STORIES in time: a graph-based interface for news tracking and discovery

Bettina Berendt and Ilija Subašić
Department of Computer Science, K.U.Leuven, Leuven, Belgium
Email: firstname.lastname@cs.kuleuven.be

***Abstract*—We present the STORIES methods and tool for (a) learning an abstracted story representation from a collection of time-indexed documents; (b) visualising it in a way that encourages users to interact and explore in order to discover temporal "story stages" depending on their interests; and (c) supporting the search for documents and facts that pertain to the user-constructed story stages. In addition, we give an overview of evaluation studies of the tool.**

***Keywords*-news analysis; temporal text mining; burstiness; visualisation**

I. Introduction [a]

STORIES is a news summarisation approach that consists of story learning (done by the system) and graphical support for story understanding and story search (provided to the user). It (a) provides convenient interfaces to both the abstracted summary and the underlying documents; and (b) allows for and encourage a flexible, (inter)active exploration of the space of the abstracted "topics" or "stories" on the one hand, and searches of the space of the documents on the other hand. The paper builds on [1]; it describes a re-implementation with additional textual overview summaries, new interface features and examples from a new corpus.

II. Related work

Our work builds on several areas of research, whose relevance is described in detail in [1]: *temporal text mining* (TTM), e.g. [2], [3], [4], [5], *topic detection and tracking* (TDT) [6], [7], [8], *burstiness* [9], [10], [11], [12], and various summarisation approaches using co-occurrence, such as [10], [13]. We aim at more flexible content sub-structures than the fixed "topics" of TTM and TDT and apply measures of burstiness and co-occurrence to characterise these.

Further influences on our work are *News and other timelines on the Web*. Summarisation like that provided by Google Trends[1] shows surges in publication and query activity in certain time periods, but these tools require one to know which sub-topic to look for (and that this sub-topic is indexed). Google News Timeline[2] provides a pre-set time period (day, week, month, year) overview of news using a timeline interface. It allows for the tracking of news sources, arbitrary queries or entities such as movies, books, music, etc. The Yahoo! Correlator[3] associates a search term with all its related "events" (i.e. co-occurrences of this term or extracted entities with dates) and displays these in a timeline. Currently, it operates on the English Wikipedia corpus. MemeTracker[4] tracks quotes from the news and visualises their burstiness using interactive charts. Zoetrope [14] presents an interactive interface for tracking single DOM elements of an HTML page over time. By interacting with the Web at this atomic level, users can track and discover parts of a story without any linguistic processing. However, in the problem we study, users are less interested in a single document or its relations to the others than in learning about the underlying story.

Building on *visualisation* strategies as described, e.g. in [15], [16], [17], our tool presents users options for interactive summarisation, search and exploration.

III. Method

Story learning: The system learns a story from a corpus of time-indexed textual documents, all relevant to a top-level topic (the whole story, e.g., "Asia Tsunami 2004" or "Enron"). First, this corpus is transformed into a sequence-of-terms representation. Subsequently, the *content-bearing terms* are extracted, defined as the 150 top-TF (term frequency) terms. This defines a lower threshold on TF: $\delta \in \mathbb{R}$. Next, the corpus C is partitioned by publication periods, e.g. calendar weeks. Thus, C is the union of all document sets c_i, with $i = 1, \ldots, I$ the time periods.

For each c_i, the *frequency* of the co-occurrence of all pairs of content-bearing terms b_j in documents is calculated as the number of occurrences of both terms in a window of w terms, divided by the number of all documents in c_i. This measure of frequency and therefore relevance is normalised by its counterpart in the whole corpus to yield *time relevance* as the measure of burstiness: $TR_i(b_1, b_2) = (freq_i(b_1, b_2))/(freq_C(b_1, b_2))$. Thresholds are applied to avoid singular associations in small sub-corpora and to concentrate on those associations that are most characteristic of the period and most distinctive relative to others: $\theta_1 \in \mathbb{N}$ is a lower bound on the total number of co-occurrences, and $\theta_2 \in \mathbb{R}$, usually with $\theta_2 > 1$, is a lower bound on the time relevance of a co-occurrence. This gives rise to the *story graphs*

