Czech University of Life Sciences Prague

Faculty of Economics and Management

Department of Information Engineering

Diploma Thesis

Mining Stack Exchange to Discover Patterns of Global Crowdsourcing

Himesha Prabhakara Wijekoon

CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

DIPLOMA THESIS ASSIGNMENT

Himesha Prabhakara Wijekoon

Informatics

Thesis title

Mining Stack Exchange to Discover Patterns of Global Crowdsourcing

---

Objectives of thesis

The goal of thesis is to discover patterns of global crowdsourcing using software tools by Oracle Inc. based on big data analysis and visualization.

Methodology

A Data dump of all user-contributed content on the Stack Exchange network will be downloaded from https://archive.org/details/stackexchange. Each site is formatted as a separate archive consisting of XML files zipped via 7-zip using bzip2 compression. Each site archive includes Posts, Users, Votes, Comments, PostHistory and PostLinks.

A literature review will be performed to identify the features of crowdsourcing and crowdsourcing behavior in software development.

Since this is a big data set with around 40 GB of data, machine learning with big data analytics approach (primarily Apache Spark) will be used to mine this data set in order to discover crowdsourcing patterns. Data visualization techniques (Oracle Data Visualization Desktop) also will be used to assist discovery of patterns and presentation of results.

The proposed extent of the thesis

60-80 pages

Keywords

Stack Exchange, Data Mining, Big Data Analytics, Crowdsourcing, Pattern Discovery, Data Visualization

---

Recommended information sources

Arash Joorabchi Michael English Abdulhussain E. Mahdi , (2016), "Text mining stackoverflow: an insight into challenges and subject-related difficulties faced by computer science learners", Journal of Enterprise Information Management, Vol. 29 Iss 2 pp. -Permanent link to this document: http://dx.doi.org/10.1108/JEIM-11-2014-0109

Estellés-Arolas, Enrique; González-Ladrón-de-Guevara, Fernando (2012), "Towards an Integrated Crowdsourcing Definition" (PDF), Journal of Information Science, 38 (2): 189–200, doi:10.1177/0165551512437638

Zaharia M., et al. Learning Spark (O'Reilly, 2015)

Expected date of thesis defence

2017/18 SS – FEM

The Diploma Thesis Supervisor

doc. Ing. Vojtěch Merunka, Ph.D.

Supervising department

Department of Information Engineering

Electronic approval: 11. 1. 2018

Ing. Martin Pelikán, Ph.D.

Head of department

Electronic approval: 11. 1. 2018

Ing. Martin Pelikán, Ph.D.

Dean

Prague on 10. 03. 2018

Declaration

I declare that I have worked on my diploma thesis titled "Mining Stack Exchange to Discover Patterns of Global Crowdsourcing" by myself and I have used only the sources mentioned at the end of the thesis. As the author of the diploma thesis, I declare that the thesis does not break copyrights of any their person.

In Prague on 31.03.2018 _____

Acknowledgement

It is with pleasure and gratitude I recall all the persons who extended their support, cooperation and expertise at the completion of this research study. First and foremost, I would like to express my thankfulness to my supervisor doc. Ing. Vojtěch Merunka, Ph.D. for his wonderful encouragement and tremendous insights given which enriched my results and findings.

I wish to thank my wife Nishanthi for her encouragement, reviews and particularly for releasing me from household vows to allow me more time for the thesis. I also wish to appreciate the support of my 3 years old little daughter Sayuni who was very patient and well behaved during long working hours I spent in processing and analysing data for the research.

Last but not least I would like to express my gratitude to all my colleagues and friends who were a strength and support rendered in numerous ways during this study. I sincerely appreciate each one of you who were behind me in this endeavour.

Mining Stack Exchange to Discover Patterns of Global Crowdsourcing

Abstract

Among many popular crowdsourcing platforms, the Question & Answer website Stack Overflow in Stack Exchange Network is used daily to share knowledge globally by millions of software professionals. Therefore Stack Overflow data can reveal important patterns in global crowdsourcing beneficial for software industry. The aim of this study was to perform data mining on Stack Exchange data, to discover some of these patterns. Main focus of this research was to analyse the global user distribution and contribution, analyse contribution related to user age, classify users with regard to their involvement and identify popular topics with trends. Big data analytic techniques were used for data mining activities using Apache Spark with Python language. Oracle Data Visualization Desktop and scikit-learn python library were used for visualization. The results show that although majority of the users are from USA and India, the average contribution is higher in European countries. Key findings also reveal that younger people ask more questions than older people, while vice versa for answers. Further, users could be classified as one-timers, question askers and answer providers. Popularity and trends of different programming languages, databases and frameworks are also presented.

