Novalist: Content Reduction for Cross-media Browsing

Franciska de Jong
Department of Computer Science
University of Twente
Enschede - The Netherlands
fdejong@ewi.utwente.nl

Wessel Kraaij
TNO
Delft - The Netherlands
wessel.kraaij@tno.nl

Abstract

The Novalist news browser described in this paper was built in order to experiment with the integration of a number of tools for the disclosure of multimedia news content via a flexible and multifaceted browser. The tools available for integration cover a wide range functionalities, including clustering, classification, extraction of headlines and proper names, indexing and summarization. They each contribute to the presentation of content in a reduced form to support efficient navigation and selection. The paper describes some of them detail and outlines the demonstration system to which they contribute.

1 Introduction

In various domains, information analysts have to deal with large amounts of information which is refreshed on a daily basis and disseminated via various media types: traditional newspapers, news wires and magazines, internet sites and also television broadcasts via air or cable. Analysis and monitoring of these open news sources, which in some cases are coupled to non-public sources of information, is often crucial for efficient and effective workflow. Various text mining tools can support the task of news analysts.

The Novalist news browser described in this paper was built in order to experiment with the integration of a number of tools for the disclosure of news content and to provide access to it via a flexible and multifaceted browser. The tools available for integration cover a wide range of functionalities, including clustering, classification, extraction of headlines and proper names, indexing and summarization. One of the most distinguishing features of the disclosure system is that it integrates a collection of news content from various open sources and in multiple formats: text, audio and video. Another salient features of the browsing environment is that at all times links are available to the underlying documents via which the content can be read, played or viewed. Also distinguishing is the fact that the content can be accessed at various levels of abstraction. The latter feature is due to the application of a variety of summarization tools.

This paper will give an overview of the components of the Novalist system in section 3, while in section 4 and section 5 the modules for respectively content abstraction and summarization will be described in more detail. The perspective on evaluation, plus some concluding remarks are presented in section 6. The remainder of the paper starts with some background information and related work.

2 Background and related work

Novalist aims to facilitate the work of information analysts in the following way: (i) related news stories are clustered to create dossiers, sometimes also called 'threads', (ii) dossiers resulting from clustering are analysed and annotated with several types of metadata, and (iii) a browsing environment provides multiple views on the dossiers and their metadata. The basis for the dossiers is formed by news content items from open sources such as journals and news broadcasts. The applied clustering generates structure in news streams, while the annotations can be applied as filters: search for relevant items need not to apply on unanalysed data but be constrained to relevant subsets of the collection. Novalist supports the fast identification of relevant dossiers during browsing and also supports visual browsing of the clusters, along with their extracted headlines. Dossiers are visualised in a compact overview window with links to a time axis.

As said, in addition to clustering, all kinds of metadata-extraction techniques can help to enrich the generated dossiers and/or to provide suitable representations at various levels of abstraction and granularity. Additional functionality could consist of the automatic generation of links to related sources, both internal and external. Similar dossier generation applications, with topic clus-

tering as basis and content reduction (or summarization) as additional functionality, could be applied in other domains, and/or for other combinations of media. In addition to text from newspapers and autocue files, transcripts for broadcast audio generated with automatic speech recognition (ASR) could be taken into account, or content from newswire websites. Assuming that the material can be properly segmented, such sources could be linked to the related topical clusters. In other domains than news, e.g., oral history archives, meeting or lecture recordings and digital story telling, ASR is a likely source of text, while combination with manually generated minutes, historical studies, policy plans, etc., can provide the required additional perspective on the recorded content. For the field of meeting modeling, which is rapidly advancing, one could also think of exploiting various types of labeled segmentations generated via either manual annotation or via automated processing (Carletta *et al.* 05).

The exploitation of so-called collateral text is also highly relevant and feasible for sports commentaries, which tend to come from mutltiple sources in a variety of languages. Here the potential added value of clustering is that a series of parallel texts, identified on the basis of e.g., a word-based profile, could be reduced to one coherent report without redundancies. Cf. (Kuper *et al.* 03). In general, for non-scripted audio and video content ASR is a major source of text, (de-Jong 04), and the generation of textual summaries is seen as a valuable contribution to the usability of browsers for this type of content. Cf. (Buist *et al.* 05).

3 Overview of components

In this section we give an outline of the Novalist architecture and content base, followed by sections that explain the concept of topic detection and the similarity concept applied in the language modeling approach that is underlying several content-processing components.

3.1 Demonstrator set-up

The Novalist architecture consists of two separate applications:

1. the disclosure module, working offline
2. the actual Novalist news browser, which offers end-users search and browsing functions in the enriched news database.

