BROWSER-BASED USER INTEREST MODELLING

Márius ŠAJGALÍK, Michal BARLA, Mária BIELIKOVÁ

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 2, 842 16 Bratislava, Slovakia
{sajgalik,barla,bielik}@fiit.stuba.sk

Abstract. The Web contains a huge amount of information, which grows daily. Also continual progress in web technologies has enabled development of many web applications, which can accommodate most of our information needs. This has resulted in the user spending substantial amount of time on the Web using her web browser. Therefore, we focus on web browser as a platform. In this paper we present our browser-based user modelling framework called Brumo. Since we are limited being on client, we developed an efficient representation of user features and resource-friendly method for key-concept extraction from web pages.

1. Introduction

Information retrieval (IR) on the Web is not an easy task. Unlike IR within a web system, where the domain is more strictly limited, the general Web is very diverse. That is also the reason, why it is called the "wild Web". There are many different domains and even a single user navigates through several of them. But that has both pros and cons. If we cut out the data about user just from a single domain, there is a lot of missing information e.g. about user's general background and we lose many valuable inter-domain links, which can help us disambiguate and explain many acts of user. As we get more data, though noisy, we could sacrifice the lack of some very specific hand-crafted domain knowledge and substitute it with an unsupervised learning of domain and user features that will discover the hidden knowledge automatically.

In our work we focus on analysis of the user browsing the Web. We attempt to discover user interests based on user's activity on the Web, or more specifically inside the web browser, where we have direct access to monitoring entire user's activity on the Web.

2. Brumo architecture

We have developed our own research environment called Brumo[1] [3], which represents a browser-based user modelling and personalisation platform. It is realised as a web browser extension, currently implemented for two major web browsers - Google Chrome and Mozilla Firefox. Brumo itself supports yet additional extensions, which can be created by any Brumo user. It provides a powerful and flexible API, which enables access to browser functionality, database, user model and communication between users (or more exactly, between instances of Brumo extensions).

Brumo comprises multiple modules (Figure 1). The main part is the background script, which links all the modules together. The database has a single interface for all supported web browsers and it creates an abstraction of key-value storage for different implementations. The channelled multicast provides the communication between multiple users, i.e. the instances of Brumo platform. Content processing module specialises in the extraction of user features (e.g. keywords) based on the browsed web content. The extracted features are aggregated and indexed by another module, which manages the whole user model. Brumo API provides the unified access to the core functionality for Brumo extensions across all major web browsers.

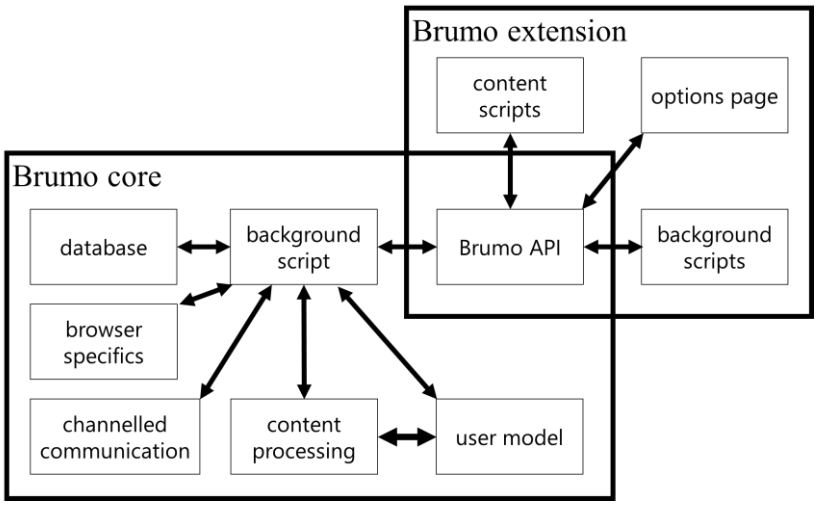

Figure 1. Architecture of Brumo.

3. Efficient representation of user features

For realisation of client-based user models, an efficient representation of features used to compute the user model is needed. In Brumo, an efficient term-based representation [1] is used to express various user characteristics such as interests, knowledge, goals, context of work, etc. For example, user interests are modelled as weighted vector of terms, where each term is linked to some URL address recorded in the user browsing history. There are multiple terms for each URL and similarly each term can be linked to multiple URLs. These links

[1] Brumo – `http://brumo.fiit.stuba.sk`

connecting terms with URLs are also weighted according to their mutual relevance. They denote the relevance of a web page at some URL to given term. Since the term represents a user interest, we can find out how interesting particular web page is by following the corresponding link. It is important to note the variability of terms that can stand for not just words extracted from read articles, but possibly other units like stems, lemmas or concepts.