Keywords: Stack Exchange, Stack Overflow, Data Mining, Big Data Analytics, Crowdsourcing, Pattern Discovery, Data Visualization, Apache Spark, Oracle Data Visualization Desktop, Software Engineering

Užití sítě Stack Exchange pro vyhledávání vzorů chování globálních skupin

Abstrakt

Mezi mnoha oblíbenými platformami pro velké skupiny je webové sídlo Question & Answer burzovní sítě. Tuto síť denně používá pro sdílení znalostí mnoho milionů profesionálů v oblasti softwaru. Tato síť je hlavně zaměřena na dolování dat na burze a může poskytnout mnoho cenných informací o vzorech chování v softwarovém průmyslu. Cílem této práce bylo provést dolování dat ze sítě Stack Exchange a vyhledat některé z uvedených vzorů. Výzkum byl zaměřen na analýzu globálního rozložení uživatelů a jejich příspěvků a analyzovat tyto informace z pohledu věku uživatelů, klasifikovat uživatele podle jejich zapojení a identifikovat populární témata a příslušné trendy. Analýza velkých dat byla provedena pomocí nástroje Apache Spark s jazykem Python. Velké datové analytické techniky byly použity pro dolování dat pomocí Apache Spark s jazykem Python. Pro vizualizaci byl použit Oracle Data Visualization desktop a Python knihovna Scikit-learn. Výsledky ukazují, že přestože většina uživatelů pochází z USA a Indie, tak průměrný příspěvek do sítě je vyšší od uživatelů z evropských zemích. Klíčová zjištění také ukazují, že mladší lidé kladou více otázek než starší lidé, zatímco naopak dávají méně odpovědí. Dále mohou být uživatelé zařazováni jako jednorázoví, jako dotazovatelé a nebo poskytovatelé odpovědí. Popularita a vývojové trendy různých programovacích jazyků, databází a frameworků jsou v práci také prezentovány.

Klíčová slova: Stack Exchange, Stack Overflow, dolování dat, analýza velkých dat, chování velkých skupin (Crowdsourcing), vyhledávání vzorů, vizualizace dat, Apache Spark, vizualizace dat pomocí Oracle Desktop, softwarové inženýrstv

Table of content

List of pictures

List of tables

List of abbreviations

CSV – Comma Separated Values
DBMS – Database Management System
ER – Entity Relationship
IBM - International Business Machines Corporation
ICT - Information and Communications Technologies
JDBC – Java Database Connectivity
KDD – Knowledge Discovery in Databases
MSDN – Microsoft Developer Network
ODVD – Oracle Data Visualization Desktop
Q&A - Question and Answer
RAM – Random Access Memory
RDD - Resilient Distributed Dataset
SQL – Structures Query Language
XML – Extensible Markup Language

1 Introduction

Crowdsourcing is basically a type of participative online activity where a person or an organization requests a loosely defined group of people (crowd) to carry out tasks for them using open calls. The crowd undertakes the tasks voluntarily driven by some kind of motivation which is not supposed to be financial reasons in all the cases.(Zhao, Zhu, 2014) Nowadays crowdsourcing is very popular in global software development. Therefore a new term called "Crowdsourced Software Engineering" has also emerged to describe the phenomena of using crowdsourcing for various software engineering tasks (Mao, Capra, Harman, Jia, 2017).

Research on crowdsourcing can be categorized into three main perspectives. Those are from Participant's Perspective, Organization's Perspective and System's Perspective (Zhao, Zhu, 2014). This research aims to look at crowdsourcing from Participant's Perspective, which is mainly related to analysing the participant's behaviour. The crowd's effort and amount of contribution also should be covered under this. A good understanding of participant's behaviour will assist to identify the target crowd and create incentive strategies to motivate them.

According to Fayyad et al., data are a set of facts and pattern is an expression in some language describing a subset of the data. They further mention that extracting a pattern in general is making any high-level description of a set of data. (Fayyad, Piatetsky-Shapiro, Smyth, 1996) Therefore data mining a popular crowdsourcing platform can help to identify patterns in its participant's behaviour.