The corpus disclosed by the demonstrator system consists of a collection of news items published by a number of major Dutch newspapers, magazines, web crawls, a video corpus of several news magazines and a video archive with all 2001 broadcasts of *NOS Journaal*, the daily news show of the Dutch public TV station. The autocue files for the video archive function as collateral text, i.e., text that is not the primary target of search, but that supports the disclosure of video via the time links to media fragments. The entire collection consists of some 160,000 individual news items from 21 different sources.

The disclosure architecture consists of three steps. In the first step, textual information is extracted from audiovisual material. This could be either closed captioning text or ASR transcripts. These textual extracts as well as the other textual data sources (e.g. newspapers, website crawls and magazines) are converted to a single XML representation, suitable for further processing. The second step is the structuring, which is implemented as an unsupervised clustering procedure. The clustering algorithm has been designed to be scalable, by limiting its resources to a time-limited selection of documents. This seems fair, since related news items are often published in the same period. The clustering step results in a set of disjunctive clusters, i.e. each news item is a member of just one cluster. The cluster algorithm will be described in more detail in the next section. The last step of the disclosure process is aimed at labeling each cluster in order to facilitate browsing and search, or more generally: navigation through indices of different types.

Eventually the types of automatically assigned metadata which Novalist can exploit for navigation are: keywords, proper names, thesaurus terms, headlines and extractive summaries. In addition to search on automatically generated metadata, analysts may search on full text, or browse publications issued on a certain date or in a certain period. This paper focuses on the disclosure module, but a screendump of the end-user application is included to illustrate the browser functionality.

3.2 Topic clustering

The clustering in Novalist is based on topic detection technology, which groups stories that discuss

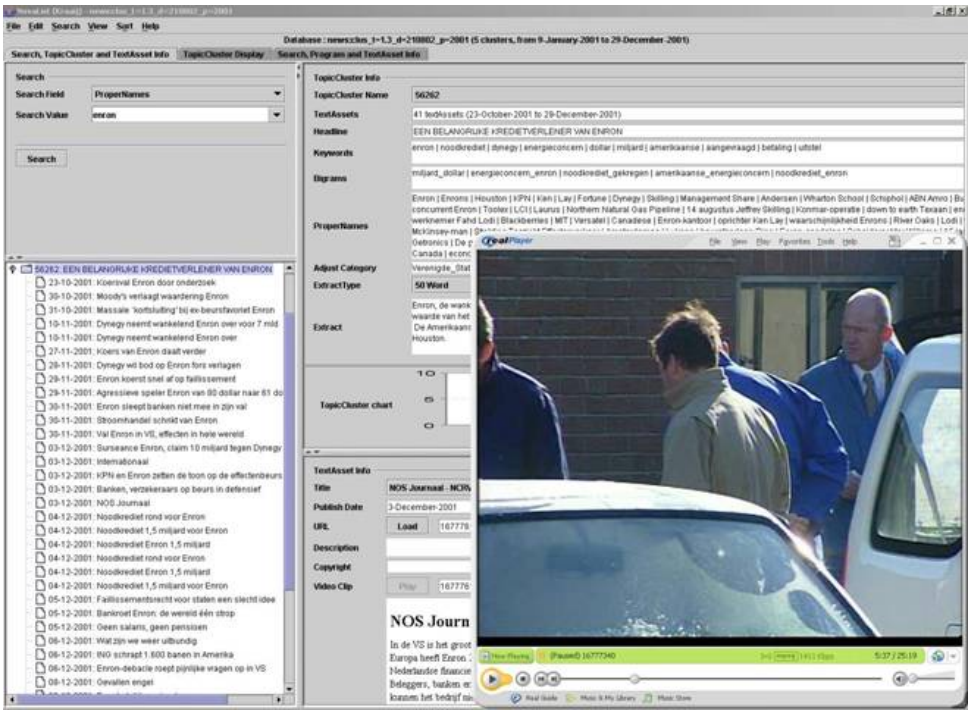

Figure 1: Novalist: browsing multimedia dossiers through associated metadata

the same event. Clustering is done incrementally: for a new incoming story, the system has to decide instantaneously to which topic cluster the story belongs. Since the clustering algorithms are unsupervised, no training data is needed. The clustering technology of Novalist has been tested in the context of the evaluation event for topic detection systems organised by NIST (TDT2001).[1] The topic detection component of Novalist is derived from the topic detection system developed and trained with the TDT3 corpus (80.000 stories, 120 topics).

