MyNewsWave: User-centered Web search and news delivery

Clint Heyer, Jamie Madden, Kelly Hollingsworth,
Peter Heydon, Keiran Bartlett, Joachim Diederich

Information Environments Program
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane Q 4072, Australia

clint@thestaticvoid.net
jamie@itee.uq.edu.au
keiranbartlett@keiranb.cjb.net

kelly@infenv.com
heydonp@acslink.net.au
joachimd@itee.uq.edu.au

Abstract

MyNewsWave uses machine learning (including support vector machines) for a user-centred approach to full-text information retrieval as well as news delivery. The system uses knowledge sources such as WordNet to refine keyword queries and learns user-preferences with regard to web search. MyNewsWave includes an audio mining system for topic detection in conjunction with background search to facilitate the retrieval of relevant multimedia information.

A special feature of MyNewsWave is the assessment of incoming information with regard to the "mood" or personal relevance to a user. DigiMood is a component of MyNewsWave that classifies web pages into mood categories. Business news, for instance, can be classified by DigiMood to access market sentiment. Marconi analyses incoming news streams and uses machine learning to adjust parameters of a text-to-speech system. The objective is to learn the appropriate voice for news items as part of a speech user interface.

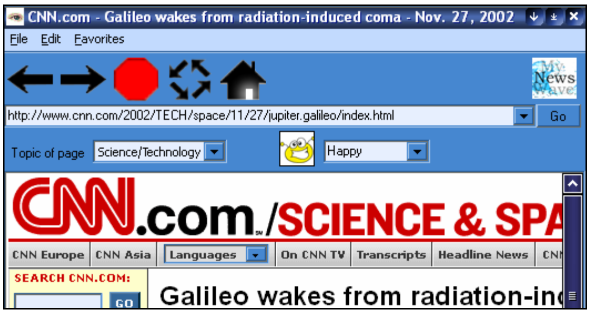
Keywords Multimedia resource discovery, Personalised documents, information retrieval.

1 Overview

We aim at integrating information search and delivery by use of machine learning systems that allow adaption to an individual user. MyNewsWave has five parts: (1) The *Tibianna* search engine uses support vector machines (SVMs) for learning ranking functions utilising user feedback and ontological knowledge. (2) *DigiMood* assesses web pages with regard to the mood expressed in the document. (3) *Peeping Tom* is a topic classification and user modelling system that classifies documents into categories relevant to the user. (4) The delievery component *Marconi* provides a *speech user interface* that learns to select voices based on user preferences. (5) The *Emily* audio mining system performs background search and analyses audio files to retrieve additional information.

Our aim is to combine active web search, including background research for related audio and textual information, with a user-centered approach to information delivery. Most components of MyNewsWave are implemented and are currently being tested. A user interface similar to a standard web browser integrates the subsystems (Figure 1).

A full introduction of all five MyNewsWave components is beyond the scope of this paper. Hence, the focus here is on text mining by use of machine learning techniques as used in the Tibianna and DigiMood subsystems.


CNN.com - Galileo wakes from radiation-induced coma - Nov. 27, 2002

http://www.cnn.com/2002/TECH/space/11/27/jupiter.galileo/index.html

Topic of page Science/Technology

Happy

CNN.com/SCIENCE & SPA

Galileo wakes from radiation-inc

Figure 1: The MyNewsWave browser.

2 Introduction: Web search and machine learning

The process by which a user can refine a search is an emerging research field with many approaches and techniques. Tomita & Kikui [1] use graphical query refinement, whereby a *query graph* is created from

**Proceedings of the 7th Australasian Document Computing Symposium,
Sydney, Australia, December 16, 2002.**

the user's search query. For each search result, a *subject graph* is created, and its similarity to the query graph determines the ranking. The user hones their search by directly manipulating the query graph as well as indicating if a returned document is relevant.

Most search engines use a global relevance ranking which is linked to the query and does not take into account the users' subjective *value* of a resource [2, 3]. This value metric is not necessarily encapsulated in search queries and documents returned from identical queries by two users may have entirely different values. Better capturing this value metric should increase the precision of the search.

Glover [2] uses an 'information need' query modifier, which refines search queries to return resources of a certain type, for example research papers, or personal home pages. This modifier allows the engine to extract more qualifiers for the search, without the user having to think about structuring the query, therefore further improving precision.

Bruza & McArthur [4] take a user-centred approach by empirically comparing various methods for query modification: standard searching, phrase-based query reformulation and hierarchical directory browsing. One of the query reformulation methods investigated takes advantages of *WordNet*[1], a large lexical database. Moldovan & Mihalcea [5] further investigate using WordNet and describe various algorithms for using an original user's query and WordNet reference data to restructure a user's search query, resulting in significant improvement in performance.