Among many popular crowdsourcing platforms used in software engineering, the Question & Answer (Q&A) website Stack Overflow[1] in Stack Exchange Network[2] is used daily to share knowledge globally by millions of software professionals. Therefore Stack Overflow data can reveal important patterns in global crowdsourcing. The identified patterns will help to get an idea about how software professionals share knowledge in a global scale. Eventually the findings will also help global software companies to formulate their strategies (positioning, recruitment, motivation etc.) while helping the crowdsourcing platforms to re-evaluate their strategies and incentive criteria.

[1] https://stackoverflow.com
[2] https://stackexchange.com

The aim of this study is to perform data mining on Stack Exchange data, to discover some of these patterns. The public data dump of all user-contributed content on the Stack Exchange Network shared in The Internet Archive[3] is supposed to be used as the main data source for this research. Even though there were many research already carried out using the Stack Exchange public data dump, this research try to uncover some more yet uncovered patterns. Thus main focus of this research is to analyse the global user distribution and contribution, analyse contribution in terms of user age, classify users with regard to their involvement and identify popular topics with trends. Hence following research questions are derived for this study.

Research Question 1: How users are distributed globally with respect to their contribution and reputation?

Research Question 2: How user contribution changes with respect to their age?

Research Question 3: Can we classify crowd into three groups: super contributors, contributors, and outliers?

Research Question 4: What are popular topics and their trends in different categories such as Programming Languages, Frameworks and Databases the crowd interested in?

This thesis is divided into 8 chapters. Chapter 2 presents the objectives and the methodology. Comprehensive literature review is included in Chapter 3. Chapter 4 describes the practical tasks carried out according to the set methodology. Results are presented, interpreted and discussed in Chapter 5. Chapter 6 has the conclusions while proceeding chapters include references and appendices.

[3] https://archive.org/details/stackexchange

2 Objective and Methodology

2.1 Objective

The objective of this thesis is to discover patterns of global crowdsourcing using software tools by Oracle Inc., based on big data analysis and visualization.

2.2 Methodology

Initially a literature review will be performed to identify the features of crowdsourcing and crowdsourcing behaviour in software development. Current status of research on crowdsourcing also should be learnt through the literature.

Then the Stack Exchange network will be studied thoroughly to understand its system and how users use it. Especially the study should focus on how users can participate and what motivates their participation. A comprehensive survey of related research which were based on publicly available Stack Exchange Data Dump[4] will also be carried out in order to find out the scope and areas which has been already covered and yet to be covered. This phase should end by specifying the scope of this research by means of revealing the research questions to be answered in this thesis.

The technical investigation should proceed after that. As per Fayyad et al. knowledge discovery in databases consists of following phases.

1. Selection
2. Pre-processing
3. Transformation
4. Data mining
5. Interpretation/evaluation. (Fayyad, Piatetsky-Shapiro, Smyth, 1996)

This research will be also based on the above mentioned approach. Therefore initially the necessary data will be downloaded from the Stack Exchange public data dump. The structure of this data should be studied and necessary data will be selected based on the relevance to the research questions identified earlier.

This selected data will be then loaded into a system (preferably a relational database) for further analysis. Pre-processing of data should be carried out whenever necessary in

[4] https://archive.org/details/stackexchange

this phase. Since the files from the data dump will not fit into memory as some of these files are even more than 10 GB, big data technologies (such as Spark) are intended to be used.

The pre-processed data will be transformed into a form which is ready for data mining. After that the data mining tasks will be carried out in order to discover crowdsourcing patterns related to the research questions derived earlier.

Data visualization techniques (i.e. Oracle Data Visualization Desktop) also will be used to assist discovery of patterns and presentation of results. Finally the discovered patterns will be interpreted and evaluated.

3 Literature Review

3.1 Crowdsourcing

This section covers a comprehensive review of literature on crowdsourcing.

3.1.1 Definition

Crowdsourcing is defined in numerous ways and contexts. It is defined as "the practice of obtaining needed services, ideas, or content by soliciting contributions from a large group of people and especially from the online community rather than from traditional employees or suppliers" in Merriam-Webster Dictionary (Merriam-Webster, 2018).

However the term crowdsourcing has been first introduced in a WiredMagazine article in June 2006 by Howe. It is defined as the action of a firm or an organization taking a task once performed by its employees and outsourcing it to a not clearly defined group of individuals in the means of an open call (Howe, 2006). The term has been set up by him to resemble "outsourcing to the crowd".