[1] http://www.google.com/trends

[2] http://newstimeline.googlelabs.com

[3] http://sandbox.yahoo.com/Correlator

[4] http://memetracker.org

$G_i = \langle V_i, E_i \rangle$ for time periods i. The edges E_i of G_i are $\{(b_1, b_2) | \#\ co\text{-}occ.s\ of\ b_1, b_2\ within\ w\ terms\ in\ doc.s\ from\ c_i \geq \theta_1\ and\ TR_i(b_1, b_2) \geq \theta_2\}$. The nodes V_i of G_i are the terms involved in at least one association in this symmetric graph: $\{b_j | \exists b_k : (b_j, b_k) \in E_i\}$. A sequence of story graphs forms the *story evolution*. To obtain a smoother story evolution, we use moving averages of co-occurrence frequency values, assigning to each period i the union over all documents from i to $(i + l - 1)$, for window size l.

Fact extraction and textual time-indexed summaries: From each document, we extract "facts", short statements with semantic role labelling, as returned by Open Calais.[5] The full set of these facts for each time period is indexed using Lucene.[6] We then use the story graphs to filter the most important facts: For each of the graph's edges, we query the index, using node names of the edge as query terms, and select the top fact as defined by Lucene's normalised TF.IDF scoring. We treat the resulting set of short textual statements as a summary of the story.

Story-space interaction: This usage interaction (see Section IV) rests on changes to the parameters: dates to specify i and l to track story evolution and "zoom in or out" of a story stage; changes to θ_2 and θ_1 and the removal of δ to "uncover" further details of a story stage.

Story search: Story search can be constrained by the nodes of a subgraph of the story graph. Retrieval is then restricted to documents relevant to these subgraphs. This is a form of query expansion similar to the method of [18]: the selection of documents of the starting corpus C corresponds to a top-level query; this query is expanded by the information from the subgraph and the time restriction. STORIES then uses all the nodes n as a query (restriction) for the documents inside c_i to obtain the pertinent document subset, as identified by a search over a Lucene index.

IV. TOOL

Story learning: We apply the method to news articles downloaded from different sources on the Web. Corpora can be compiled either on a continuous basis (e.g., subscribed-to feeds) or in response to a top-level query to a search engine. For example, the corpus can be a set of documents retrieved by a search in Google News or Blogdigger, usually in the archives of such search engines to control publication times. The top-level query describes the whole story (e.g., "Enron" or "*person_name*" for crime cases or celebrity reporting). Data cleaning and other data preparation steps are then applied, in particular HTML wrapper induction and removal, tokenisation, cross-document named-entity recognition, lemmatisation, and stopword removal. Finally, document and term measures as described in Section III are computed.

[5]http://www.opencalais.com/documentation/calais-web-service-api

[6]http://lucene.apache.org

Story-space interaction: The main goal of story-space interaction is exploration for sense-making. We implemented the methods as a Java application using the GUESS[7] visualisation library. The primary representations are visualisations of story graphs. They provide functionalities to

- *scan* over time to track the global story evolution (see Fig. 2). This corresponds to changing time period i. (The user may also inspect a morphing sequence that traces story evolution through all periods.)
- *zoom* by adapting the period-window size. Figure 1 illustrates the use of a dual slider for setting or changing the time period for which the graph is shown. By moving the slider to the left or right, the user can scan over time; by changing its length, she can zoom in to a shorter period and see a specific overview, or unzoom to a longer period and see a more general overview.
- *uncover:* Slide rulers allow the user to filter out story elements below individually set θ_1 (absolute number of occurrence of an association) or θ_2 (time relevance) thresholds. A configurable colour coding with different colour schemes accentuates time relevance differences.
- *track:* By selecting any node by mouse-clicking, the user can set a tracking focus on it. This outputs a graph of bursty co-occurrences that include this "tracking node" as one of its terms. The usual TF filter on terms δ gets disabled. This allows users to drill down beyond just the top nodes (see Fig. 2).
- *get a global overview:* By requesting the "facts" for the chosen time granularity, the user can get a textual overview of the whole story (see Fig. 2).

Story search: Users may select an edge and then highlight a subgraph which contains the selection's adjoining edges and neighbouring nodes (Fig. 1). Each selected edge expands the query by adding its nodes to the query 'shopping cart', as long as the query has fewer than six distinct terms, a common upper bound on query size. In this way the user incrementally builds the query and at the same time can discover and learn about the story.

Search provides an (internally unstructured) list of documents. In addition, we are currently experimenting with an extension of the interface by navigation in document space based on a visualisation of multidimensional similarity [19].

