An Empirical Study on Neophytes of Stack Overflow: How Welcoming the Community is towards Them

Abdullah Al Jobair, Suzad Mohammad, Zahin Raidah Maisha, Md. Jubair Ibna Mostafa and Md. Nazmul Haque

Department of Computer Science & Engineering,
Islamic University of Technology, Boardbazar, Gazipur, Bangladesh

Keywords: Neophyte, Stack Overflow (SO), New User, Hostile Environment.

Abstract: Stack Overflow (SO) is the most popular question and answers (Q&A) platform for programmers with a rapidly expanding community of new users. However, the unwelcoming environment towards new users has been under discussion for several years, which is a major concern towards the enhancement of a skillful community. In this work, we study a specific group of users who are either registered in the last 45 days or have a reputation less than or equal to 50 and term them as *"neophytes"*. We investigate whether neophytes actually face hurdles while collaborating in Stack Overflow and, if so, identify the reasons behind this phenomenon by qualitative and quantitative analysis. Our study finds that neophytes are indeed facing hurdles while collaborating in the platform. The reasons behind the hurdles include harsh moderation of posts, negligence of the posts, deleting or closing of posts, downvoting without providing any proper reasoning, etc. Our findings can provide guidelines to create a more user-friendly Stack Overflow community. Furthermore, this study can guide researchers to observe the reactions of neophytes in adverse situations and recommend some steps for the community to make positive changes to the Stack Overflow environment.

1 INTRODUCTION

The exponential growth of the software development industry leads to forming a community for aiding one another with wisdom and experience. Q&A Platforms are the result of such a need which eventually establishes a community for sharing knowledge. In the community, users share skills and techniques among themselves to solve different problems. Among all the online software development Q&A platforms, Stack Overflow is the largest and the most renowned one (May et al., 2019). From the dawn of its origin, a total of 16.5 million users have registered on the site with an average of 3,370 new users registering every day and making around 11,203 posts on a daily basis[1] (based on a query run in August 2021). Today's massive repository of 21 million questions and 31 million answers in Stack Overflow (Moutidis and Williams, 2021) is the result of the gradual progress of the community since 2008.

The accessibility of this extensive dataset has brought about a number of researches on this platform[2] including evolution of community, posts, code snippets, along with user behavior, user participation, mining SO and associated technologies and many more (Ahmed and Srivastava, 2017; Adaji and Vassileva, 2016). However, a limited number of studies are focused on the environment of the community for new users.

With the community's swift expansion (Mamykina et al., 2011), Stack Overflow's environment draws significant attention. Any hostile nature of the community may turn off the eagerness for participation, which hinders the lively ambience of the platform. However, related studies intimate the existence of unwelcoming environments specially to the new users. Less experienced users become frustrated due to the obscurity in closing questions (Tóth et al., 2020). It ultimately leads the community to become hostile and unsupportive, mostly to the new users. The study of Abbas presented unanswered questions, negative feedbacks and deleted questions as the root of a massive discouraging impact towards users (Abbas, 2019). According to (Slag et al., 2015), 47% of users

[1] https://data.stackexchange.com/stackoverflow/query/1541382

[2] https://stackoverflow.blog/2009/06/04/stack-overflow-creative-commons-data-dump/

Al Jobair, A., Mohammad, S., Maisha, Z., Mostafa, M. and Haque, M.
An Empirical Study on Neophytes of Stack Overflow: How Welcoming the Community is towards Them.
DOI: 10.5220/0011081100003176
In *Proceedings of the 17th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2022)*, pages 197-208
ISBN: 978-989-758-568-5; ISSN: 2184-4895

An Ensemble Approach to Detect Code Comment Inconsistencies using Topic Modeling

Fazle Rabbi, Md. Nazmul Haque, Md. Eusha Kadir, Md. Saeed Siddik and Ahmedul Kabir
Institute of Information Technology
University of Dhaka, Bangladesh
Email: {bsse0725, bsse0635, bsse0708, saeed.siddik, and kabir}@iit.du.ac.bd

***Abstract*—In modern era, the size of software is increasing, as a result a large number of software developers are assigned into software projects. To have a better understanding about source codes these developers are highly dependent on code comments. However, comments and source codes are often inconsistent in a software project because keeping comments up-to-date is often neglected. Since these comments are written in natural language and consist of context related topics from source codes, manual inspection is needed to ensure the quality of the comment associated with the corresponding code. Existing approaches consider entire texts as feature, which fail to capture dominant topics to build the bridge between comments and its corresponding code. In this paper, an effective approach has been proposed to automatically extract dominant topics as well as to identify the consistency between a code snippet and its corresponding comment. This approach is evaluated with a benchmark dataset containing 2.8K Java code-comment pairs, which showed that proposed approach has achieved better performance with respect to the several evaluation metrics than the existing state-of-the-art Support Vector Machine on vector space model.**