Even though the term crowdsourcing has been introduced as a buzzword mainly for online activities, there are some historical examples which can be categorized as crowdsourcing. Lynch in an article describes some early crowdsourcing activities from as early as 1714 (Lynch, 2010).

In this thesis, the main focus is towards crowdsourcing related to global software engineering. After the popularity of Web 2.0, the internet user has become an active contributor without not merely been a passive viewer of the content. Howe's definition seem to be too narrow to describe these new online activities.

According to Arolas & Ladrón-de-Guevara, the term crowdsourcing is still in its preliminary phase which undergoes frequent changes as newer applications of it emerge (Arolas, Ladrón-de-Guevara, 2012). Therefore they have tried to come up with a more generic definition in order to withstand the continuous changes as below.

"Crowdsourcing is a type of participative online activity in which an individual, organization, or company with enough means proposes to a group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task, of variable complexity and modularity, and in which the crowd should participate bringing their work, money, knowledge and/or

experience, always entails mutual benefit. The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantage that what the user has brought to the venture, whose form will depend on the type of activity undertaken." (Arolas, Ladrón-de-Guevara, 2012, p. 11)

This definition is very generic which fits well with the theme of this thesis because it can explain most of the existing crowdsourcing processes.

3.1.2 Process

Guazzini et al. mention crowdsourcing as a method of gathering the ideas, opinions or information from many autonomous contributors, in order to come up with the top solution for a specific problem (Guazzini et al., 2015).

In their paper "Evaluation on crowdsourcing research: Current status and future direction", Zhao and Zhu explains the typical process of the crowdsourcing as follows. A firm create jobs and publish them in internet for the crowd of outsiders to pick those. These outsiders complete those jobs for the firm for a financial or any other motivation. These jobs can be done by individual or a collaborative manner by the people in the crowd. After that they submit the finished jobs to the crowdsourcing platform, for the firm to assess the quality (Zhao, Zhu, 2014).

They also exclusively mention that crowdsourcing is not merely for business uses, but Non-Profit Organizations and academics also use it in their work (Zhao, Zhu, 2014). However in the topic of this thesis a firm can be an individual person in most of the cases. Otherwise it can be a person who posts a question on behalf of a firm as well. Zhao and Zhu has also illustrated the complete crowdsourcing scenario in Figure 1 in page 17.

Since many individuals in crowd can participate to solve a single problem, there should be a proper mechanism to evaluate the submissions. Reidl et al. discuss this in their paper "Rating Scales for Collective Intelligence in Innovation Communities: Why Quick and Easy" (Riedl, Leimeister, Kassel, 2010). They have found out that multi-attribute scales are better than famous reviewing criteria such as thumbs up/down or star rating for internet innovation groups. But Zhao and Zhu mention that there should be much research in this area to improve the validation of the crowd submissions (Zhao, Zhu, 2014).

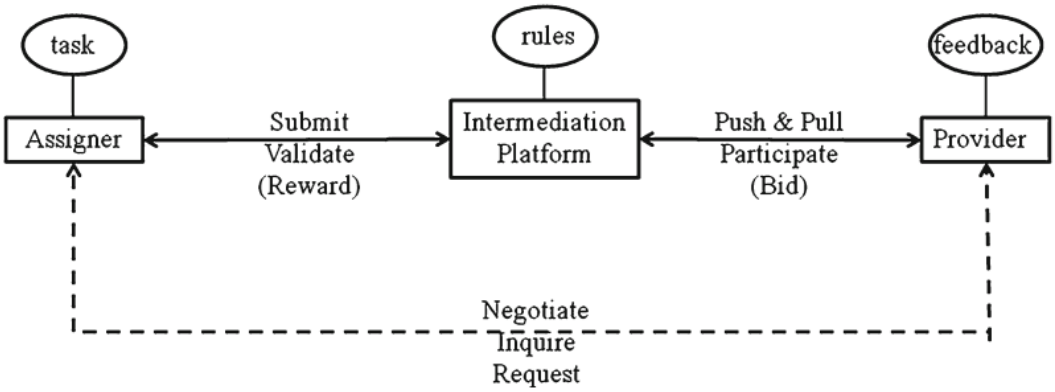

Figure 1 Components, Processes and Actions in Crowdsourcing ***[Source: Zhao & Zhu, 2014]***

3.1.3 Participant's Behaviour

Participant's involvement is the main success factor for any crowdsourcing platform. The platform should be attractive enough to motivate the crowd to participate. Therefore there should be financial or any other incentive for the participant to involve.

