Clustering of Search Results using Temporal Attributes

Omar Alonso and Michael Gertz
Department of Computer Science
University of California at Davis
{oralonso,gertz}@ucdavis.edu

ABSTRACT
Clustering of search results is an important feature in many of today's information retrieval applications. The notion of hit list clustering appears in Web search engines and enterprise search engines as a mechanism that allows users to further explore the coverage of a query. However, there has been little work on exposing temporal attributes for constructing and presentation of clusters. These attributes appear in documents as part of the textual content, e.g., as a date and time token or as a temporal reference in a sentence. In this paper, we outline a model and describe a prototype that shows the main ideas.

Categories and Subject Descriptors
H.3.3 [Information Search and Retrieval]: Clustering

General Terms
Algorithms, Experimentation, Human Factors.

Keywords
Clustering, information extraction, temporal retrieval, user interfaces, experimentation.

1. INTRODUCTION
Hit list clustering has emerged as an alternative mechanism to display similar documents in one page without forcing the user to go through hundreds of items. Clustering of the result set can lead to better user interfaces and therefore can lead to better user experience. A recent study on user search experience has found that users do prefer to use clustering when they are trying to get an overview of a topic [3].

Current hit list clustering engines like Vivísimo rely on a search engine that provides some information like page title, URL, and snippet for the construction of the clusters. Rarely a time attribute appears as part of the labels. A quick look at any of the current search engines shows that the temporal aspect is restricted to sorting the hit list by date only. The date attribute is mainly the creation or last modified date of a Web page. In some cases, it can be misleading, because the time stamp is provided by the web server and may not be accurate.

A query for "football world cup" now returns information about the forthcoming event in Germany. But every football fan knows that the event happens every four years. Another example is "Iraq war" where the results are based on the latest events with little from the 90s war. Wouldn't it be helpful if the system is more aware of the temporal attributes inside the documents and present results grouped by them?

As the amount of Web content coverage increases (including Web archives), the notion of restricting search by time becomes more important. Time and time measurements can help in recreating a particular historical period or set the context of Web documents.

Our approach focuses on adding time attributes to hit list clustering based on a combination of metadata and information extraction. Documents contain creation date (metadata) and time information as part of the content (tokens that reference time). We provide a user interface that combines topics and time clusters.

Interestingly, there is only little work on exploiting temporal information for clustering search results. The time frames project is an approach to augment news articles by extracting time information [5]. A popular hit list clustering construction is suffix trees [2]. A system for extracting temporal attributes of documents using information extraction to obtain temporal references besides a data format is presented in [6]. Recently, new research has emerged for future retrieval [1] where temporal information can be used for exploring the future.

2. MODEL
A document typically has temporal metadata, such as the creation date or modification date, if a version control system is in place. We can also observe that the content of a document can have temporal references to the past and/or future. These temporal entities are either (1) explicitly represented, such as date and time ("March 12, 2004"), (2) denoted as events that have an associated time value like a holiday ("Christmas" or "Thanksgivings"), or (3) represented as a vague reference ("by Friday"). The 1999 NE recognition task definition contains a precise list of temporal expressions as part the TIMEX tag element [8].

We argue that there is a lot of temporal information in any corpus of documents and not just Web content. Financial news tend to be rich in describing near future events. Resume documents contain several references to the past in a very precise way. Project documentation involves phase milestones that are captured in time. How can one take advantage of time attributes for retrieval purposes that go beyond sorting a hit list by date?

We provide the user with an alternative presentation of a hit list clustered by temporal attributes. By temporal attributes we mean a combination of metadata and named-entity extraction results. The idea is to present the most important temporal attributes as clusters for hit list presentation. In essence, the hit list clustering has two views: *topics* and *time*. The topic view is based on traditional approaches (clustering on title, snippet etc.). The time view is based on temporal attributes.

SIGIR'06, August 6-11, 2006, Seattle, Washington, USA.
ACM 1-59593-369-7/06/0008.

There are two aspects for extracting time information from a document. The first one is metadata and the second one is time extraction. The metadata part is just a lookup for the creation date of the document and can be done at crawling or indexing time. Time extraction involves using named-entity extraction approaches to obtain time information from each document. The output of named-entity extraction is a document that contains the desired entities in a specific format, for example, XML.