A metasearch engine is a search tool that combines the results of one or more external search engines, applying its own ranking function and then presenting the hybrid results to the user [6]. Whereas some metasearchers rely only on the results pages from the source search engines to form the meta result set, Lawrence & Giles [7] present a metasearch engine, *NECI*, which downloads the top ranked pages from each source, and performs its own analysis. This extra layer of analysis can help to weigh differences in page ranking algorithms used by the various engines, and provides a final, consistent ranking for all engines. NECI also transforms particular user queries into a style that is more likely to be present in a web page. For example, transforming "What does NASDAQ stand for?" into "NASDAQ stands for", "NASDAQ is an abbreviation" and "NASDAQ means". These queries are searched in parallel with the user's original search, with the combined results shown to the user.

2.1 Web search and support vector machines

Machine learning can be used to improve search results. Cortes & Vapnik [8] introduce support vector machines which are a novel approach to machine learning. Support vector machines are based on the structural risk minimisation principle. Support vector machines find the hypotheses out of the hypothesis space H of a learning system which approximately minimises the bound on the actual error by controlling the VC-dimension of H. SVMs are very universal learning systems [9]. In their basic form, SVMs learn linear threshold functions. However, it is possible to "plug-in" kernel functions so that they can be used to learn polynomial classifiers, radial basis function (RBF) networks and three or more layered neural networks.

The most important property of SVMs for text classification is that learning is independent of the dimensionality of the feature space [9]. SVMs evaluate hypothesis by use of the margin they use for separating data points, not the number of features or attributes. This allows for good generalisation even in the presence of a large number of features. Joachims [10] lists the following reasons why SVMs are a preferred method for learning text classifiers:

(1) SVMs can process high-dimensional input spaces: If every word of a text is a feature, the input space can easily be larger than 100,000. SVMs control overfitting internally and, therefore, large feature spaces are possible.

(2) Few irrelevant features: Feature selection is normally used to avoid input spaces of high dimensionality. In text classification, this is either not practical or many features are equally important. Therefore, SVMs are a convenient way to learn a text classifier with limited preprocessing.

(3) Document vectors are sparse: For the reasons mentioned above, SVMs are ideally suited for sparse input vectors of high dimensionality.

(4) Most text categorisation problems are linearly separable: This has been empirically determined by a number of authors.

Glover et al. [11] describe a system that uses SVMs in conjunction with learned query refinement to increase search relevancy. This approach looks at categorising resources into groups such as papers, research papers and product reviews. Their SVM is trained on the top 100 features from each resource, taking into account HTML mark-up (for example, words appearing in the title of the page were weighted higher than those in -2 size font) as well as positioning of terms within the document.

A more complex system is proposed in [12]. SVMs are employed to screen out irrelevant resources. Bayesian networks are then used to learn

[1] http://www.cogsci.princeton.edu/~wn/

regular expressions to filter documents for relevant blocks of text. Kwok [9] used SVMs in an automated manner to categorise the Reuters corpus to much success, even with minimal preprocessing.

Most relevant to the work presented in this paper is that of Joachims [13] in which SVMs produce a *ranking* rather than the usual binary positive/negative decision. This new development in SVMs allows the re-ordering to be much more specific; ranking results according to their actual relevancy.

3 Machine learning to complement traditional keyword-based search: The *Tibianna* search engine

Keyword-based search engines rely on good metadata derived from content or supplied by a human indexer. If the right keywords are not known (for example, when searching a topic area that is completely new to the person), results are inferior compared to a search with a perfect combination of keywords. *Tibianna*, a new search engine embedded in MyNewsWave, gives better accuracy to these 'fuzzy' searches by way of (1) reordering existing search engine results, (2) gathered result relevancy user feedback and (3) provision of ontology-based mechanisms provide query refinement functions.

Tibianna is particularly suited for multimedia content, where exact keywords or metadata for the resource is often hard to quantify (as opposed to a document, where the search can be performed on the document itself). Of course, any metadata that is available for a resource should still play a large part in determining search result rankings[2].

In more detail, *Tibianna* uses search session history (i.e, what the user has searched for previously in this session), and ontological data to help the user to refine search. The system works as follows:

- The user starts a web search, the server keeps track of a session.
- Where possible, the server adds lexical refinement options to provide more context for the user. Sources for this data are databases such as WordNet. Ontologies can be used for semantic or lexical disambiguation (e.g. 'Java' which is a drink, an island and a programming language) by allowing the user to select the relevant meaning.
- After viewing a result link, or result summary, the user has the option to rate the 'fitness' of the page, according to their own relevancy function.
- As the user progresses through the search, the SVM learns progressively more about the search intent of the user. After a reasonable number of results have been ranked, Tibianna begins to reorder search results based on what it has learned. The user can also delve deeper into the search by considering the lexical and semantic data that is presented.

4 DigiMood: Classifying web pages based on emotional content

For a prediction of the potential impact of a news item, an assessment of the "emotional" content of an article can be as valuable as a ranking with regard to personal relevance. DigiMood assesses the mood of any web page. An iconic representation is displayed in the browser once the mood has been established.

The learning component predicts the mood of the web page after an initial SVM learning period. The user classifies web pages during this phase, teaching the SVM to match the user's own mood categories. The number of web pages needed for the learning will be determined based on information gathered from a testing phase, and also informed by experimentation conducted in [14].

