique providing users with several additional interaction capabilities and attempting at decreasing users' cognitive overload when observing the evolution of term relationships.

3 Detecting and Visualizing Coordinate Terms over Time

In this section, we present our method of mining online news article collections for detection and visualization of changes in coordinate terms over time. The proposed application retrieves data from Google News Archive[1] search interface. Google News

[1] http://news.google.com/archivesearch

Archive is a service for online searching in archived collections of newspapers over the last 200 years. The service allows entering temporally structured query composed of keywords and a desired time period. It ranks results taking into account the full text of each article, its publication venue, timestamp and other factors.

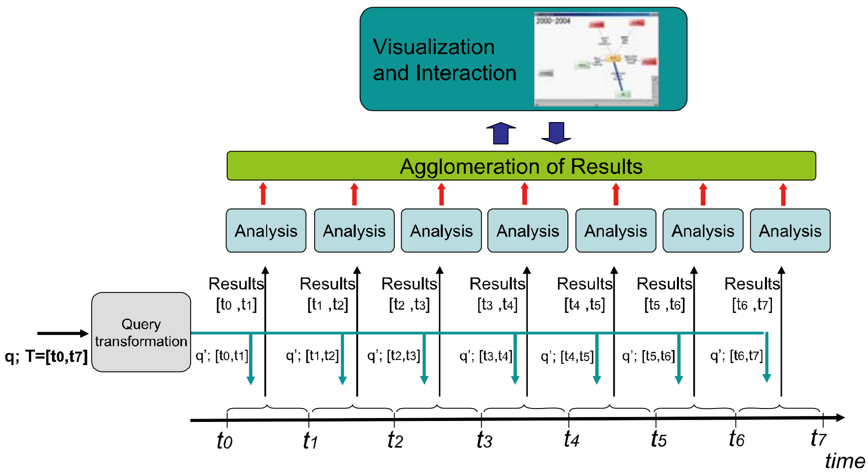


Fig. 1. Outline of the proposed method

The overview of our method is shown in Figure 1. First, a user enters query *q* with a certain selected time period *T*. The query is then transformed and sent to Google News Archive search as a series of queries for mutually exclusive and consecutive unit time segments within *T*. The time granularity can be also decided by the user (by default it is equal to T/10), although, the user should be aware of the fact that increasing the number of unit time periods raises the time cost. The system then collects the top results returned over each unit segment and detects coordinate terms using certain lexico-synactical patterns. The detected terms are then agglomerated in order to provide an interactive presentation. In addition, context of each relation between the terms is retrieved. Coordinate terms are shown in the form of a network, in which nodes represent terms and the edges represent the strength of coordinate relationships between the terms together with their contextual descriptions. User can drag the slide bar in order to see the changes in the graph structure for different time units.

3.1 Detection of Coordinate Terms

3.1.1 Coordinate Term Detection in Unit Time Segment

We describe here the method for detecting coordinate terms within a unit time period. When a query is issued it is converted to specially crafted "pattern queries" (denoted as *q'* in Figure 1), whose objective is to estimate the support of coordinate relationships between the query term and detected candidate terms. There are two pattern queries for each unit time period: one containing expression "*q* or" and containing expression "or *q*". The returned 100 snippets for each of these pattern queries are then analyzed for the occurrence of both syntactical patterns. Terms that frequently appear before or after the pattern queries (i.e., "*q* or *x*", "*x* or *q*", where *x* is a discovered candidate term) are deemed to be coordinate terms to the query term. We use such bi-direct patterns in our method for the purpose of determining the necessary cut points for terms [11]. Otherwise the system could not detect the lengths of noun phrases that are composed of more than one term (e.g., Hong Kong). Thus only the terms that appear at least once in each of the pattern are taken into account. In addition, extracted terms are considered to be coordinate terms to the query term if the geometric average of their both pattern frequencies within returned snippets is higher than some predefined threshold[2]. The system also rejects common terms via a stop word list.

Additionally, an option is also provided that extends the above query patterns with a contextual term. For example, one could search for coordinate terms to query "apple" in the context of computer companies rather than fruits by appending a contextual term "computer".

Lastly, pattern queries "*q* vs.", "vs. *q*", "*q* and", "and *q*" are also used instead of the one with conjunctive "or" for returning more pronounced rival-type relationships.