***Index Terms*—Source Code, Code Comment, Topic Modeling, Software Artifact Analysis**

I. Introduction

Code comments with its corresponding source code are the main artifact of any software systems. For the management of software evolution and maintenance, developers provide comments with a code fragment which give insightful information about a software system. Comments are very important as they are more natural, descriptive and easy to understand than source code [1], [2]. In large projects, new developers are highly dependent on code comments to understand its corresponding source codes. Researchers found that code and comments evolve over time [3] and this evolved codes and comments become inconsistent to each other. Because of changing codes frequently and keeping corresponding comments same, comments become invalid or inconsistent with corresponding source code.

Tracking the inconsistency of source code and its comment, several diverse approaches have been proposed. Where most of the approaches apply Information Retrieval (IR) techniques to collect lexical information with the assumption that the textual information of source code and comment are same. However, that assumption can be violated [4] in several cases, for example, the vocabulary developers use to write source code can be different from the vocabulary of comment (e.g. synonym). Nevertheless, there is no sufficiently rich literature to track this inconsistency because of lacking standard datasets. A benchmark dataset has been provided [5] with a proposal to measure the coherence between source code and comment. Lexical similarity has been collected by using Vector Space Model to classify the text using tf-idf [6] and finally the code-comment inconsistency is measured using Support Vector Machine (SVM). However, this approach uses all of the vocabulary as features which can take a huge execution time.

By analyzing existing literature, some insights of source code and comments have been found, which are concluded below as the research direction in this domain.

- A single word (topic) is more important than a large number of similar words (features). For example, if a bag of words is found from a java method like, “dropdown”, “chrome”, “menu”, “http” or “browser”, a topic related to “browser” can represent these words.
- The size of comments is less than the size of source code. So, the source code and comment need to be represented into a fixed-sized common topic.
- Synonymous words have been chosen by developers while writing comment with respect to source code. So, to capture the semantic information between source code and comment, the vocabulary information needs to be incorporated.

To capture these insightful information, several Research Questions (RQ) have been raised to propose an efficient inconsistency detection approach, which are listed below.

- **RQ1:** How to comprehend the insight meaning of a code and comment pair?
- **RQ2:** How to measure the relation between the code and comment pair?

We focused on the above research questions as our objectives and tried to answer them throughout the newly proposed code comment inconsistency detection technique. This paper proposes an automated approach to identify the inconsistency of source code with its respective comments. The breakdown of the contributions of this paper are listed as follows.

- Datasets are pre-processed to capture more meaningful information about source code and comments, e.g., de-

DOI reference number: 10.18293/SEKE2020-062.

Nurse Care Activity Recognition: A GRU-based Approach with Attention Mechanism

Md. Nazmul Haque
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
bsse0635@iit.du.ac.bd

Mahir Mahbub
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
bsse0807@iit.du.ac.bd

Md. Hasan Tarek
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
bsse0818@iit.du.ac.bd

Lutfun Nahar Lota
Department of Computer Science & Engineering
East West University
Dhaka, Bangladesh
lota.nahar@ewubd.edu

Amin Ahsan Ali
Department of Computer Science & Engineering
Independent University, Bangladesh
Dhaka, Bangladesh
aminali@iub.edu.bd

ABSTRACT

Human activity recognition is a challenging task due to complexity and variations of human movements while performing activities by different subjects. Extracting features to model the temporal evolution of different movements plays an important role in this task. In this paper, we present the approach followed by our team, Dark_Shadow, to recognize complex nurse activities in the "Nurse Care Activity Recognition Challenge" [1]. We present a deep learning method to capture the movements of essential body parts from time series of human activity data collected by sensors and then classify them. Deep learning approaches have provided satisfactory results in various human activity recognition tasks. In this work, we propose a Gated Recurrent Unit (GRU) model with attention mechanism to recognize the nurse activities. We obtain approximately 66.43% accuracy for person-wise one leave out cross validation.

KEYWORDS

Activity recognition, Nurse care activity, GRU, Attention mechanism

ACM Reference Format:
Md. Nazmul Haque, Mahir Mahbub, Md. Hasan Tarek, Lutfun Nahar Lota, and Amin Ahsan Ali. 2019. Nurse Care Activity Recognition: A GRU-based Approach with Attention Mechanism. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2019 International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct), September 9–13, 2019, London, United Kingdom.* , 5 pages. https://doi.org/10.1145/3341162.3344848

UbiComp/ISWC '19 Adjunct, September 9–13, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6869-8/19/09…$15.00
https://doi.org/10.1145/3341162.3344848

1 INTRODUCTION

Human activity recognition mainly focuses on recognizing different human activities by using data collected from videos or sensors. There are diverse applications of activity recognition in areas such as robotics [17], video analysis [9], gaming, animation, surveillance, [12], and human computer interaction [14]. However, accurate recognition of the human actions from sensor data is challenging because of the complexity of activities, between and within subject variations in performing the activities, and noisy sensor data [10].

