Discovery of Technical Expertise from Open Source Code Repositories

Rahul Venkataramani[1], Atul Gupta[2]
[1]International Institute of Information Technology
[2]Indian Institute of Information Technology, Design and Manufacturing, India
rahul.venkataramani@iiitb.org, atul@iiitdmj.ac.in

Allahbaksh Asadullah, Basavaraju Muddu, Vasudev Bhat
Infosys Labs, India
{allahbaksh_asadullah, mbraju, vasudev_d}@infosys.com

ABSTRACT

Online Question and Answer websites for developers have emerged as the main forums for interaction during the software development process. The veracity of an answer in such websites is typically verified by the number of 'upvotes' that the answer garners from peer programmers using the same forum. Although this mechanism has proved to be extremely successful in rating the usefulness of the answers, it does not lend itself very elegantly to model the expertise of a user in a particular domain. In this paper, we propose a model to rank the expertise of the developers in a target domain by mining their activity in different opensource projects. To demonstrate the validity of the model, we built a recommendation system for StackOverflow which uses the data mined from GitHub.

Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/Machine Systems—*Human Information Processing*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Information Networks*

General Terms

Experimentation

Keywords

github, stackoverflow, knowledge discovery, technical expertise, source code repository, recommendations.

1. INTRODUCTION

During the software development process, developers post questions and answer doubts in various Question and Answer(Q&A) websites. The 'usefulness' of an answer is determined by the number of "upvotes" the answer receives from peer programmers. In such a model, the reputation of the user answering the question is ignored. Hence, it is difficult for an amateur developer to ascertain if the user answering the question has any experience of working on a project related to the domain of the question. Another drawback in most of the present day Q&A websites is the unavailability of a recommendation system to help a user posting a question identify the potential developers who can answer that question. Also, a user willing to answer a question has the additional task of finding questions which match his competence[1].

One of the ways of quantifying the expertise of a developer in a domain is by measuring his contributions in the said domain. Sometimes, the dataset in a domain is insufficient to make accurate recommendations. Winoto et al.[3] provide insights into how such a dataset can be augmented with an auxiliary dataset to overcome the information deficit. One of the key observations they make is that the two datasets must be semantically close to each other to make better recommendations. For e.g., movies and music are closer to each other than movies and textbooks. During the software development process, developers use GitHub to host and share code among peer programmers. They also use collaboratively edited Q&A websites like StackOverflow to discuss the problems that arise during this process. Since the content in these websites are semantically close to each other, they make a good case for cross domain recommendations. Dabbish et al.[2] have pointed out that humans, while browsing a large source code repository, make a number of non trivial semantic inferences about the quality of commit, the proficiency of the author etc. In this paper, we propose a model to capture the technical expertise of developers by mining their activity from the open source code repositories and its application in a target domain. We validated this model by building a recommendation system for online Q&A websites like StackOverflow by mining user expertise from GitHub.

2. DEFINITIONS

- **Technical Terms:** The various programming language constructs like method names, class names etc. constitute the technical terms. These terms T are mined from the source code repositories.
- **Tags:** In Q&A websites, users annotate the questions with words that describe the question. These are referred to as "tags". Let *tags* represent the set of tags. e.g. javaio, IOException could be the tags for a question pertaining to a problem in Java I/O.

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

3. PROPOSED MODEL

The proposed model demonstrates the capture of expertise of a developer in a domain by mining open source code repositories and its application in a target domain like providing recommendations in an online Q&A website. The model is constructed in three stages as described in the following sections:

3.1 Capturing the Expertise of a Developer

The familiarity of a developer with a technical term is measured by his frequency of use of the term. The familiarity of a set of developers D with terms T can be represented by the relation $\gamma : D \times T \rightarrow n$, where increasing values of n denote better familiarity of the developer with the particular term. A value of $n = 0$ denotes that the developer has never used the term. In our model, a project P is modelled as $P = (F, D, \alpha)$, where F represents the set of source code files and $\alpha : F \times D \rightarrow \{0, 1\}$. A value of $\alpha(F_i, D_i) = 1$ denotes the file F_i is modified by a developer D_i. From the set of files F, we extract a set of technical terms used in each file. The set of terms and their frequency of occurrence is denoted by $\beta : (F \times T) \rightarrow n$ where n denotes the frequency of occurrence of a term Ti occurring in file Fi. Using the relations α and β, we arrive at the value for γ.