V. EVALUATION ASPECTS

We evaluated the story graphs' *information-retrieval capabilities* using two examples, a missing-persons case and celebrity reporting [1]. We established a ground truth by extracting time-indexed "real" events from the associated Wikipedia article and a 'celebrity-watch' Web site, respectively. Wikipedia in particular was chosen to obtain a ground truth that can be expected to have high inter-rater agreement. To find out whether and which parts of the story-graph

[7]http://graphexploration.cond.org

summaries capture that ground truth, we first optimised the method's parameters by an (automated) technique used for the evaluation of textual summarisations [20], and then asked raters to (manually) map story-graph edges to the ground-truth events (both within the same time period).

While recall was quite high (up to 80% of all events were judged as being represented in the graph), precision was much lower (no more than about 33% of edges were thought to represent ground-truth events). This raises the question of whether it would be possible to select "more meaningful" parts of the overall story evolution and of the individual story graphs, in order to make a concise and sensible summary available to users. Ideally, the meaningful parts would be automatically computable *and* stand out in the visualisation so as to attract attention and entice users to search by them, while the others could be left in the picture to provide context. We initially concentrated on *story-graph topological properties as possible predictors of events*. A first analysis [19] showed that global properties of the graphs like graph size and number of connected components as well as the existence of nodes with high degree centrality were useful for predicting "real" eventfulness of a period.

Fact extraction is comparable in quality to current multi-document summarisation approaches.[8] Evaluations of *search quality* [1] demonstrated that the STORIES search finds coherent subsets of documents, that its quality is comparable to or better than state-of-the-art clustering, and that the tool enables people to answer questions on ground-truth events accurately and quickly.

VI. Outlook

In future work, we will investigate more advanced language processing (linguistic parsing, semantic role labelling, etc.), the use of lexical resources and other background knowledge, as well as different sources of media bias/viewpoints. We also plan to explore aggregation and analysis dimensions other than time, such as multilinguality, as investigated in ElectionWatch[9] or Found in Translation[10]. Further quantitative and qualitative evaluations will be carried out. We will also study user behaviour and attitudes towards different forms of story understanding and search.

References

[1] I. Subašić and B. Berendt, "Discovery of interactive graphs for understanding and searching time-indexed corpora," *Knowledge and Information Systems*, in press.

[2] Q. Mei and C. Zhai, "Discovering evolutionary theme patterns from text: an exploration of temporal text mining," in *KDD'05*, ACM, 2005, pp. 198–207.

[3] R. Schult and M. Spiliopoulou, "Discovering emerging topics in unlabelled text collections," in *ADBIS 2006*, LNCS 4152. Springer, 2006, pp. 353–366.

[4] X. Wang and A. McCallum, "Topics over time: a non-markov continuous-time model of topical trends," in *KDD '06*. New York, NY, USA: ACM, 2006, pp. 424–433.

[5] C. C. Chen and M. C. Chen, "TSCAN: a novel method for topic summarization and content anatomy," in *SIGIR '08*. New York, NY, USA: ACM, 2008, pp. 579–586.

[6] J. F. Allan, *Topic Detection and Tracking*. Berlin etc.: Springer, 2002.

[7] T. Štajner and M. Grobelnik, "Story link detection with entity resolution," in *Proc. of Semantic Search'09 Workshop at WWW 2009*, 2009.

[8] R. Nallapati, A. Feng, F. Peng, and J. Allan, "Event threading within news topics," in *CIKM '04*. New York, NY, USA: ACM, 2004, pp. 446–453.

[9] J. M. Kleinberg, "Bursty and hierarchical structure in streams," *Data Mining and Knowledge Discovery*, vol. 7, no. 4, pp. 373–397, 2003.

[10] G. P. C. Fung, J. X. Yu, P. S. Yu, and H. Lu, "Parameter free bursty events detection in text streams," in *VLDB '05*. VLDB Endowment, 2005, pp. 181–192.

[11] D. Gruhl, R. V. Guha, R. Kumar, J. Novak, and A. Tomkins, "The predictive power of online chatter," in *KDD'05*, ACM, 2005, pp. 78–87.

[12] Q. He, K. Chang, E.-P. Lim, and J. Zhang, "Bursty feature representation for clustering text streams," in *SIAM 2007*. SIAM, 2007.