Brumo provides a powerful indexer. It was designed to achieve the best possible time and memory complexity for all needed operations to be ready for real-world user modelling and personalisation in a web browser environment. It contains two basic data structures - user interest tree and domain interest tree, which are rather flexible in its application and can be utilised in various personalisation scenarios. User interest tree enables us efficiently e.g. to retrieve the most characteristic features like user interests or retrieve the most interesting web pages from user web browsing history. Domain interest tree broadens these possibilities by supporting basically the same operations as the user interest tree, but efficiently filtering the results by some particular domain.

4. Keyword-based user model in Brumo

Currently, in our Brumo platform, we use keyword extraction from browsed web pages to infer the interests of a user. To extract keywords from a web page, we combine multiple methods. First, we consider the text content of a web page. We extract the article using Readability[2] and utilise the web browser's built-in functionality[3] to obtain raw text. In further preprocessing we tokenise the text into words using jspos[4] lexer, filter out stop-words and all words shorter than 3 characters and consider further only nouns as tagged by jspos POS tagger. With these feasible words extracted, we compute relevance of each word as an average of normalised TF-IDF [7] and TextRank [4]. We normalise the TF-IDF value by the text length. Afterwards, we look at keywords metatag in HTML structure and propagate these keywords by doubling their relevance value. The IDF values are obtained from Google N-gram corpus[5].

5. Key-concept extraction

Recently, we have developed a new method of key-concept extraction [2]. It differs from keyword extraction in using concepts instead of words, which are represented as WordNet [5] concepts (synsets). The advantage of using concepts over simple words is that concepts, apart from words, are unambiguous and we know the exact meaning of each concept.

The core idea of our approach is to combine the PageRank-based word sense disambiguation [6] with idea of another PageRank-based method, TextRank [4]. The PageRank-based word sense disambiguation method presented in [6] takes a subgraph of all WordNet concepts that are reachable from the concepts containing some words present in text and

---

[2] Readability – `https://code.google.com/p/arc90labs-readability/`

[3] textContent property – `http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/core.html\#Node3-textContent`

[4] JavaScript part-of-speech tagger – `http://code.google.com/p/jspos/`

[5] Google N-gram corpus – `http://storage.googleapis.com/books/ngrams/books/datasetsv2.html`

runs PageRank, inferring the probability of each concept being mentioned in text. TextRank utilises word collocations to construct word graph over text with collocated words linked together and runs PageRank to infer the relevance of each word to the text. Besides these two methods, we use yet additional notion of information content of concepts, i.e. we weight the concept relevance by information content, which is analogical to using inverse document frequency in keyword extraction.

6. Conclusions

We have developed Brumo platform with modular and extendible architecture, which provides suitable environment for realisation of multiple personalisation scenarios. In our further research, we plan to develop novel methods of feature extraction that could be used to model the user interests. We will focus on processing the browsed web content enriched with the client-based metadata about the content and user's interaction with it. We would like to further explore the utilisation of distributed representation of words, which we consider very promising for use in keyword extraction and combine it with other designed features into a dynamical model of user interests, which would support also the temporal nature of user's interests.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11 (2011-2014) and the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

to other papers publishing the results that are summarized here

[1] Šajgalík, M., Barla, M., Bieliková, M.: Efficient Representation of the Lifelong Web Browsing User Characteristics. In: *Proceedings of the Lifelong User Modelling Workshop at UMAP 2013 User Modeling, Adaptation, and Personalization*, CEUR, (2013), pp. 21-30.

[2] Šajgalík, M., Barla, M., Bieliková, M.: From ambiguous words to key-concept extraction. In: *Proceedings of 10th International Workshop on Text-based Information Retrieval at DEXA 2013*, IEEE, (2013), pp. 63-67.

[3] Šajgalík, M., Barla, M., Bieliková, M.: BrUMo – personalisation platform in web browser (in Slovak). In: *DATAKON 2013: Proceedings of the Annual Database Conference*, (2013).

Other references

[4] Mihalcea, R., Tarau, P.: TextRank: Bringing Order into Texts. In: *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, ACL, (2004), pp. 404-411.

[5] Miller, George A.: WordNet: a lexical database for English. *Communications of the ACM*, (1995), vol. 38, no. 11, pp. 39-41.

[6] Ramakrishnanan, G., Bhattacharyya, P.: Text Representation with WordNet Synsets using Soft Sense Disambiguation. *Ingenierie des Systems d'Information*, (2003), vol. 8, no. 3, pp. 55-70.

[7] Salton, G. and Buckley, C.: Term-weighting approaches in automatic text retrieval. *Information Processing and Management: an International Journal*, (1988), vol. 24, no. 5, pp. 513-523.