Tapping Huge Temporally Indexed Textual Resources with WCTAnalyze

Sebastian Gottwald[†], Matthias Richter[†], Gerhard Heyer[†], Gerik Scheuermann[‡]

University of Leipzig
[†]Natural Language Processing Group / [‡]Image Processing Group
Johannisgasse 26, DE-04103 Leipzig
{sgottwald,mrichter,heyer,scheuermann}@informatik.uni-leipzig.de

Abstract

WCTAnalyze is a tool for storing, accessing and visually analyzing huge collections of temporally indexed data. It is motivated by applications in media analysis, business intelligence etc. where higher level analysis is performed on top of linguistically and statistically processed unstructured textual data. WCTAnalyze combines fast access with economically storage behaviour and appropriates a lot of built in visualization options for result presentation in detail as well as in contrast. So it enables an efficient and effective way to explore chronological text patterns of word forms, their co-occurrence sets and co-occurrence set intersections. Digging deep into co-occurrences of the same semantic or syntactic describing wordforms, some entities can be recognized as to be temporal related, whereas other differ significantly. This behaviour motivates approaches in interactive discovering events based on co-occurrence subsets.

1. Introduction

Success and survival of economical and governmental institutions as well as individuals hinges upon their ability to locate, analyze, and use information efficiently, competently and adequately. In this paper we present WCTAnalyze, a novel tool that makes possible the management and interactive visual analysis of large temporally indexed collections of unstructured textual data. Besides static patterns, longitudinal and temporal patterns of huge changing text amounts can be extracted. WCTAnalyze aims at aiding an analyst interactively in the tedious process of finding, extracting, arranging and deducing knowledge represented in natural language by semi-automatic methods, rapid access and visualizations.

This paper is structured as follows: In Section 2. we introduce the original problems of mining large newswire collections and business intelligence which have lead to the development of WCTAnalyze. WCTAnalyze's design and implementation is described in Section 3.. Finally we conclude.

2. Collections of Temporally Indexed Textual Resources

Since in 1977 the General Inquirer was used to perform an almost online analysis of the Detroit News's data tapes (DeWeese III and McCombs, 1977), terabytes of news stories and other types of unstructured, periodical data have been published. There exists an ongoing tradition of dealing with them in computerized form. Summarization, clustering, filtering, tracking, extraction any many more tasks have been covered for example in TREC and TDT (Wayne, 2000) and real world systems such as Google News, the California Newsblaster (McKeown et al., 2002), NewsJunkie (Gabrilovich et al., 2004), NewsInEssence (Radev et al., 2005), the Words of the Day (Eiken et al., 2006), the European Media Monitor (Steinberger et al., 2005) and many more. Temporal Text-Mining on technical database texts, patent data etc. delivers crucial information everywhere from early stage R&D to management's decision making. Efficiency in storage and speed of access for an interactive analysis hasn't always been addressed much in these works. But with ever growing amounts of data the issues has exacerbated to a serious challenge.

For the application described here a couple of preprocessing steps are applied that result in a temporally indexed text database (or: *time slice database*) with frequency information about occurrences of all word types. Additionally neighbour and sentence co-occurrences are calculated based on the log-likelihood measure (Dunning, 1993) with a 1% significance threshold for the sentence based co-occurrences and a 5% threshold for the neighbours. The preprocessing and calculation is done with the tools described in (Biemann et al., 2004; Quasthoff et al., 2006). All data is available as time series data on occurrences and co-occurrences for all types in a configurable granularity (e.g. daily for news).

While for a single user developing prototypes with smaller amounts of data storage in a couple of highly optimized tables in a MySQL database on a standard PC has been fast enough, interactive performance isn't even near bearable when concurrent users access several year's data. A data example from the German daily news collection illustrates the performance problem: Covering a large portion of the five years from 2001–2005 results in approximately 1800 daily slices. The collection's size is approximately 25 million sentences / half a billion tokens. From the 4 million types contained, 62 million data records exist in the database. From the 63 million co-occurrences extracted, 184 million data records are to be stored. This performance problem had to be tackled with a software solution such as the one we present now in WCTAnalyze (Gottwald et al., 2007) since at some point performance gains by tuning database indexes weren't possible anymore.

3. Storing, Accessing and Visualizing Data with WCTAnalyze

We derived constraints for a data structure time slice corpora data from the example's figures. On the todo list of

the implementation have been an appropriate index structure, the support of data aggregation and multiple levels of granularity. Storage has to be efficient in disk space. Access has to be cheap in disk reads. The data is growing over time, therefore additions of data must be easy and must not force a time consuming reorganization. The implemented data structure consists of three closely coupled parts: a data container, an index for containers and an index for types. An overview of the architecture is depicted in Figure 1.