[13] R. Choudhary, S. Mehta, A. Bagchi, and R. Balakrishnan, "Towards characterization of actor evolution and interactions in news corpora," in *ECIR 2008*, LNCS 4956. Springer, 2008, pp. 422–429.

[14] E. Adar, M. Dontcheva, J. Fogarty, and D. S. Weld, "Zoetrope: interacting with the ephemeral web," in *UIST '08*. New York, NY, USA: ACM, 2008, pp. 239–248.

[15] C. Chen, *Mapping Scientific Frontiers*. London: Springer, 2003.

[16] F. A. L. Janssens, W. Glänzel, and B. D. Moor, "Dynamic hybrid clustering of bioinformatics by incorporating text mining and citation analysis," in *KDD'07*. ACM, 2007, pp. 360–369.

[17] L. Leydesdorff and T. Schank, "Dynamic animations of journal maps: Indicators of structural change and interdisciplinary developments," *JASIST*, vol. 59, no. 11, pp. 1810–1818, 2008.

[18] B. M. Fonseca, P. Golgher, B. Pôssas, B. Ribeiro-Neto, and N. Ziviani, "Concept-based interactive query expansion," in *CIKM '05*. New York, NY, USA: ACM, 2005, pp. 696–703.

[8] using the dataset and baselines of DUC 2007, http://www-nlpir.nist.gov/projects/duc/past_duc_aquaint/duc2007/results/

[9] http://electionwatch.enm.bris.ac.uk

[10] http://foundintranslation.enm.bris.ac.uk

[19] B. Berendt and I. Subašić, "Measuring graph topology for interactive temporal event detection," *Künstliche Intelligenz*, vol. 02/09, pp. 11–17, 2009.

[20] C.-Y. Lin, "Rouge: a package for automatic evaluation of summaries," in *Proc. WAS Workshop*, 2004.