3.1.2 Aggregation of Coordinate Terms over Time

After coordinate terms have been found for each individual time segment the system aggregates them over time. This allows for determining the relative strengths of coordinate relationships within the whole time period by comparing their pattern frequencies. Weighted smoothing is done in order to decrease the volatility of term relationships. This is because sometimes a given relationship occurs during a long time frame, yet it ceases to be detected in a short time period (e.g., one or two time units) that is s sub-period of that time frame. This may happen due to the lack of newspaper articles mentioning one of the terms for certain time units. To alleviate this effect the system uses window of a predefined length (three time units by default) with adjustable weights equal to 0.2, 0.6 and 0.2[3]. Thus, with this weight setting 60% of the smoothed value of a given relation in a certain time unit comes from the actual support for this relation in that time unit, while the remaining 40% comes from the supports for the relation in the both adjacent time units.

3.2 Detecting the Context of Coordinate Relationships

As mentioned above the system also detects and visualizes the context of each coordinate relationship. This is because real world objects can be rivals (or peers) within their different contexts. For example, countries like India and China can be listed as Asian countries, the most populous countries or emerging economic powers. We determine the context of coordinate relationship by using the set of modified versions of *TF*IDF* weighting scheme (Equations 1-4) that take into account the distribution of terms in a single time window or in the whole time period of analysis.

---

[2] By default the threshold is equal to 1, which means that coordinate terms should appear at least once in each of the both patterns ($\sqrt{1}*\sqrt{1}=1$).

[3] Both the weights and the window size can be adjusted by the system operator, for example, to further decrease the volatility.

The simplest way is to consider only the amount of returned snippets that contain the coordinate terms inside a unit time period (Equation 1).

$$S(a, p_b; t_i) = SF(a, p_b; t_i) \tag{1}$$

Here $S(a, p_b; t_i)$ is weight of a context term a of a relationship p_b, which bounds the query and coordinate term b, in a time period t_i. $SF(a, p_b; t_i)$ is the number of snippets that support p_b (contain lexical patterns binding the query and term b) and also contain the term a within time period t_i.

Next we can also consider the total number of snippets that contain the term a within t_i (Equation 2). If this number is high then the weight of a context term a will be low since the term appears in many snippets and thus may not be representative for the particular relationship p_b.

$$S(a, p_b; t_i) = SF(a, p_b; t_i) * \log_2\left(\frac{M(t_i)}{DF(a; t_i)} + 1\right) \tag{2}$$

$M(t_i)$ denotes here the count of snippets in t_i and $DF(a; t_i)$ is the number of snippets containing a in t_i. Another way is to take into account all the snippets returned over the whole time frame.

$$S(a, p_b; t_i) = SF(a, p_b; t_i) * \log_2\left(\frac{M(T)}{DF(a, p_b, T)} + 1\right) \tag{3}$$

Here, $M(T)$ denotes the count of snippets within the whole time period T and $DF(a, p_b, T)$ is the number of snippets supporting p_b that also contain term a within T.
Lastly, the combined approach is used (Equation 4).

$$S(a, p_b; t_i) = SF(a, p_b; t_i) * \sqrt{\log_2\left(\frac{M(t_i)}{DF(a; t_i)} + 1\right) * \log_2\left(\frac{M(T)}{DF(a, p_b, T)} + 1\right)} \tag{4}$$

The three terms that have the highest values of weights according to one of Equations 1- 4 are then chosen as the context of the coordinate relation between the query and term b within t_i. We apply Equation 4 as a default method.

3.3 Visualization

Visualization of the results is done through an interactive, animated graph (Figures 2-3), in which nodes represent terms, and edges represent coordinate relationships with the thickness of an edge indicating the strength of the relationship. Each edge is also annotated with the top three contextual words. First, a user enters a query term and then by dragging the horizontal slider she or he can sequentially observe coordinate terms over different time periods. The length of a unit time segment is chosen by users. At any time point, the user can click on a selected node in the graph. This sends another query that contains the term in the clicked node. Consequently, the coordinate terms to both the query and the term in the clicked node will be shown in the expanded graph (see right-hand side example in Figure 2).