In this paper, we focus on nurse care activity recognition for "Nurse Care Activity Challenge" [1]. In this challenge six activities are to be recognized from different sensor data. Inoue et. al prepared a similar dataset and performed Bayesian estimation by marginalizing the conditional probability of estimating the activities for a segment sample attaining an accuracy of only 73.18%. This nurse-care activity recognition task is challenging because the nature of performing the same activities by different nurses vary and some activities contain sub-activities that are similar to other activities [8].

Most existing methods for human activity recognition analyze 3D depth data by constructing mid-level part representations, or using trajectory descriptors of spatial-temporal interest points. Research efforts have also been devoted to motion data analysis, motion detection and recognition, which is widely known as human motion evaluation [2, 4–6]. With the advent of low cost, non-intrusive depth sensors such as Microsoft Kinect [14], research efforts are now devoted to the utilization of 3D skeleton joint positions. Because, the features related to the body part movement extracted from Kinect sensor are more discriminatory as they represent actions properly.

Hossein et. al. [16] use sequences of joint angles and relative positions of joints as features. Among these features they selected most informative sequences of joint angles using entropy of the joint angles. They also determine the most informative relative motions, considering inter/intra-action variations for the differences in 3D positions of each joint pair. After that they calculate Longest Common Sub Sequence within the feature vectors to find the the similarities among different activities. Patrona et. al. [15] propose a framework for online action detection and recognition based on an

1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT 2019)

GRU-based Attention Mechanism for Human Activity Recognition

Md. Nazmul Haque
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
bsse0635@iit.du.ac.bd

M. Tanjid Hasan Tonmoy
Dept. of CSE
University of Dhaka
Dhaka, Bangladesh
2015-116-770@student.cse.du.ac.bd

Saif Mahmud
Dept. of CSE
University of Dhaka
Dhaka, Bangladesh
2015-116-815@student.cse.du.ac.bd

Amin Ahsan Ali
Dept. of CSE
Independent University of Bangladesh
Dhaka, Bangladesh
aminali@iub.edu.bd

Muhammad Asif Hossain Khan
Dept. of CSE
University of Dhaka
Dhaka, Bangladesh
asif@du.ac.bd

Mohammad Shoyaib
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
shoyaib@du.ac.bd

***Abstract*—Sensor data based Human Activity Recognition (HAR) has gained interest due to its application in practical field. With increasing number of approaches incorporating feature learning of sequential time-series sensor data, in particular the deep learning based ones has performed reasonably in uniform labeled data distribution scenario. However, most of these methods do not capture properly the temporal context of time-steps in sequential time-series data. Moreover, the situation becomes worse for imbalanced class distribution which is a usual case for HAR using body-worn sensor devices. To solve this issues, we have integrated hierarchical attention mechanism with recurrent units of neural network in order to obtain temporal context within the time-steps of data sequence. The introduced model in this paper has achieved better performance with respect to the well-defined evaluation metrics in both uniform and imbalanced class distribution than the existing state-of-the-art deep learning based model.**

***Index Terms*—Human Activity Recognition, Attention Mechanism, Gated Recurrent Unit**

I. Introduction

Human Activity Recognition(HAR) is a domain of research aimed at recognizing human actions and movements from a series of observations. The increasing public adoption of smart devices with sensors such as accelerometer and gyroscope has created the opportunity to organize considerable amount of sensor data for classification of human activity. The research activities focused on HAR incorporates the compilation of sensor readings into sequential time-series data and develops models for recognition of activities by analyzing the acquired sequential sensor readings. HAR poses a variety of promising application domain which includes physical activity annotation in the field of medical data analysis [1], personal assistant system [2], augmented and virtual reality [3] and many others.

In the past years, the state-of-the-art solutions to HAR mainly based on the traditional machine learning techniques. This techniques mainly depend on heuristic based hand-crafted feature engineering that rely on low level representations. As the traditional machine learning models use low level representations, it lacks the characteristics of generalization [4], [5]. High level abstraction along with low level representations are necessary for a likely solution to this generalization problem.

Deep learning based methods deal with both low and high level representations of data. Therefore, recently two variants of deep learning methods namely convolutional [6] and recurrent [7] neural network models are become dominant over traditional methods in terms of performance. For example, to identify HAR, convolutional neural network(CNN) is used in [8] and [9] and recurrent neural network(RNN) is used in [7] and [10].