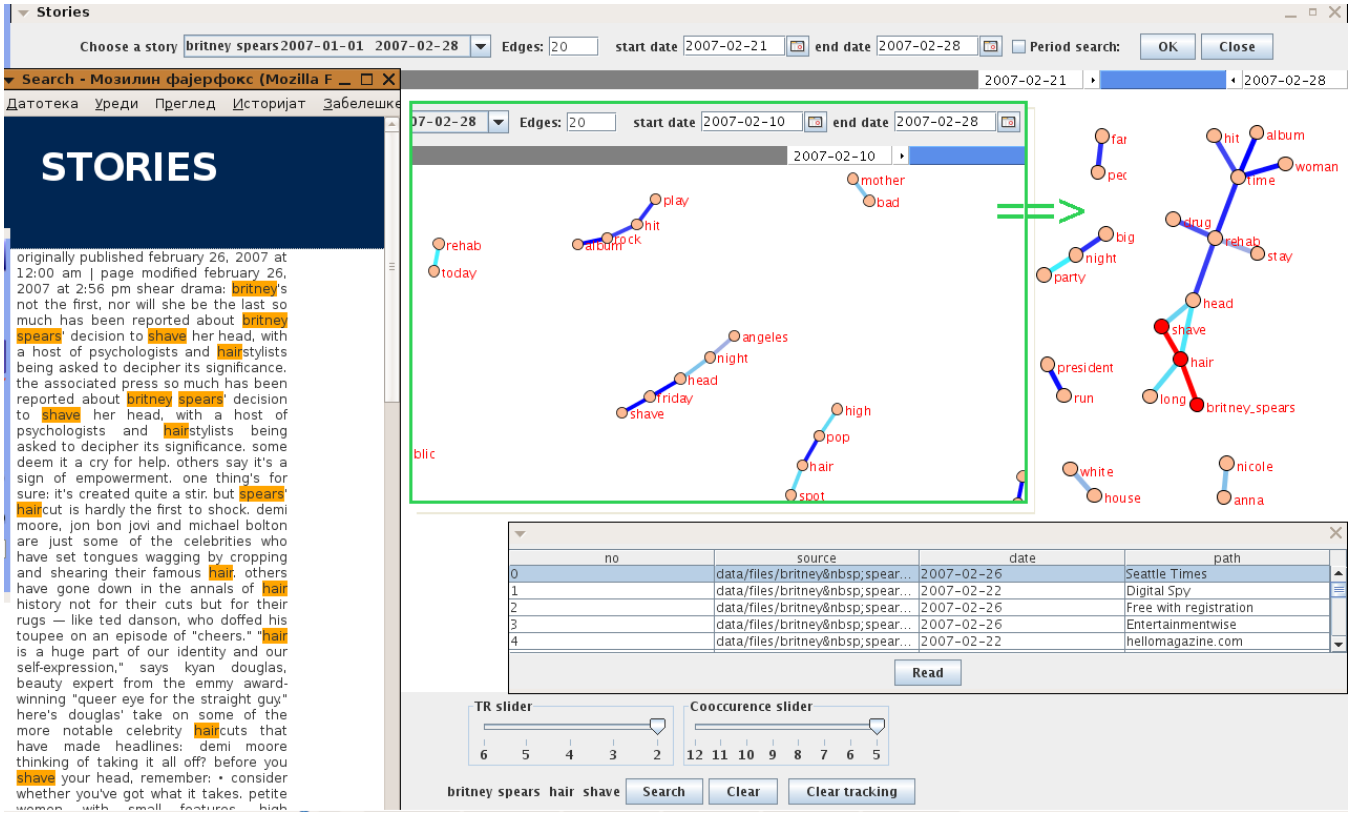

Figure 1. Story understanding by zooming in time and story search: The main window contains the story graph. *Centre, top:* The screenshot fragment in the green box on the left side of the main window shows a first story graph for a corpus of reports on Britney Spears from 2007. A wide time window contains, among others, seemingly unrelated story lines on "head", "shave", and "rehab". *Right:* Narrowing the time window reveals that it was Britney Spears who shaved her head (and that there is a possible connection to her entering rehab). By marking the edges connecting the three terms, the user obtains a list of pertinent documents (*centre, bottom*), whose text can be inspected (*left*).

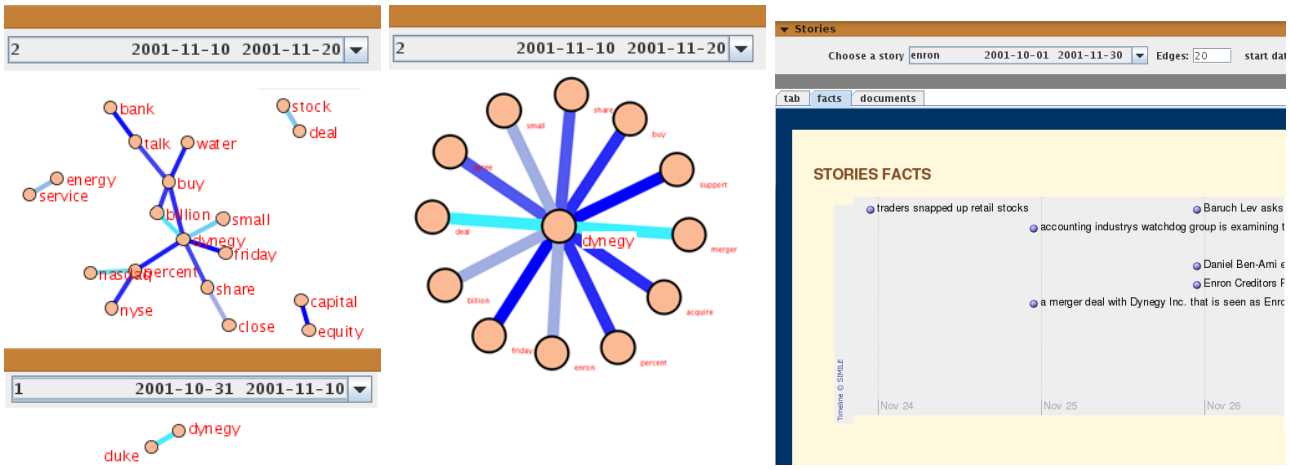

Figure 2. Scanning and tracking: In a corpus of reports on Enron from 2001, the company Dynegy appears first in period 2 (*left, top*: part of the story graph). Marking the node for tracking removes the TF filter. Going back to period 1 shows that it was already mentioned there, but only very infrequently (*left, bottom*). Then scanning forward to period 2 shows a wealth of associations associated with Dynegy's planned takeover of Enron, which became public then (*middle*; "dynegy" label enlarged for readability). *Right:* a part of the facts timeline.