We have used a spring-type graph in which nodes' positions are determined on the basis of attraction/repulsion forces acting between the nodes. The former occurs between any two nodes bound by a coordinate relationship, while the latter is applied to the nodes that do not have such a relationship. The equilibrium position of all nodes in the graph is determined by the situation in which all acting forces are balanced. Users can freely interact with the network by changing positions of any nodes.

From a temporal viewpoint, noticing and understanding changes in the graph structure is rather difficult as it poses high cognitive overload for users. In order to facilitate the change understanding we decided to make several enhancements (see Figure 2 for example). First, the node of a query term is coloured in orange and placed in the central position. Second, the nodes whose underlying terms cease to have coordinate relationship with a query term in the next consecutive time unit are shown in gray colour. In contrast, the newly added nodes are coloured in red. In addition, due to the spring type of the graph, the new nodes move towards each other until they are stopped in their equilibrium positions and bound by edges. On the other hand, the nodes whose terms cease to have coordinate relationships in the next time unit are released from their edges and are pushed away from each other until they move outside the range of any acting forces. This colouring and animation schemes help users with noticing changes in co-ordinate relationships between the consecutive time units.

3.3.1 Summary Viewing Mode

As the above working mode shows only micro-scale changes in terms' relationships, a macro-scale overview should be also useful for users. Naturally, users could simply decrease the time granularity when interacting with a graph; for example, by changing the duration of time units from 1 to 4 years. However, a summarized overview of coordinate terms in a single snapshot would make it easier for users to obtain a rough overview of graph's evolution (Figure 3). In this kind of visualization, the width of edges indicates the overall relationship strength measured as the average pattern frequency over all the unit time segments. Each edge is annotated with its main relationship context, which is expressed as the top 3 contextual terms that occur most frequently over the whole time period for that relationship. Lastly, we also indicate the age of relationships by varying the thickness of node frames. The thick frame indicates terms that have been coordinate for a relatively long time, while the thin one denotes the recent coordinate terms.

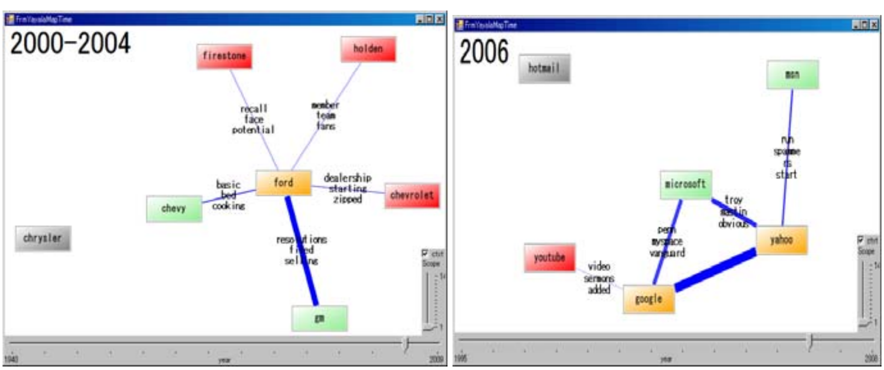


Fig. 2. Coordinate terms to the query "ford" and "google/yahoo"