DigiMood takes the form of a web browser plug-in component. When a web page is loaded, the user selects the mood that best describes the page. During the learning period, the URL, page content (stripped of HTML encoding) and emotional state selected are appended to an XML file. Learning starts once a sufficient number of documents are available for training by SVM^{light} [9]. Once SVM training has completed, DigiMood commences mood classification of pages viewed by the user. The user can adjust the predicted emotion if necessary, thereby providing feedback for further learning periods.

5 Searching for multi-modal background information: The Emily audio mining system.

Topic Detection and Tracking (TDT) refers to computerised techniques for finding topically related material in streams of data of various type (audio, video, text, image etc.). Hence, TDT is multi-modal by definition. Emily includes a method for automatically extracting content from speech so that MyNewsWave approaches TDT functionality. Like the other components of MyNewsWave, the method utilises machine learning.

Emily has two parts: (1) "background" searching on a topic, i.e. the engine will gather related, multi-modal information and (2) topic detection by use of audio data. Standard speech recognition is used to generate a transcript which is then input to a machine learning system that performs topic categorisation.

The learning component of Emily is based on the audio data processed during the input stage of the system. It is assumed that there will be at least a 50% error rate in the transcription from audio to text. The outcome of the learning process is to decide whether a specific piece of audio belongs to a topical category. The newly categorised transcription can then be used

[2] A potential use of Tibianna is the correction of deliberately misleading metadata.

as additional input to the background search. Transcripts are also added to the original input document in an attempt to help with retrieving documents within the correct category.

Background searches take the topic classifications by Emily of audio or web HTML resources to construct search queries. After determining the topic(s) for a resource, Emily queries WordNet for related terms that are then assembled into Google queries. Several of these searches are run, with the union of the results presented to the user in brief form as a side bar.

6 Conclusions

MyNewsWave can support journalists, editors and other knowledge workers by providing a range of web search facilities. MyNewsWave ranks search results according to personal preferences and allows for the classification of multimedia documents into topic categories. Furthermore, web pages can be assessed with regard to the mood they express, and even if the user is away from a machine, a speech user interface allows communication.

Acknowledgements

Special thanks to Harald Schuetz from *Deutsche Welle* (German Foreign Broadcasting Cooperation) who guided this research by providing valuable feedback. Thank you also to Stephen Viller for his comments.

References

[1] J. Tomita and G. Kikui, "Interactive Web search by graphical query refinement," presented at 10th World-Wide Web Conference, Hong Kong, 2001.

[2] E. Glover, S. Lawrence, G. Michael, W. Birmingham, and C. L. Giles, "Web Search - Your Way," *Communications of the ACM*, vol. 44(12), pp. 97-102, 2001.

[3] S. Lawrence, "Context in Web Search," *IEEE Data Engineering Bulletin*, vol. 23(3), pp. 25-32, 2000.

[4] P. D. Bruza, R. McArthur, and S. Dennis, "Interactive Internet Search: Keyword, Directory and Query Reformulation Mechanisms Compared," presented at 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2000.

[5] D. Moldovan and R. Mihalcea, "Using WordNet and Lexical Operators to Improve Internet Searches," *IEEE Internet Computing*, vol. 4(1), pp. 34-43, 2000.

[6] E. Selberg and O. Etzioni, "The MetaCrawler Architecture for Resource Aggregation on the Web," *IEEE Expert*, vol. Jan-Feb, pp. 11-14, 1997.

[7] S. Lawrence and C. L. Giles, "Context and Page Analysis for Improved Web Search," *IEEE Internet Computing*, vol. 2(4), pp. 38-46, 1998.

[8] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20(3), pp. 273-297, 1995.

[9] T. Joachims, *Making Large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning*: MIT-Press, 1999.

[10] T. Joachims, "Text Categorization with Support Vector Machines: Learning With Many Relevant Features," presented at Proceedings of ECML-98, 10th European Conference on Machine Learning, Heidelberg, Germany, 1998, pp. 137-142.

[11] E. Glover, G. Flake, S. Lawrence, W. Birmingham, A. Kruger, C. L. Giles, and D. Pennock, "Improving Category Specific Web Search by Learning Query Modifications," presented at Symposium on Applications and the Internet, SAINT, San Diego, CA, 2001.

[12] A. Kruger, C. L. Giles, F. Coetzee, E. Glover, G. Flake, S. Lawrence, and C. Omlin, "DEADLINER: Building a New Niche Search Engine," presented at Conference on Information and Knowledge Management, Washington, DC, 2000.

[13] T. Joachims, "Optimizing Search Engines Using Clickthrough Data," presented at ACM Conference on Knowledge Discovery and Data Mining, 2002.

[14] C. Heyer and J. Diederich, "Tibianna: A Learning-Based Search Engine with Query Refinement," (to appear) 7th Annual Australasian Document Computing Symposium, Sydney, Australia, 2002.