3.1. Storing and Accessing temporally indexed language data

The distribution of language data can be approximated by Zipf's Law (Zipf, 1935). The overwhelming majority of items are rare items, this still holds true in the case of time as an additional dimension as experiments (Richter et al., 2006) showed. In any slice only a small fraction of the complete items is observed (a maximum of 60.000 of 4 million words and 1.5 million of 63 million co-occurrences in the data example). Thus, the principle of storing data in backward-linked lists has got huge advantages in terms of I/O and disk space consumption. Using a set of 1800 time slice databases would force a prohibitively time-costly scan on any of them until a query were processed. Static data structures such as arrays allow for random access but waste too much space: 1.25 TB for co-occurrences and 55 GB for types would be needed. In contrast our data structure reduces this amount down to 1.25 GB for types and to 5.5/4.25 GB for co-occurrences (unoptimized/optimized).

Types and their co-occurrences are internally represented with word numbers. A bijective mapping from types and word numbers exists for presentation reasons and for interaction with the user. The time slice data in the data container, more precisely: the entry points of the master elements, are indexed such that an injective mapping exists from word ids respectively from co-occurrences to the entry point of the corresponding data set. The time complexities for calculating the entry point of an item from the container's index are O(1) for types and O(log n) for co-occurrences, n being the number of co-occurrences of a word form. This can be achived due to word numbers that represent index positions within an array of entry points. Figure 2 illustrates the architecture of these indexes.

In each data container time slice data is stored in structured binary representation. For each slice there is an entry for every occurring type and co-occurrence (in the following to be called item) containing the measured data and a position pointer. For every item there is also a master record that serves two purposes: the position of the record in the data container is a pointer to an item's data, the content of the record is the record's size. It is used for optimized read and growth operations which are described further down below. Records are implemented as backward-linked lists. The master record of an item points to position of the last inserted data of the item, this record points to the least recently inserted one and so on, with the first record containing a null pointer. Figure 3 illustrates this organization of an unoptimized data container.

The backward-linked list structure was chosen because it makes the addition of new records very easy and keeps

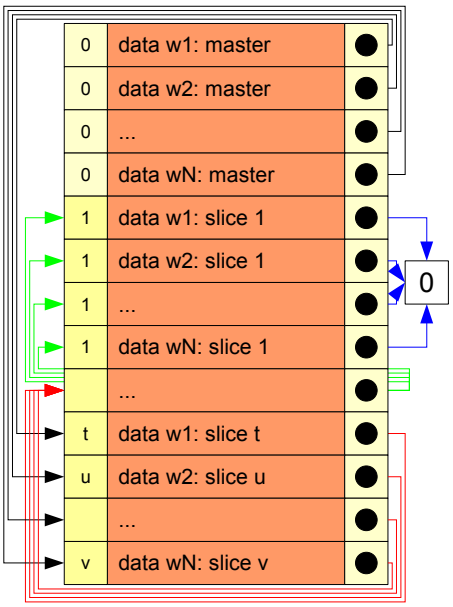

Figure 3: Architecture of an unoptimized data container.

costs for inserting newer data low. When adding data to a slice or adding a new slice the size of the structure and the value Z of the pointer only need to be read from the master record, data is simply written to the end of the data container and the pointer Z is adjusted to the new position.

3.2. Analysis and Visualization

WCTAnalyze currently implements three types of visualization: line plot, histogram and co-occurrence table. Combined with a lot of visualization options this allows to present results in detail and in contrast as well as the analysis of chronological text patterns of word forms or sets of word forms, in particular word forms that describe the same semantic or syntactic entity (here called *sense classes*), their co-occurrence sets and co-occurrence sets' intersections.

All curves can be smoothed by filters. Besides the display of raw frequencies, several data scaling and aggregation methods can be employed. Taking frequency ratios instead of frequencies reduces the danger of misinterpreting peaks in raw data as real peaks in measurement. For being able to compare numbers of very different sizes, which according to Zipf's law certainly will be needed, scaling of each signal to the fraction of its maximum or to a value between 0 and 1 mapped from the signal's minimum and maximum can also be used.

For co-occurrences in subsequent slices three cases can occur: it exists in either of the slices, it is only present in the former or it is only present in the latter. Cumulative sums on the relative frequencies of a sense class, respectively on the relative number of co-occurrences open another possibility of comparing developments directly. Besides this the k most frequent or high similarity ranked co-occurrences of a word form or sense class can be visualized in a line plot.