In their paper Zhao and Zhu discuss different research done related to participant's motivation. They mention that the reasons for crowd to participate may vary according to the context of the crowdsourcing platform (Zhao, Zhu, 2014). However they propose the need of building some theoretical frameworks and models using concepts from areas such as Psychology, Economics, and Communication etc. to fully understand this.

Stewart et al. discuss about the participation inequality of the crowd in their paper "Crowdsourcing Participation Inequality: A SCOUT Model for the Enterprise Domain" (Stewart, Lubensky, Huerta, 2010). They devise a rule based on their analysis as follows.

"*(S)uper Contributors are the 1% who consistently give super effort in terms of quantity and are driven mainly by altruism (intrinsic reward); (C)ontributors are the 66% who provide moderate effort in terms of quantity and are mainly driven by extrinsic reward; and OUT(liers) are the 33% that only provide low-level effort not sufficient for receiving an award."* (Stewart, Lubensky, Huerta, 2010, p. 33)

This research provides a good foundation to categorize crowdsourcing participants according to their contribution. The crowdsourcing platform creators and maintainers can experiment with the incentive criteria to increase the percentage of Super Contributors over the others.

Zhao and Zhu also points out that since that a lesser portion of participants account for the huge portion of effects, and most participants become not active after only after a few tasks (Zhao, Zhu, 2014).

3.1.4 Crowdsourcing Research

In their study concerning 55 academic papers on crowdsourcing, Zhao and Zhu emphasis that 50% of these papers were focused on applications (Zhao, Zhu, 2014). Figure 2 summarizes their suggestions for future research areas on crowdsourcing.

This study focuses towards the direction of analysing the participant's behaviour through Participant's Perspective.

Figure 2 The road ahead: crowdsourcing for IS scholars *[Source: Zhao & Zhu, 2014]*

3.2 Stack Exchange

This section covers a comprehensive review of literature on Stack Exchange.

3.2.1 Introduction

Stack Exchange is a group of Question and Answer websites. These separate websites cover subjects in varied fields. Stack Exchange has been founded in 2010 and consists of 133 Q&A communities as at now. According to their website they currently serve 50 million developers each month. It is also stated that Stack Exchange creates business partnerships to study, hire and collaborate with developers worldwide. They also have

products and services on developer marketing, technical recruiting, market research, and enterprise knowledge sharing (Stack Exchange Inc, 2018a).

However the main popular website Stack Overflow has been founded by Joel Spolsky and Jeff Atwood in 2008 for Q&A related to software development. Then they have formed Stack Exchange Network bringing Stack Overflow style Q&A to new topics in 2010 by extending their initial venture (Stack Exchange Inc, 2018b).

According to their website, Stack Exchange claims following figures in 2015.

Table 1 Stack Exchange Figures in 2015 ***[Source: Stack Exchange Inc, 2018a]***

Visits	101 million monthly unique visitors
Page Views	7.9 billion (5.7 billion for Stack Overflow)
Questions Asked	3.7 million
Answers Submitted	4.6 million
Registered Stack Overflow Users	5 million

As per Bhat et al. Q&A sites such as Stack Overflow, Quora, WikiAnswers, Yahoo! Answers, Naver, LiveQnA, etc., are getting extremely popular with the development of the Internet (Bhat et al., 2015). Further they explain these websites as follows.

"These are large collaborative production and social computing platforms of the Web, aimed at crowdsourcing knowledge by allowing users to post and answer questions. They not only provide a platform for experts to share their knowledge and get identified but also help novice users solve their problems effectively."(Bhat et al. , 2015 , p. 1)

Anderson et al. discuss about the long term value of Q&A websites whose service is not only limited to the question asker or the registered users, but also to other people who search things using search engines. Therefore they point out the importance of studying these platforms since they will be enormous warehouses of important knowledge (Anderson, Huttenlocher, Kleinberg, Leskovec, 2012).

3.2.2 Stack Overflow

Stack Overflow is the major Q&A website which belongs to Stack Exchange network. Stack Overflow caters wide range of computer programming subjects or topics. As illustrated in Table 1, 5.7 billion page views out of 7.9 billion is for Stack Overflow (72%). Also by 2015 number of registered Stack Overflow users has reached 5 million.

