Context-based Improvement of Search Results in Programming Domain

Jakub KŘÍŽ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
jacob.kriz@gmail.com

When programming the programmer encounters many difficulties he cannot instantly overcome. These difficulties might be rather trivial, like errors in compilation or more complex, like tasks he cannot solve or tasks he does not remember the solution of. When solving one of these problems the programmer usually uses a web search engine to look for help. However, the search results might not be effective – the user usually enters a query consisting of a few words only. This query often does not describe the matter properly and so the search results might end up being inaccurate.

The programmer does many other things along with writing the actual source code. He opens applications and other source codes, searches them, writes notes and, last but not least, he searches the internet for solutions of problems or errors he might have encountered. He does all these tasks in a context or in order to do something, usually to solve a problem. Via identification of this context we can understand the meaning behind the programmer's actions. This understanding can be used to make the web search engine to be context-aware – to make web search results more accurate and relevant to the current task. In this work we analyse the existing methods used to mine the context in other domains and analyse their usability in the programming domain.

Context-based recommender systems or search engines are quite common on the web nowadays. When working with the web search engines, the context is often understood to be the search history of a particular user in the current session and the visited documents from the searches performed. The data used to create the context model with is gathered from the metadata from visited documents with various approaches. For example, White et al. [3] use document categorization whereas Kramár et al. [1], among other methods, use document content analysis, namely keyword extraction.

The most important source of contextual information in programming domain seems to be the source code the programmer is currently working on. The first goal of this work is to build a contextual model based on the metadata extracted from the

* Supervisor: Tomáš Kramár, Institute of Informatics and Software Engineering

Spring 2013 PeWe Workshop, April 5, 2013, pp. 15–16.

source code. Metadata extraction from source code files is an area which has not been thoroughly researched in other works. Therefore we intend to use approaches used for metadata extraction from general documents and modify them, like standard methods for keyword extraction.

Using general statistical keyword extraction methods like tf-idf to extract keywords from source codes has been evaluated before and showed promising results. A modification of the algorithm was proposed and evaluated in. Ohba et al. [2]. The authors modify the standard tf-idf algorithm to extract what is called conceptual keywords. This method is designed to help programmers with reading and understanding source codes previously unknown to them. Its goal is to mine the keywords, which express helpful, key concepts for understanding of the algorithm. The method proved quite effective and it should, after some adjustments, also prove effective in order to build the conceptual model.

To help the programmer with errors and specific tasks we also need to find out what technologies he currently works with. Therefore we intend to modify the method to also extract technical keywords. A very important part of keyword extraction via tf-idf is choosing the right corpus. When it comes to source code files, multiple choices exist – we can use all files written in the same language, all files in the same project, or, when extracting keywords from a fragment of the source code, only the current file. We intend to evaluate multiple approaches in order to determine the best approach to mining metadata from source code files.

We can combine this method with described methods generally used to create context models, like the analysis of visited websites and with other, probably much less significant sources of context, like the keywords extracted from notes.

To actually improve the search results we can use standard methods, like search query expansion, which is often used with great success [1]. The designed methods will be experimentally evaluated on a large dataset of logs made from programmers' activities.

Acknowledgement. This contribution is the partial result of the Research & Development Operational Programme for the project Research of methods for acquisition, analysis and personalized conveying of information and knowledge, ITMS 26240220039, co-funded by the ERDF.

References

[1] T. Kramár, M. Barla, M. Bieliková. Personalizing Search Using Metadata Based, Socially Enhanced Interest Model Built from the Stream of User's Activity. In *Journal of Web Engineering*. Vol. 12, No. 1&2, pages 65-92. 2013.

[2] M. Ohba, K. Gondow. Toward mining "concept keywords" from identifiers in large software projects. In *Proceedings of the 2005 international workshop on Mining software repositories*, MSR '05, 2005.

[3] R. W. White, P. Bailey, and L. Chen. Predicting user interests from contextual information. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information re-trieval*, SIGIR '09, pages 363-370, New York, NY, USA, 2009. ACM.