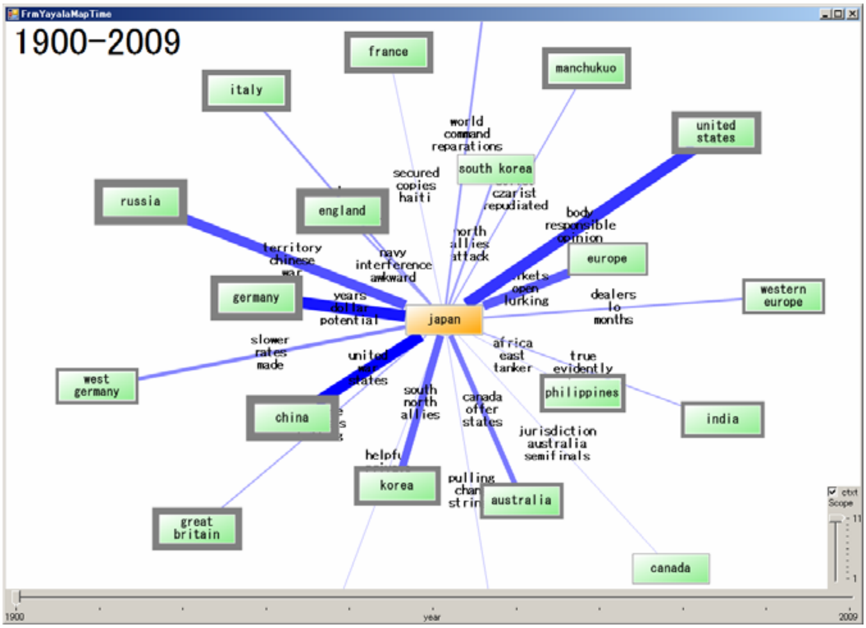


Fig. 3. Summary view of coordinate terms to the query "japan"

4 Experimental Evaluation

In this section we report on the results of the evaluation of our method and the implemented system which consists of measuring the precision of coordinate term extraction and performing user study.

4.1 Evaluation of Co-ordinate Term Detection

We have chosen a set of 20 queries for evaluation of the precision of coordinate term extraction. The queries were grouped into four categories: persons, companies and institutions, places and others (ex. physical objects or concepts). Table 1 shows the queries chosen for the evaluation. For the coordinate term extraction we have used three lexical patterns with conjunctives "or", "vs." and "and". We have applied a threshold equal to 1 which means that a given term was deemed to be a coordinate one in a certain time unit if among the results obtained over this time unit there were at least one pattern "*q* or *x*" and one pattern "*x* or *q*" for the case of the conjunctive "or".

For each query we have checked the correctness of their coordinate terms by consulting Wikipedia[4] and the Web. For example, for the queries "Ford" and "Jack Nicholson" we assumed coordinate terms to be correct if they represent other companies that operate in the same market and other movie actors or directors, respectively.

[4] Wikipedia: http://www.wikipedia.org

Table 1. Queries used for the evaluation with their corresponding time periods and granularities shown in parentheses

<table>
<tr><th>Groups</th><th colspan="4">Queries</th></tr>
<tr><td rowspan="2">Persons</td><td>Barry Bonds
(1986-2009;
2 years)</td><td>Hillary Clinton
(2000-2009;
1 year)</td><td>Barack Obama
(2000-2009;
1 year)</td><td>Michael Jordan
(1986-2009;
2 years)</td></tr>
<tr><td>Ronaldinho
(2000-2009;
1 year)</td><td>Ayrton Senna
(1986-2009;
2 years)</td><td>Nick Faldo
(1986-2009;
2 years)</td><td>Jack Nicholson
(1986-2009;
2 years)</td></tr>
<tr><td>Places</td><td colspan="2">Japan
(1900-2009; 10 years)</td><td colspan="2">Poland
(1900-2009; 10 years)</td></tr>
<tr><td rowspan="2">Companies, institutions</td><td>Google
(2000-2009;
1 year)</td><td>IBM
(1940-2009;
5 years)</td><td>Ford
(1940-2009;
5 years)</td><td>NASA
(1940-2009;
5 years)</td></tr>
<tr><td colspan="2">Harvard
(1900-2009; 10 years)</td><td colspan="2">Toyota
(1940-2009; 5 years)</td></tr>
<tr><td>Others</td><td>baseball
(1900-2009;
10 years)</td><td>dollar
(1900-2009;
10 years)</td><td>TV
(1940-2009;
5 years)</td><td>Internet Explorer
(2000-2009;
1 year)</td></tr>
</table>

Table 2 shows the average numbers of detected coordinate terms in unit time segments and precision values for different lexical patterns. We found out that the lexical pattern containing the conjunction “or” results in the highest precision (0.890). The lexical pattern containing the conjunction “vs.” achieves also reasonably high precision (0.875). However, in this case the average number of retrieved coordinate terms within single time segments is very low (0.06) as this pattern is rarely used (ex. mostly in sports news). Both conjunctions “or” and “and” are on average more useful as producing higher number of results. Although, the pattern with the conjunctive “or” achieves higher precision (0.890) than the one with the conjunctive “and” (0.825), the latter produces on average about 35% more correct results.