The technique deployed is called topic *detection* (or topic discovery), since the system has to deal with dynamic information, about which no full prior knowledge is available. There is no fixed number of target topics and events types. The system must both discover new events as the incoming stories are processed, and associate incoming stories with the event-based story clusters created so far.

For the grouping of stories we combined a simple single pass method to establish an initial clustering and a reallocation method to stabilize the clusters within a certain allowed deferral period (Spitters & Kraaij 02). The combination of a cluster initialization step and a reallocation step was previously (successfully) applied for topic detection by a.o. BBN (Walls *et al.* 99) and Dragon (Yamron *et al.* 00). The similarity of an incoming story S_n to an existing cluster C is defined as the average of the similarities of S_n to each story $S_i \in C$. These individual similarities are computed by taking the sum of the generative probabilities $P(S_n|S_i)$ and $P(S_i|S_n)$ where S_i and S_n are modeled as unigram language models. Because these story language models are based on extremely sparse statistics, the word probabilities are smoothed using a background model.

A cluster which has not changed for an uninterrupted period of fifteen days is frozen, which means that it is no longer considered an 'active event'. The cluster is removed from the list of candidate clusters for new stories. This cluster evolution monitoring has two advantages. First of all it limits the computational complexity, because the number of clusters a story has to be compared with stays within certain bounds. Second, it can be argued that restricting the temporal extent of an event is beneficial for detection performance because it prevents different events with similar vocabulary (like different attacks or political elections) to be grouped together (Yang *et al.* 99).

The NIST evaluation of the cluster algoritm

[1] For details on the NIST evaluation of TDT, cf. http://www.nist.gov/speech/tests/tdt. See (Allan 02) for a detailed overview of the TDT project.

showed that Novalist produces highly precise, small clusters. The granularity of the clusters depends on a single fixed parameter, which makes the system a bit inflexible, since the chosen granularity might not always coincide with the preferred granularity that an information analyst needs for a certain task. To remedy this problem, a new version of the cluster algorithm has been designed which produces a hierarchical clustering thereby offering multiple levels of cluster granularity. This system has been evaluated at TDT 2004 (Trieschnigg & Kraaij 05).

3.3 Language model-based similarity

The basic idea behind the language modeling approach to information retrieval is to estimate a (usually unigram) language model for each document and to rank documents by the probability that the document model generated the query. Absolute probabilities are not important for ranking in the IR situation. For other applications, i.e. topic tracking and also topic detection, scores have to be comparable on an absolute scale. For tracking, we found that modeling similarity as a likelihood ratio and normalizing this likelihood ratio by the (test) story length was adequate (Kraaij & Spitters 03). This normalized likelihood ratio is presented in equation (1), where $\mathrm{LLR}_{Norm}(T_1, T_2, ..., T_n|S_k)$ denotes the normalized log likelihood ratio of a story consisting of the terms $T_1, ..T_n$ given the story S_k in comparison with background model $\mathcal{B}$.

$$\mathrm{LLR}_{Norm}(T_1, T_2, ..., T_n|S_k) = \frac{1}{n}\sum_{i=1}^{n} \log \frac{P(T_i|S_k)}{P(T_i|\mathcal{B})} \quad (1)$$

In our clustering approach, the similarity between two stories S_n and S_i is based on a combination of the probability that the language model representing S_n generated story S_i and the reverse: the probability that the language model representing S_i generated story S_n. This approach results in the symmetrical similarity measure, presented in the following equation:

$$Sim(S_n, S_i) = \mathrm{LLR}_{Norm}(S_n|S_i) + \mathrm{LLR}_{Norm}(S_i|S_n) \quad (2)$$

Because the language models are estimated on very limited amounts of text (single stories), it is very important that the word probabilities are smoothed using some background model.

4 Content abstraction

Effective content abstraction is a key feature for improved efficiency of the information analysis task. In this context the notion 'abstraction' refers both to conceptual structure, as well as to (reduced) content size. In Novalist the automatically extracted/assigned metadata function as descriptors for a dossier and allow their presentation in a highly condensed form. Various useful levels of abstraction can be distinguished, as different analysis tasks may impose different requirements on the level of conciseness, and even different perspectives on the content can correspond to different metadata requirements. For example, a proper name index on a cluster gives another perspective as a list of topic labels generated by thesaurus-based classification. Novalist demonstrates that multiple document abstractions effectively mediate different levels of granularity analysis. Metadata types such as keywords and headlines help the user to select potentially interesting clusters for further inspection. This more detailed inspection step can subsequently involve looking at the titles of the individual news items and reading a multi-document extract.