Using domain dependent sense classes as the data basis and using the proposed visualization options makes possible a

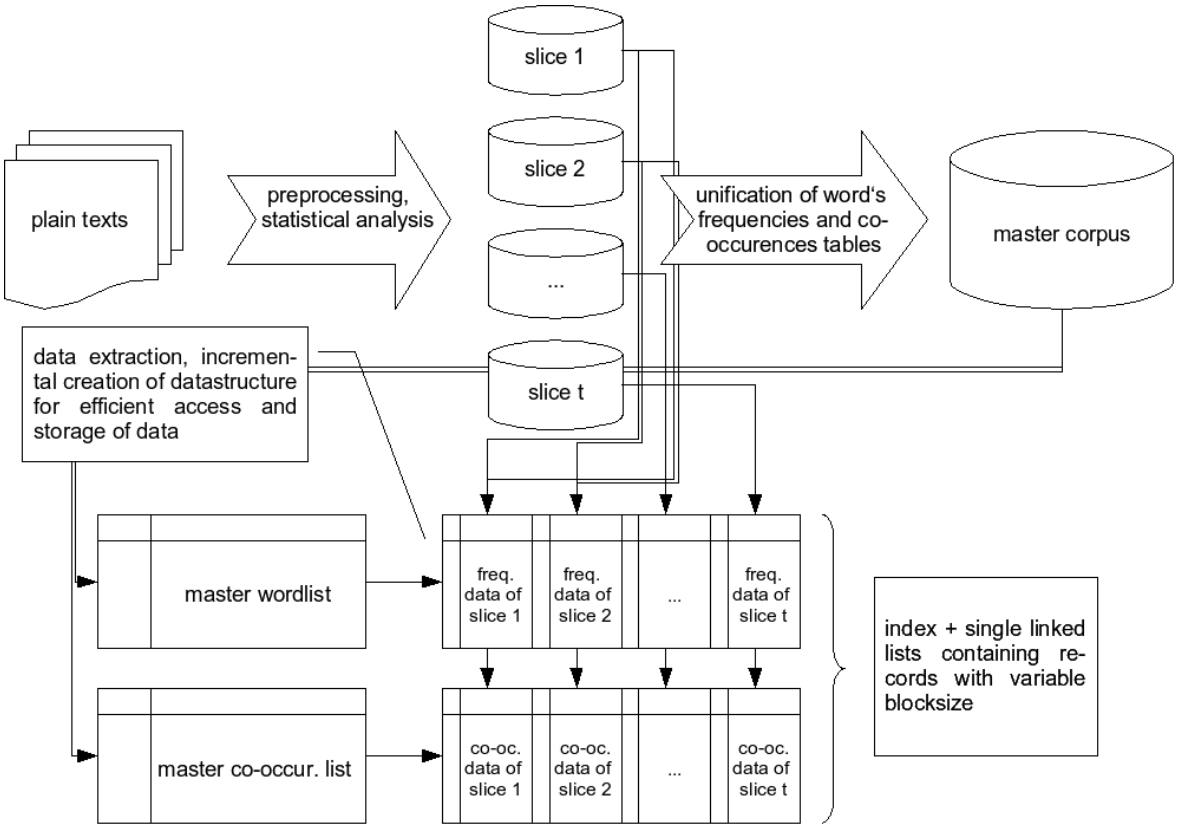

Figure 1: Architecture overview of WCTAnalyze.

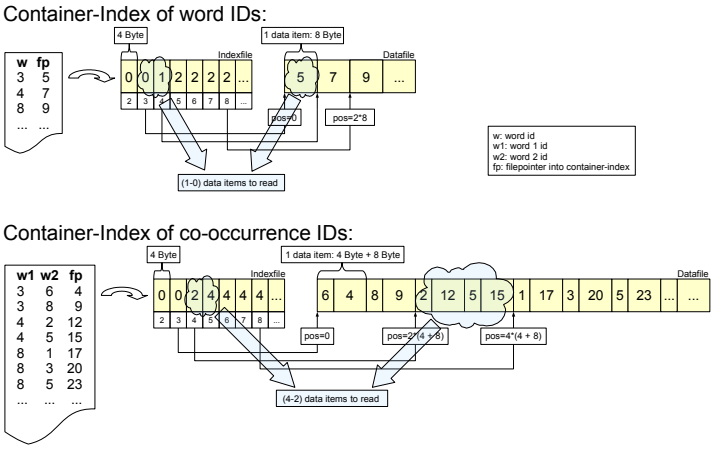

Figure 2: Architecture of the container's index.

quick overview of a domain in science, business and the like. In the line plots displays of the most significant co-occurrences and relative frequencies are the most efficient ones to compare and analyze sense classes and their co-occurrences. A human intuitively recognizes chronological patterns in the appropriately zoomed and focused selection and is able to spot events (such as "attack, September 11th") and their mode of occurrence.