We assume that for a document D, its time feature vector consists of $\{a_1,...,a_n\}$ attributes. At least one attribute is the creation date and the rest is the set of all named-entities that represent time. The named-entity extraction step performs the feature selection and extraction. All time attributes are normalized as part of the data cleaning. This step has some problems of its own like normalizing time zones, which we are not addressing in this paper. With the document now annotated with time, we can use the information to cluster by time in combination with metadata. We assume that the quality of the entities is good so we pick them as cluster labels. We use a hierarchical clustering algorithm based on complete-link since they tend to provide more compact clusters [7].

The advantage of our approach is that, in absence of explicit time information in document metadata, it generates time annotations based on document analysis. This important step provides the necessary information to detect similarities and finally compute the clusters.

3. PROTOTYPE

In our prototype, we have used a combination of technologies to demonstrate the feasibility of such a method and system. For indexing and storage, we use Oracle Text because of the combination of text and XML searching. Metadata information (creation/modification date) is captured in the internal schema of the prototype. We use the ANNIE information extraction component that is part of GATE [4] for performing the time annotation of documents. ANNIE relies on finite state algorithms for performing extractions. Some predefined entities are available through gazetteers and, if needed, the built-in rule language can be used for defining new entities.

For each document in the corpus, a Java program extracts time information and produces an XML document that is annotated accordingly. The output is simple but extensible enough so it can be replaced with emerging standards like TimeML. The output of a document looks like the following text fragment:

```
The World Cups <Date>between 1930 and 1998</Date>
were all held in ...The <Date>2002</Date> World
Cup was the first World Cup held outside
```

For each document, we then load the XML annotated version in Oracle and create a separate index. The internal structure now has the original version of the document and its time annotated view.

When a user enters a query, the engine retrieves the document ids that satisfy the query. The hit list cluster construction has two phases. The first one is the construction of the topics based on main attributes such as title, summary or snippet. The second phase involves retrieving the time annotation fragments associated with the document ids. Using XPath to extract and group the time elements, we then form a sub tree that contains time information. An approach based on tree merging is used to compute the time clusters. Each output is presented as a view in the interface (Figure 1). In our preliminary findings, cluster labels appear not to be a problem, because the number of time elements is small and also precise.

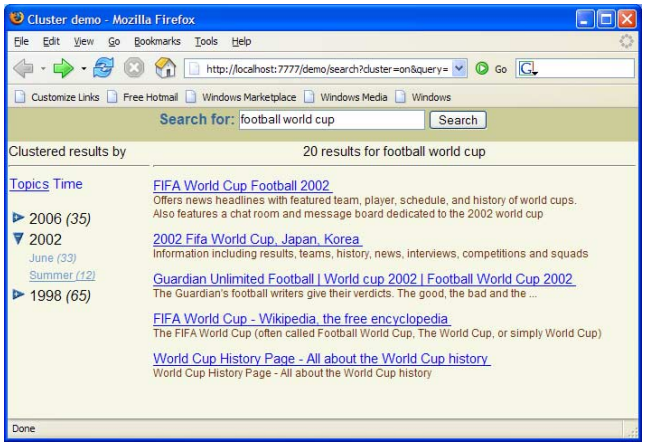

Figure 1. Clustering by time and topics

4. CONCLUSIONS

As information retrieval applications gather and archive massive data sets, presenting information that has temporal relevance becomes more important. We have provided an outline of a method and an interface that presents clustered search results by time and topic, providing the user with another way for document discovery and exploration. Future work includes adding other data sources beyond Web document and extending the temporal entity set for the extraction component.

5. REFERENCES

[1] R. Baeza-Yates "Searching the Future" *SIGIR Workshop MF/IR* (2005).

[2] O. Zamir and O. Etzioni "Web Document Clustering: a Feasibility Demonstration", *Proc. of 21st SIGIR* (1998), 46-54.

[3] A. Aula, N. Jhaveri, and M. Kaki. "Information Search Re-access Strategies of Experienced Web Users", *Proc. of 14th WWW* (2005), 583-592.

[4] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications", *ACL'02*. PA, July 2002.

[5] D. Koen and W. Bender. "Time Frames: Temporal Augmentation of the News". *IBM System Journal*, Vol. 39, No. 4 (2000), 597-616.

[6] F. Schlider and C. Habel. "From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages", *ACL'01 Workshop on temporal and spatial information processing* (2001), 1-8.

[7] A. Jain, M. Murthy, and P. Flynn. "Data Clustering: A Survey". *ACM Computing Surveys*, 31(3):264-323, (1999)

[8] http://www.itl.nist.gov/iad/894.01/tests/ie-er/er_99/er_99.htm