Next, we investigated how the precision changes for different query groups. Table 3 lists the results of each query group for the lexical pattern using conjunctive “or”. We have obtained the average precision on about 0.890 for all the query groups. The system achieved the highest precision for the queries denoting person and place names (0.969 and 0.942, respectively), while the lowest precision was obtained for “others” category (0.808). The latter was probably due to many different meanings and functions of the objects indicated by these queries. We also noticed that the queries representing place names resulted in the highest average number of correct coordinate terms within a unit time segment (5.18) and the queries in the “others” group has the lowest number (2.18).

Note that we have evaluated the correctness of coordinate terms extraction irrespectively of time periods of their occurrence. The evaluation of time-based precision (i.e. correctness of coordinate term extraction in particular time segments) is however quite difficult as finding past coordinate terms is non-trivial. Instead, we have

Table 2. Results for different conjunctives

	Conjunctive "or"	Conjunctive "vs."	Conjunctive "and"
Average number of correct results for single time unit	2.91	0.06	4.45
Precision (correct/returned results)	0.890 (683/767)	0.875 (14/16)	0.825 (1045/1266)

Table 3. Results for different query groups

	Per-sons	Places	Companies, institutions	Others	Average
Average number of correct results for single time unit	3.12	5.18	2.51	2.18	**2.91**
Precision	0.969	0.942	0.810	0.808	**0.890**

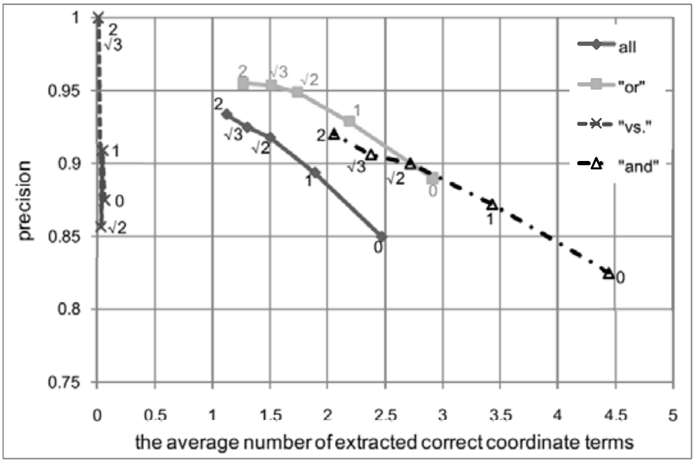
superficially checked whether there are any coordinate terms that should not be reported in particular time periods. On average, most of the coordinate terms appeared to occur within their correct periods. If there were any mistakes they usually resulted from too coarse granularity applied. For example, when using 10 years as the length of a unit time period our system reported "Soviet Union" and "Czechoslovakia" to be co-ordinate terms of Poland from 1900 to 2000. This is only partially correct as actually, the Soviet Union has been disestablished in December 26, 1991[5] and Czechoslovakia dissolute in December 31, 1992[6].

In the above evaluation, we have not calculated recall as it is rather difficult considering the large number of potential coordinate terms for the selected query words. Instead, to gauge the overall performance, we have plotted the precision rates against the average number of correct coordinate terms detected in unit time segments with different threshold values (see Figure 4). The different threshold levels are indicated on the graph by numbers. This allowed us to see how many correct coordinate terms have been on average dropped when the threshold is increased and how the precision changes in such cases. We can see that using the lexical pattern with the conjunctive "or" results in the best performance closely followed by the one with the conjunctive "and". The line denoted by "all" indicates the case when the results of all the lexical patterns are considered together.

4.2 User study

We now show the results of an experimental evaluation of our system on a group of users. We have asked 14 subjects who are computer science students to interact with our system. First, we have briefly explained the way to operate the system. Next, the subjects had to complete 3 tasks each lasting 5 minutes. Prior to conducting each task

[5] http://en.wikipedia.org/wiki/Soviet_Union
[6] http://en.wikipedia.org/wiki/Czechoslovakia



Figu. 4. Precision against the average number of extracted correct coordinate terms in unit time segments over different threshold levels (the values of the threshold levels are indicated on the graph)

with our system they were also asked to attempt to do the tasks by using conventional Web search engine within 5 minutes. Below we briefly describe the tasks:

1. For “Ford” query find 2 past (but not present) and 2 present rivals/peers as well as 2 entities that were rivals in the past and continue to be rivals now. Set the time frame ranging from 1940 until the present with a unit time period of 5 years when using the proposed system.
2. Do the same for query “Japan” with the time frame spanning from 1900 until the present with a unit time period of 10 years.
3. Do the same for “Google” and “Yahoo” terms (time frame: 1995 until the present, 2 years unit time period) and find also their common, present rivals.