Though content abstraction implies content reduction, the reverse only holds if the reduced data (summaries, headlines) are representative from one or more perspectives. In this section we will describe a number of abstraction techniques that do not yield representations in running text. Though an absolute distinction between the abstraction and reduction would not make sense, we reserve an explanation of the role of reduced textual representations for the next section.

4.1 Proper name extraction

In news content, proper names are the type of cues that link the content to high level background knowledge. It is the type of information also occurring in classical indexing systems that exploit lists of index terms. Therefore a list of all proper names occurring in a dossier can function as an abstract layer over the content. The named entity recognizer developed at TNO is based on a hybrid framework combining a maximum entropy machine learning approach with hand-crafted rules as a post-editing step.

The maximum entropy model is able to combine a large set of contextual features without assuming feature independence. The combined approach reduces development time (developing rules for named entity detection is a time consuming process) but retains the flexibility of a rule-based framework, which is useful to quickly tune the recognizer to a new domain.

4.2 Automatic topic classification

In general, searching for manually assigned thesaurus terms is much more precise than full text search. In Novalist dossiers can be labeled automatically to clusters by the incorporation of ADJUST, an automatic classification system developed at TNO. The labels (terms from a general news thesaurus) are assigned based on interpretation of the content in terms of statistical profiles rather than just extraction of terms without processing the context. The ADJUST functionality allows users to navigate within a document base by browsing through the terms in a hierarchical concept directory (thesaurus). Assignment of such labels does not presuppose that homophonous words occur in the dossier. This classification functionality requires the availability of a pre-classified training set.

4.3 Keyword extraction

When a topic classification system is not available, simple keywords or keyphrases can be quite helpful for characterization of the cluster content. Novalist applies metrics which are related to the KL-divergence between language models estimated on the cluster and a collection model for the selection of distinguishing terms.

5 Summarization: content reduction

As said the concept of 'summarization' refers to data reduction over the original document sources. The shorter descriptions are commonly representations in the form of one or more sentences. In this section we will describe two techniques to create them: extractive summarization and headline generation.

5.1 Extractive summarization

The extraction-based summarization component of Novalist is based on a machine learning approach and has been developed and evaluated using datasets available for the DUC 2001 and 2002 benchmarking workshops organized by NIST. Cf. results (Kraaij *et al.* 01; Kraaij *et al.* 02). A corpus is needed with examples of texts and their summaries.

The system applies a simple Naive Bayes classifier for the selection of salient sentences from a set of documents. A variety of continuous and discrete features has been used (where the continuous feature are binned into discrete classes), ranging from unigram language model score to the presence of cue phrases or sentence length(Kraaij *et al.* 01). The effectiveness of several features was demonstrated in earlier work e.g., (Kupiec *et al.* 95; Edmundson 69).

The combined model determines a salience value for each extracted sentence. This ranked list of sentences forms the input for the summary generation module. This module tries to generate a summary which consists of the most salient sentences, with minimal redundancy and maximal coherence/readability.

5.2 Headline extraction

Instead of producing a running text as summarization for a dossier, navigation and selection can also be supported by the type of summary that usually is called 'headline'. For the task of assigning a headline to each cluster, we deployed a system for extractive summarization: a combination of identifying salient sentences and extraction of noun phrases (NPs).

The first step is to locate the NPs describing the cluster topic. For this purpose a 'trigger word' must be chosen. To decide on the trigger word first a 'trigger word pool' is generated by automatically summarizing each single document from a cluster and taking the highest ranked sentence for every document. To this pool the titles of the documents are added. The word frequency in the pool is calculated, using the same stoplist (which is rather extensive) as used in the actual summarization. The highest ranked word is marked as the trigger word for the cluster at hand.

The second step decides which NP containing the trigger word is the most appropriate headline. Applying multi-document summarization to the cluster yields a ranked order of the most salient cluster sentences. From these the NP including the trigger word is selected as headline. As to not get too short or too long headlines, the optimal NP length is between two and ten words, not counting determiners. If in one sentence two NPs contain the trigger word, the longest is se-

lected. The same goes for a draw between the trigger words. All documents contribute equally to the trigger pool.

Experiments with stemming indicated that it only affected the selection of NPs, and we preferred the headlines selected without stemmed trigger words. The biggest challenge is the selection of the most appropriate of the potential NPs and to avoid too specific headlines. The latter is even more important when the cluster theme is more general than the content of the individual documents.

6 Evaluation perspective and conclusion

Some of the components of Novalist have been evaluated in formal evaluations on unseen data. The extractive summarization has been evaluated at DUC 2001 and 2002 and outperformed baseline lead based summaries. The clustering component has been evaluated during TDT 2001 and achieved best results (Spitters & Kraaij 02). A more comprehensive evaluation of Novalist is currently carried out with real users. The user study comprises interviews and logfile analysis in order to measure whether information analysts can improve the effectiveness and/or efficiency of their search tasks. It also addresses the question which is the best combination of metadata to be offered to the user during navigation. The preparatory work also comprised scaling the system to a larger document collection.