An example question which is posted in Stack Overflow is *"How do I read a file line-by-line into a list?"*[5]. Figure 3 shows how it is displayed in Stack Overflow website.

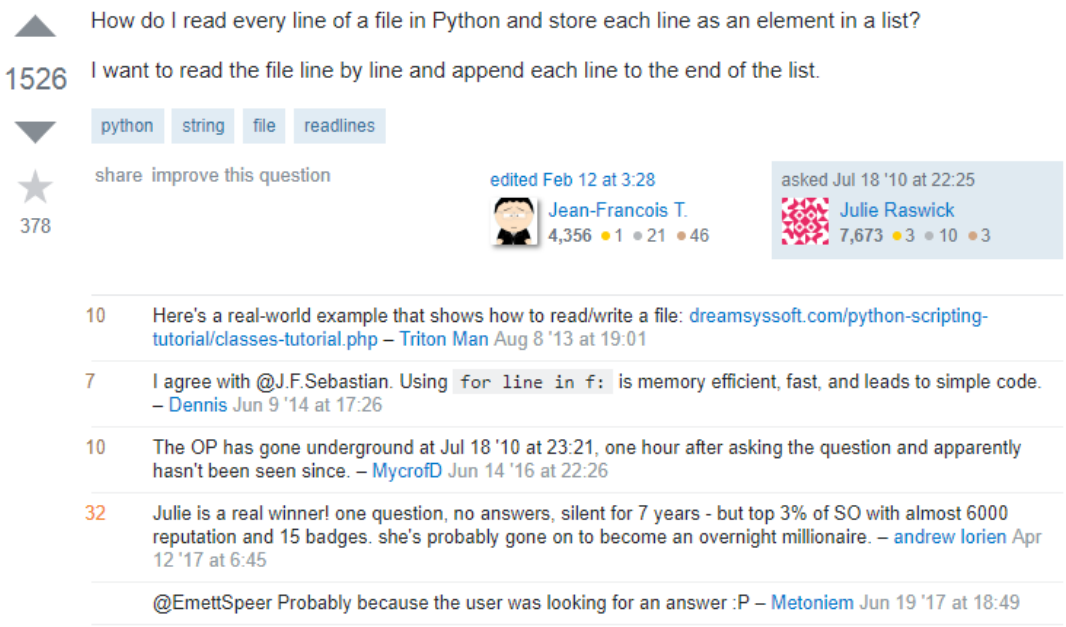

Figure 3 An Example Question from Stack Overflow *[Source: Stack Overflow]*

Figure 4 elaborates how Stack Overflow website displays answers for the above sample question.

Figure 4 An Example Answer from Stack Overflow *[Source: Stack Overflow]*

[5] https://stackoverflow.com/questions/3277503/how-do-i-read-a-file-line-by-line-into-a-list

In their survey paper named “A survey of the use of crowdsourcing in software engineering” Mao et al. comprehensively discuss about the use of crowdsourcing in software engineering. They define the term ‘Crowdsourced Software Engineering’ to represent the use of crowdsourcing practises when developing software. Mao et al. mention that Crowdsourced Software Engineering is the intersection of Software Engineering and Crowdsourcing. They devise a classification for the crowdsourcing applications used in software engineering and place Stack Overflow as a Generic/Bespoke/Q&A category as displayed in Figure 5. (Mao, Capra, Harman, Jia, 2017)

Mao et al. also mention that even though the Stack Overflow is not an integral part of software development, it provides a positive impact on software development process as a whole by providing facility to solve developer issues (Mao, Capra, Harman, Jia, 2017).

Considering all these facts, analysing data on Stack Overflow, could reveal important information on global crowdsourcing patterns. These findings can be very useful in devising strategies to effectively use crowdsourcing in software development worldwide.