As expected, the users could not find the rivals or peers of the query terms within only 5 minutes using any conventional Web search engines or even the Google News Archive Search interface itself. On the other hand, they could successfully complete all the tasks within the required time using our proposed system. Note that the tasks were still easy in the sense that we did not ask subjects to find coordinate terms within specific, shorter time periods.

After completing the tasks, we asked questions shown below in order to check user’s impressions and collect comments for the further improvement of the system.

1. Is it easy to complete tasks using conventional Web search engines? (“very”, “so so”, “I do not know”, “not so”, “not at all”)
2. Is it easy to complete tasks using the proposed system? (“very”, “so so”, “I do not know”, “not so”, “not at all”)
3. Please comment on good and bad aspects of the normal viewing mode and the summary viewing mode.
4. What aspects of the system are good and what bad?

5. What would you like to change or add in the system?
6. For what kind of objects do you want to find coordinate terms using the system (e.g., countries, companies, persons)?
7. What kind of other historical knowledge would you like to obtain (e.g., other relations between objects)?

The results for the first and second questions are shown in Figures 5 and 6. They indicate the general usefulness of our system and its advantage for past coordinate term detection over conventional Web search engines.

For the 3rd question, the subjects generally appreciated the availability of a macro and micro-scale viewing modes. The good points in the summary viewing mode were providing an overall impression and showing entities that are in general rivals/peers. In addition, there was no need for any manual interaction as the results were presented directly in a single frame. However, the respondents reported some problems with understanding the information that is conveyed through the summary viewing mode. Normal viewing mode was considered as useful for seeing changes over time and interacting with the system.

For the 4th question the users generally agreed that it is interesting to visualize the changes of relationships over time and that the system can show rivals to an entity represented by a given query. Three subjects wrote that it is thus possible to better understand the history of certain objects. However, the same number of subjects agreed that the way to operate the system is somehow complicated mostly due to the confusion with the meanings of colors and width of edges in the graph. This was especially difficult in the summary viewing mode. Thus there is still room for improvement at the interface level. Also, one user complained about the occurrence of synonyms presented as different nodes in the graph (e.g., “alta vista” and “altavista”, “soviet” and “russia”) and one about coordinate terms appearing and disappearing unexpectedly. The former issue could be improved by employing more complex NLP techniques, while the latter could be alleviated by the suitable choice of weight settings in the smoothing. In addition, users told us that the system sometimes showed friendship relations rather than rival ones making it hard to distinguish between the both, even when their context was provided. This was especially true for the entities that the users did not know much about.

For the next question the users proposed improving contextual description of relations, for example, by providing larger textual summaries, and showing changes more clearly and directly. Some other interesting comments were: to show precisely when relationships start and end, to make automatic slideshow mode, and to implement the system in multi-touch panels. In addition, one subject wanted to use the system on the Web as an online service.

For the 6th question, persons and products were the most often selected answers (5 users) followed by companies (4 users). Some of users also expressed wishes to see the history of the rival relationships of academic societies, musicians, countries, diseases, songs, foreign words and sport teams. However two subjects stated that there are not many objects for which we could get interesting results and, they may actually, rarely want to know any rival relationships.