The concept of cross-media browsing for the use scenario's described here would get an even wider perspective if image features could also be taken into account. But though the ambition to link low-level features to higher level or even semantic annotation layers figures on many research agendas, generally applicable methods haven't been delivered. For the time being the Novalist approach that exploits textual resources of whatever kind and source, illustrates how this gap can be filled.

Acknowledgements

This work was partly supported by Dutch Telematics Institute project DRUID, the EU project InDiCo (IST-FP5-34306) and the EU project AMI (IST-FP6-506811).

Over the years, the following people have contributed to Novalist: Martijn Spitters, Anette Hulth, Harry Wedemeijer, Dolf Trieschnigg, Stephan Raaijmakers.

References

(Allan 02) James Allan, editor. *Topic Detection and Tracking.* Kluwer academic publishers, 2002.

(Buist *et al.* 05) A.H. Buist, W. Kraaij, and S. Raaijmakers. Automatic summarization of meeting data: A feasibility study. In *proceedings of the 15th CLIN conference*, 2005. forthcoming.

(Carletta *et al.* 05) J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner. The ami meetings corpus. In *Proceedings of the Measuring Behavior 2005 symposium on "Annotating and measuring Meeting Behavior"*, 2005.

(deJong 04) F.M.G. de Jong. Disclosure of non-scripted video content: InDiCo and M4/AMI. In *Proceedings of CIVR2004*, volume 3115 of *Lecture Notes in Computer Science*, pages 647–654. Springer Verlag, 2004. ISBN=3-540-22539-0.

(Edmundson 69) H.P. Edmundson. New methods in automatic abstracting. *Journal of the Association for Computing Machinery*, 16(2), 1969.

(Kraaij & Spitters 03) Wessel Kraaij and Martijn Spitters. Language models for topic tracking. In Bruce Croft and John Lafferty, editors, *Language Models for Information Retrieval.* Kluwer Academic Publishers, 2003.

(Kraaij *et al.* 01) W. Kraaij, M. Spitters, and M. van der Heijden. Combining a mixture language model and naive bayes for multi-document summarisation. In *Proceedings of the DUC2001 workshop (SIGIR2001), New Orleans*, 2001.

(Kraaij *et al.* 02) W. Kraaij, M. Spitters, and A. Hulth. Headline extraction based on a combination of uni- and multidocument summarization techniques. In *Proceedings of the ACL workshop on Automatic Summarization/Document Understanding Conference (DUC 2002).* ACL, June 2002.

(Kuper *et al.* 03) J. Kuper, H. Saggion, H. Cunningham, T. Declerck, F.M.G. de Jong, D. Reidsma, Y. Wilks, and P. Wittenburg. Intelligent multimedia indexing and retrieval through multi-source information extraction and merging. In *18th International Joint Conference of Artificial Intelligence (IJCAI)*, pages 409–414, Acapulco, Mexico, 2003.

(Kupiec *et al.* 95) Julian Kupiec, Jan O. Pedersen, and Francine Chen. A trainable document summarizer. In *SIGIR'95, Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Seattle, Washington, USA.*, pages 68–73. ACM Press, 1995.

(Spitters & Kraaij 02) Martijn Spitters and Wessel Kraaij. Unsupervised clustering in multilingual news streams. In *Proceedings of the LREC 2002 workshoop: Event Modelling for Mu ltilingual Document Linking*, 2002.

(Trieschnigg & Kraaij 05) Dolf Trieschnigg and Wessel Kraaij. Hierarchical topic detection in large digital news archives. In *Proceedings of the 5th Dutch Belgian Information Retrieval workshop (DIR)*, 2005.

(Walls *et al.* 99) F. Walls, H. Jin, S. Sista, and P. van Mulbregt. Topic detection in broadcast news. *Proceedings of the DARPA Broadcast News Workshop*, 1999.

(Yamron *et al.* 00) J.P. Yamron, S. Knecht, and P. van Mulbregt. Dragon's tracking and detection system for the TDT2000 evaluation. *Notebook papers of the Topic Detection and Tracking Workshop (TDT) 2000*, 2000.

(Yang *et al.* 99) Y. Yang, J. Carbonell, R. Brown, T. Pierce, B.T. Archibald, and X. Liu. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems: Special Issue on Applications of Intelligent Information Retrieval*, 14(4):32–43, 1999.