An approach with a different visualization metaphora but with the same potential of gaining insight into the unknown for the analyst is the co-occurrence table combined with histograms generated from it. By looking at intersections of co-occurrence sets and these types' co-occurrences we can track and analyze the relations between topics of interest and their mutual influences. We can also identify the influences which gradually changed them in an iterative process. This visual analytics (Thomas and Cook, 2005) inspired approach is innovative in the area of chronological text patterns and remains the subject of current research: Digging deeper into the co-occurrences of a sense class revealed that they are more or less closely related as they have got frequency graphs that range from almost identical to completely different compared to the sense classes' ones. This motivated an approach to interactive event discovery

using Shared Nearest Neighbour Clustering (Ertöz et al., 2003) on co-occurrence subsets. A preliminary evaluation has supported this approach.

4. Conclusions and Future Work

We presented WCTAnalyze, a tool for storing, accessing, analyzing and visualizing large collections of temporally indexed textual data. Processing this data from raw material linguistically and statistically is only a first step. If an analyst wants to work with and conclude from this data on a higher level, rapid data access becomes crucial. WCTAnalyze not only solves this problem, but additionally gives the user visual aids to leverage her work. We believe that based on such tools human judgment can be assessed and improved by a verifiable and reliable data source in a lot of applications.

In the process of transcending from a mere static presentation of preselected results to a set of interactive analytics tools WCTAnalyze has been a decent milestone as far as fast data access is concerned. The planned visual and analytical enhancements will build on top of this. Most notably in this context are a topological view on document collections in an interactive view which is as intuitive as the ThemeRiver (Havre et al., 1999) but less restrictive and destructive when it comes to relations and number of topics covered. Also the idea of sense classes will be pursued further, so that their contents can be thought of as concepts, that can be associated with words, sentences, paragraphs and texts instead of just bags of words and who in turn can be related to more concepts, words, sentences, paragraphs and texts in their respective ways.

5. References

C. Biemann, S. Bordag, G. Heyer, U. Quasthoff, and C. Wolff. 2004. Language-independent Methods for Compiling Monolingual Lexical Data. In *Proceedings of the LREC-04*.

C. L. DeWeese III and M. E. McCombs. 1977. Computer Content Analysis of "Day-Old Newspapers": A Feasibility Study. *Public Opinion Quarterly*, (41):91–94.

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1).

U. C. Eiken, A. T. Liseth, H. F. Witschel, M. Richter, and C. Biemann. 2006. Ord i dag: Mining Norwegian Daily Newswire. In *Proceedings of the FinTAL*, number 4139 in LNAI.

L. Ertöz, M. Steinbach, and Kumar V. 2003. Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data. In *Proceedings of the SIAM International Conference on Data Mining 03*.

E. Gabrilovich, S. Dumais, and E. Horvitz. 2004. Newsjunkie: Providing personalized newsfeeds via analysis of information novelty. In *Proc. of www13*.

S. Gottwald, G. Heyer, M. Richter, and P. Walde. 2007. WCTAnalyze - Collecting, Indexing, Accessing and Visualizing Temporally Indexed Textual Resources. In *Proceedings of the IEEE TIME Symposium 2007, Alicante, Spain*.

S. Havre, B. Hetzler, and L. Nowell. 1999. ThemeRiver: In Search of Trends, Patterns, and Relationships. In *Proc. of IEEE Symposium on Information Visualization 99, San Francisco, CA, Oct 24 - Oct 29, 1999*.

K. R. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, J. L. Klavans, C. Sable, B. Schiffman, and S. Sigelman. 2002. Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster. In *Proceedings of the Human Language Technology Conference*.

U. Quasthoff, M. Richter, and C. Biemann. 2006. Corpus Portal for Search in Monolingual Corpora. In *Proceedings of the LREC-06*.

D. Radev, J. Otterbacher, A. Winkel, and A. Blair-Goldenson. 2005. NewsInEssence: Summarizing Online News Topics. *Communications of the ACM*, 48(10):95–98.

M. Richter, U. Quasthoff, E. Hallsteinsdóttir, and C. Biemann. 2006. Exploiting the Leipzig Corpora Collection. In *Proceedings of IS-LTC06*.

R. Steinberger, B. Pouliquen, and C. Ignat. 2005. Navigating multilingual news collections using automatically extracted information. In *Proc. of 27th International Conference on Information Technology Interfaces*, pages 25–32, Cavtat / Dubrovnik.

J. J. Thomas and K.A. Cook, editors. 2005. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Press.

C. Wayne. 2000. Multilingual Topic Detection and Tracking: Successful Research Enabled by Corpora and Evaluation. In *Proceedings of the LREC-00*, pages 1487–1494.

George K. Zipf. 1935. *The Psycho-Biology of Language. An Introduction to Dynamic Philology.* Houghton-Mifflin.