3.2 Mapping Technical Terms to Tags

The output from the first stage is a relation between a developer and a term. However, the tags used in the Q&A websites do not capture information at this level of granularity. Hence, there needs to be a mapping between the terms obtained from mining source code repositories and the technical tags used in the target domain. The mapping function M is defined as: $M : T \times tags \rightarrow n$. M is a one to many non-injective function i.e. every technical term can be a part of multiple tags.

3.3 Mapping Developer Expertise in the Target Domain

Using the relations γ and M, we model the expertise of a developer in the target domain by the relation $\delta : Tags \times D \rightarrow n$, where n denotes the relative expertise of a developer in a topic denoted by the given tag. Thus, we have modelled the expertise of a developer in a target domain by using a vocabulary of $tags$.

4. MODEL EVALUATION & FUTURE WORK

To test the validity of the model, we built a recommendation system for StackOverflow. In particular, the two types of recommendations we make are:

- **Developer Recommendation:** *Given a question q, recommend one or more developers who possess the necessary expertise to answer the question.*
- **Question Recommendation:** *Given a developer d, recommend one or more questions for which he/she possesses the necessary expertise to answer.*

For building the prototype, we considered 5000 questions of Java domain from StackOverflow and 20 Java projects from GitHub. This data enabled us to build a database

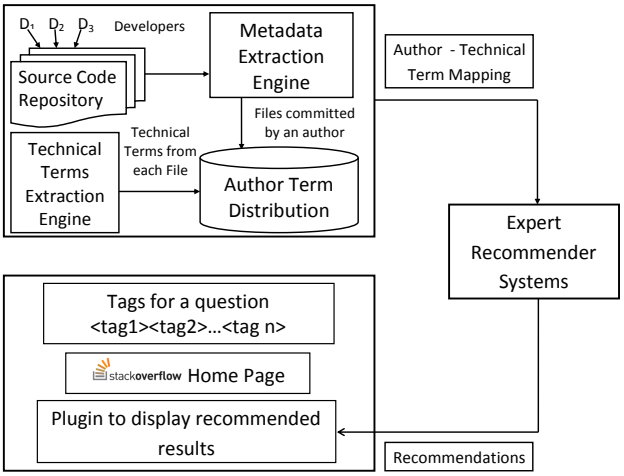

Figure 1: Block Diagram of Proposed Model

which maps the proficiency of every author w.r.t all the tags which co-occur with tag *Java* on StackOverflow using the proposed model. For lack of an API which allows us access of common users in GitHub and StackOverflow, we manually checked the tags of the questions to which the authors in GitHub, that we mined, have answered. For a sample of 15 authors, we found that around 7 authors answered questions with tags which our model has discovered he/she is proficient in. The rest of the authors answered more number of questions pertaining to their area of interest rather than the programming language concepts that they were proficient in.

In the current work, the expertise of a developer is modelled only on the basis of his proficiency in programming language concepts and not necessarily in his area of interest. In the future, we plan to extend the current model to reflect the user's area of interest along with his programming language expertise. This work is only a prototype in the larger vision of harnessing information from different information networks and leveraging the same to derive richer semantics. These semantics lend themselves to a number of applications like skilled recruitment, recommendation systems for Q&A websites, finding related content, etc.

5. REFERENCES

[1] D. Cosley, D. Frankowski, L. Terveen, and J. Riedl. Suggestbot: using intelligent task routing to help people find work in wikipedia. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 32–41. ACM, 2007.

[2] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. Social coding in github: transparency and collaboration in an open software repository. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 1277–1286. ACM, 2012.

[3] P. Winoto and T. Tang. If you like the devil wears prada the book, will you also enjoy the devil wears prada the movie? a study of cross-domain recommendations. *New Generation Computing*, 26(3):209–225, 2008.