A Survey and Taxonomy of Intent-Based Code Search

Shailesh Kumar Shivakumar, IGNOU, India

ABSTRACT

In this paper, the authors introduce the novel concept of intent-based code search that categorizes code search goals into a hierarchy. They will explore state-of-the-art techniques in source code search covering various tools, techniques, and algorithms related to source code search. They will survey the code search field through the core use cases of code search such as code reusability, code understanding, and code repair. They propose a user intent-based taxonomy based on the code search goals. The code search goal taxonomy is derived based on deep analysis of literature survey of code search, and the taxonomy is validated based on their exclusive developer survey conducted as part of this paper. The code search goal taxonomy is based on logical categorization of code search goals and shared characteristics (query type, expected response, and such) for each of the categories in the taxonomy. The paper also details the latest trends and surveys the code search tools and the implications on tool design.

KEYWORDS

Code Search, Software Search

1. INTRODUCTION TO SOURCE CODE SEARCH

Source Code search primarily involves searching internal or external code base to match the search query input. Software developers search for relevant code during various phases of software development lifecycle and while implementing a code change (Sadowski et al., 2015) or during the start of new functionality development. Source code search mainly involves search, evaluation, retrieval and application of the source code from various sources to solve a development problem (Hummel & Atkinson, 2006). 52% of people use general purpose search engines to find reusable software (Hucka & Graham, 2016). Other sources of information are asking colleagues (45%), literature survey (34%), social sites (25%), public software repository (21%) and mailing list (12%) (Hucka & Graham, 2016).

Developers like to look at existing working code sample and use it as reference for their development needs. With vast amount of code available on web, we could use web as a source code repository (Hummel & Atkinson, 2006). Though code reusability is the primary purpose of code search, developers also use search tool for other things such as code understanding, code repair (Ke et al., 2015), impact analysis and such. We are going to look at the main motivations for code search in coming sections.

Availability of Free and open source software (FOSS) has further increased the scope and effectiveness of code search (Rao, 2013) and helps in code implementation (Gallardo-Valencia, 2013).The emergence of Social media platforms, Web 2.0 technologies has given rise to a new set of code sharing platforms such as StackOverflow (Ponzanelli et al., 2014), YouTube, Yahoo Answers, Facebook questions, Quora and such (Barzilay et al., 2013). These platforms mainly harness the collective intelligence of the crowd through users' active participation and contribution. Public code repositories such as GitHub, BitBucket, SourceForge and such also provide rich source of reusable code. Modern code search engines leverage the crowd sourced code sources to match the relevant search results.

1.1 Contribution of This Paper

Given below are the high level contributions of this paper:

1. The paper does extensive literature survey of code search goals and introduces the novel concept of "intent-based code search" and defines the taxonomy based on searcher's intent/search goals. Intent based code search identifies the primary search goal (code reuse, code understanding or code repair) and customizes the search process (indexing, matching, result display) based on the identified search goal. For each of the goals we have identified the sub goals, search methods, query types, and query matching and techniques available in state of the art tools.
2. Analysis of challenges and gaps with the state of the art in each of the code search goals.
3. We have conducted a developer survey about code search to re-validate the code search goal taxonomy categories and the paper discusses the high level findings from the developer survey.
4. We have analyzed and identified the implications of code search goals on search tool designers and have recommended the key features needed based on the analysis.
5. The paper provides comprehensive survey of key code search tools along with their details such as matching techniques, ranking algorithms used, UI, data source and such.
6. The paper elaborates the key metrics that are used for evaluating the quality of code search results.
7. Compilation of trends and potential research topics in the area of code search.

Note: We have used "code search goals" and "user intent" synonymously in this paper.

2. RELATED WORK AND MOTIVATIONS

On the web search front, researchers have developed the taxonomy based on goals and queries. Broder (2002) developed a taxonomy of generic web search by mining the log of a web search engine (Broder, 2002). The taxonomy defined high level categories that reflected users' intent. Broadly the search queries were categorized as navigational queries (to satisfy the users seeking a specific information), informational query (to satisfy the users who want to understand more about a topic) and transactional query (to facilitate the users' interaction with the web site). Rose and Levinson (2004) (Rose & Levinson, 2004) built upon Broder's work to further elaborate the categories and built a taxonomy of search goals and introduced various sub-categories for each of the three broad categories. Mili et al. (Mili et al., 1998) surveys the method to store and retrieve the software libraries for reuse. The authors characterize software libraries into categories such as Information retrieval methods (based on information stored in the software components), descriptive methods (based on relevance of the keywords), Operational semantics (based on differentiating features of the software), Denotional semantics (based on denotional semantics of the software), Topological methods (based on feature matching) and Structural methods (based on their structure).

In the code search space, several surveys (Gallardo-Valencia & Sim, 2011) (Sim et al., 1998) (Russo & Umarji, 2008) (Sadowski et al., 2015) (Hucka & Graham, 2016) aim to understand the

Related Content

A Comparative Analysis of Reliability Assessment Methods for Web-Based Software
Jinhee Park, Yeong-Seok Seoand Jongmoon Baik (2013). *International Journal of Software Innovation (pp. 31-44).*
www.irma-international.org/article/a-comparative-analysis-of-reliability-assessment-methods-for-web-based-software/105630

Weaving Security into DevOps Practices in Highly Regulated Environments
Jose Andre Morales, Hasan Yasarand Aaron Volkmann (2018). *International Journal of Systems and Software Security and Protection (pp. 18-46).*
www.irma-international.org/article/weaving-security-into-devops-practices-in-highly-regulated-environments/221157

Automatic Test Sequence Generation and Functional Coverage Measurement From UML Sequence Diagrams
Nazm Umut Ekiciand Tugkan Tuglular (2023). *International Journal of Information System Modeling and Design (pp. 1-21).*
www.irma-international.org/article/automatic-test-sequence-generation-and-functional-coverage-measurement-from-uml-sequence-diagrams/332865

Dynamic Management of Resources in Cloud Computing
Pradeep Kumar Tiwariand Sandeep Joshi (2020). *International Journal of Software Innovation (pp. 65-81).*
www.irma-international.org/article/dynamic-management-of-resources-in-cloud-computing/243380

Applications of Social Media in Academic Library Services: A Case of the Hong Kong Polytechnic University Library
Elaine Wei San Kong, Dickson K. W. Chiuand Kevin K.W. Ho (2016). *International Journal of Systems and Service-Oriented Engineering (pp. 53-65).*
www.irma-international.org/article/applications-of-social-media-in-academic-library-services/153681