Although deep learning based methods show promising performance, conventional sliding window based approach for CNN is unable to fully capture the temporal context of the sensor reading [11] which is required for better classification of activities. For sequence data, RNN performs better than CNN in most cases as it captures sequence information [12]. However, RNN faces long term dependency problems [13] when the sequence is long enough. Note that, the sequence information found in HAR data is usually long. So, it is necessary to capture the long term dependency information for better classification.

Gated Recurrent Unit (GRU) is a variant of RNN which incorporate long term dependency information [14]. It is expected that GRU will perform better in case of HAR data as it is able to capture temporal context of sensor data. It is noteworthy to mention here that, all temporal context are not equally important for classification, some are more important than others. Hence, it is necessary to give more attention to the important temporal context than others. Moreover, during the

978-1-7281-3445-1/19/$31.00

2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and
2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)

Bug Severity Classification Based on Class-Membership Information

Sayeda Shamma Alia
and Md. Nazmul Haque
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
bit0427@iit.du.ac.bd
bsse0635@iit.du.ac.bd

Sadia Sharmin
Dept. of Computer Science and Engineering
East West University
Dhaka, Bangladesh
sadia.sharmin@ewubd.edu

Shah Mostafa Khaled
and Mohammad Shoyaib
Institute of Information Technology
University of Dhaka
Dhaka, Bangladesh
khaled@du.ac.bd
shoyaib@du.ac.bd

***Abstract*—Now-a-days bug classification along with its severity prediction has become an important issue for better maintenance and cost reduction of software. Different approaches have already been introduced in this regard including topic modeling, concept profile etc., however, most of them require a lot of parameter tuning, different types of computations for identifying bug severity. Furthermore, for reduced computational cost many of the state-of-the-art methods use standard Naïve Bayes for classification. However, priors used in this case may degrade the results due to the unavailability of training data. Moreover, cross project performance has rarely been discussed. To address these issues, we introduce a method namely Class-Membership Information of a Term (CMT) that does not use any priors, is computationally simple, free from parameter tuning and performs better compared to the existing methods. Rigorous experiments based on three benchmark datasets demonstrate that CMT on an average performs at most 5% and 12.5% better than other state-of-the-art methods considering within and cross project classification respectively.**

***Keywords*—component; CMT; bug severity; classification;**

I. Introduction

A bug represents an error or an unexpected behavior of a software and a bug report contains all the necessary information about that bug. There are some bug reporting/tracking systems (e.g., Bugzilla[1]) which is used to manage the bug reports. It provides several information, such as, bug ID, summary and description of that bug, product and component name, reporter name, priority, etc.

Recently the increasing demand of software development is being observed due to its various applicability in scientific, business and daily-life activities. The development, specially the maintenance of these software, involves more than 85% of the total cost [1] and manual effort. Usually the developers report a lot of bugs, most of which are not real. Therefore, a manual judgment is required to identify whether a reported bug is really a bug or not. If it is really a bug, the severity level should also be judged to assign that bug to an eligible bug fixer.

To reduce the cost of software development and the burden of manual judgment, researchers have been working for several years. The early works mainly focus on automatic decision on whether a given bug is severe or not [2], [3], [4], and later, the researchers pay attention to find the level of severity of these bugs [5], [6], [7], [8], [9], [10]. For both cases, similar algorithms are proposed as discussed in the following parts of this section.

A. Severe Bug Identification

One of the earliest works [11] that divide bug reports into severe and non-severe, uses the summary of the bug reports and Naïve Bayes (NB) as a classifier under the assumption that there exists potentially significant terms in the bug reports with good discrimination ability of severity which is also assumed commonly by other researchers. Their classification has been done per component basis and obtained reasonable accuracies. They also compare the performances of summary and descriptions separately and find that summary performs better than description. Later the authors [2], have compared four well known classification algorithms and concluded that Multinomial Naïve Bayes (MNB) performs the best for severity prediction. This is even better than Support Vector Machine (SVM).

Instead of using only summary, the authors in [12] suggest to incorporate more source of information and use MNB as a classifier to improve the performance. They first group the bug reports based on their Product followed by grouping them using Component and Reporter. Then information from summary and description are extracted. Similar observation is also found in [13] where the authors have used five different information namely summary, two types of structural information (extracted from long description) and two other information such as attachment and report length. They also use MNB as classifier and calculate the posterior ($P(\psi_k \mid t_1 \cdots t_n)$) using Eq. 1 and conclude that combining all these five information improves the overall performances.

$$P(\psi_k) \prod_{i=1}^{n} P(t_i \mid \psi_k) \tag{1}$$

here, ψ_k represents the severity levels and $t_1 \cdots t_n$ are different terms of the bug report.

[1]https://www.bugzilla.org/

978-1-5386-5163-6/18/$31.00 ©2018 IEEE