In the last question, the users expressed wishes for knowing various different types of historical knowledge such as the historical reputation of objects, historical values

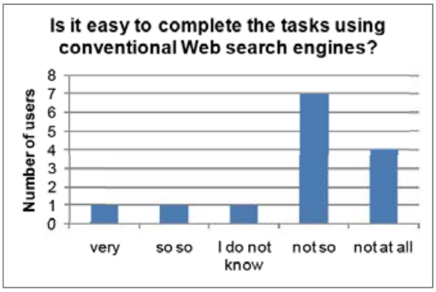


Fig. 5. Answers to Question 1

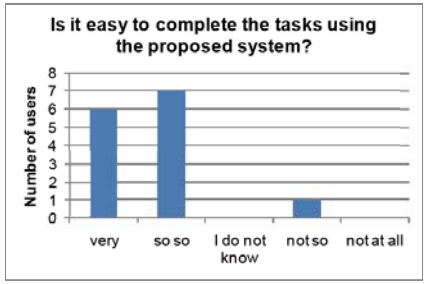


Fig. 6. Answers to Question 2

of stock prices, different types of past relationships among persons and companies, personal histories with their related information, historical wars, enemies and old buzz terms. An interesting comment was to display the information on the past common sense which cannot be currently captured using conventional techniques.

5 Conclusions

In this paper, we have proposed a method for the detection and visualization of changes in coordinate terms over time from online news article collections. This is an example of mining search engine interfaces that allow for temporally structured queries in order to extract knowledge of temporal character. Using the proposed system users can discover terms that 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. The experimental results have indicated high effectiveness of the proposed approach of extracting coordinate terms over time and confirmed the usefulness of our visualization approach.

Acknowledgments

This research was supported in part by the National Institute of Information and Communications Technology, Japan, by Grants-in-Aid for Scientific Research (Nos. 18049041, 21700105, and 21700106) from MEXT of Japan, by the Kyoto University Global COE Program: Informatics Education and Research Center for Knowledge-Circulating Society, and by a Microsoft IJARC CORE4 project: Toward Spatio-Temporal Object Search from the Web.

References

[1] Allan, J. (ed.): Topic detection and tracking: event-based information organization. Kluwer Academic Publishers, Norwell (2002)
[2] Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring semantic similarity between words using web search engines. In: WWW 2007, pp. 757–766. ACM Press, New York (2007)

[3] Brandes, U., Corman, S.R.: Visual unrolling of network evolution and the analysis of dynamic discourse? Information Visualization 2(1), 40–50 (2003)
[4] Chen, C.: Information Visualization. Springer, Heidelberg (2006)
[5] Chen, C., Morris, S.: Visualizing Evolving Networks: Minimum Spanning Trees versus Pathfinder Networks. In: INFOVIS 2003 (2003)
[6] Cilibrasi, R., Vitányi, P.M.B.: The Google Similarity Distance. IEEE Trans. Knowl. Data Eng. 19(3), 370–383 (2007)
[7] Erten, C., Harding, P.J., Kobourov, S.G., Wampler, K., Yee, G.: Exploring the computing literature using temporal graph visualization. In: Proceedings of SPIE. Visualization and Data Analysis, vol. 5295 (2004)
[8] Hearst, M.A.: Automatic Acquisition of Hyponyms from Large Text Corpora. In: COLING 1992, pp. 539–545 (1992)
[9] Kleinberg, J.: Temporal dynamics of on-line information streams. In: Garofalakis, M., Gehrke, J., Rastogi, R. (eds.) Data Stream Management: Processing High-Speed Data Streams. Springer, Heidelberg (2005)
[10] Ohshima, H., Jatowt, A., Oyama, S., Tanaka, K.: Visualizing Changes in Coordinate Terms over Time: An Example of Mining Repositories of Temporal Data through their Search Interfaces. In: Proceedings for the International Workshop on Information-explosion and Next Generation Search (INGS 2008), IEEE CS Digital Library, pp. 61–68 (2008)
[11] Ohshima, H., Oyama, S., Tanaka, K.: Searching Coordinate Terms with Their Context from the Web. In: Aberer, K., Peng, Z., Rundensteiner, E.A., Zhang, Y., Li, X. (eds.) WISE 2006. LNCS, vol. 4255, pp. 40–47. Springer, Heidelberg (2006)
[12] Shinzato, K., Torisawa, K.: A Simple WWW-based Method for Semantic Word Class Acquisition. In: RANLP 2005, pp. 493–500. Springer, Heidelberg (2005)
[13] Swan, R., Allan, J.: Automatic generation of overview timelines. In: Proceedings of the 23rd Conference on Research and Development in Information Retrieval, Athens, Greece, pp. 49–56. ACM Press, New York (2000)