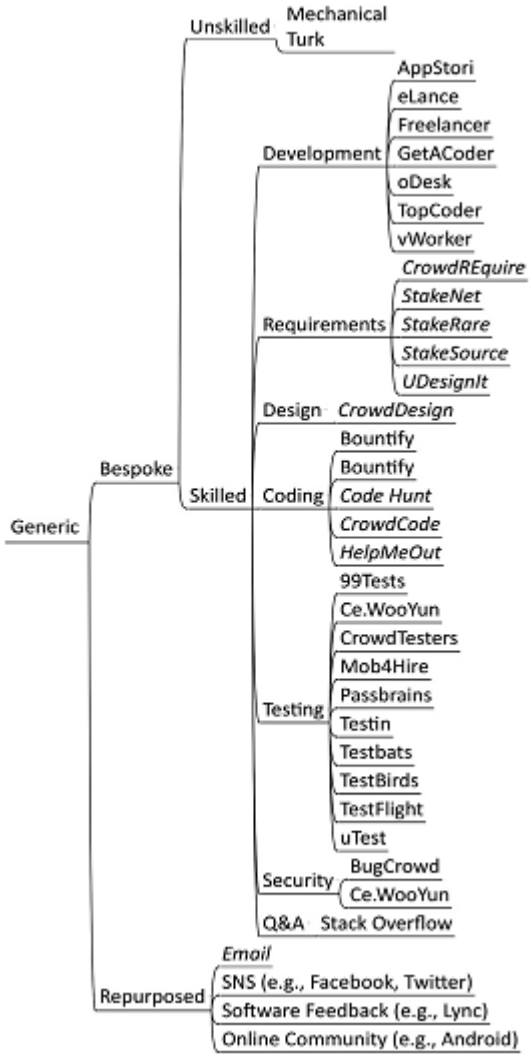

Figure 5 Scheme of Crowdsourced software engineering platforms *[Mao et al., 2017]*

3.2.3 Reputation and Moderation

Success of a community driven website depends mainly on the user contribution as discussed in section 3.1.3. Users post questions expecting the community to provide answers to those questions. But the answers should be of a certain quality and within reasonable time frame. Otherwise the platform is not sustainable. This creates two concerns.

1. How to ensure the quality of the answers.
2. How to motivate the community to provide answers.

Stack Overflow has come up with a Reputation and Moderation system to tackle the above concerns. The attitude regarding ensuring quality of Stack Overflow can be recognised through the following quote by its co-creator Atwood.

"Stack Overflow is run by you! If you want to help us run Stack Overflow, you'll need reputation first. Reputation is a (very) rough measurement of how much the Stack Overflow community trusts you. Reputation is never given, it is earned by convincing other Stack Overflow users that you know what you're talking about." (Atwood, 2009)

The reputation can be gained by posting worthy questions and valuable answers. For example a user will earn reputation when his,

- question is voted up: +5
- answer is voted up: +10
- answer is marked "accepted": +15 (+2 to acceptor)[6]

Similarly users can lose reputation as well. For an example if a question or an answer is voted down reputation points will be deducted (Stack Exchange Inc, 2018c).

Atwood states in one of his blog posts in 2009 (Atwood, 2009), that he believes in community moderation. Users with higher reputation will have higher privileges that ordinary users do not have. Some example higher privileges can be,

- Flagging posts
- Reviews posts from new users
- Edit any question or answer[7]

However the main activities of Stack Overflow (Asking, Answering and Editing) do not require any reputation (Stack Exchange Inc, 2018c).

Apart from the reputation users will receive badges for their active participation and contribution. These badges appear in the user's profile page, flair, and posts (Stack Exchange Inc, 2018d). Some example badges awarded are shown in Table 2.

The reputation and badge system provides a good motivation for the users to contribute in Stack Overflow network.

[6] This list is not exhaustive.
[7] This list is not exhaustive.

Table 2 Some Stack Overflow Badges *[Stack Exchange Inc, 2018d]*

Answer Badges	Guru	Accepted answer and score of 40 or more
	Teacher	Answer a question with score of 1 or more
Question Badges	Student	First question with score of 1 or more
	Scholar	Ask a question and accept an answer
Participation Badges	Auto biographer	Complete "About Me" section of user profile
	Pundit	Leave 10 comments with score of 5 or more
Tag Badges	Silver	You must have a total score of 400 in at least 80 non-community wiki answers to achieve this badge.
	Gold	You must have a total score of 400 in at least 80 non-community wiki answers to achieve this badge.
Moderation Badges	Marshal	Raise 500 helpful flags
	Proof Reader	Approve or reject 100 suggested edits

3.2.4 Stack Exchange Data Dump

A public data dump of all user-contributed content on the Stack Exchange network can be downloaded as zip files from The Internet Archive. A set of XML files are zipped into a one file for each website. The data includes Posts, Users, Votes, Comments, PostHistory and PostLinks. User content in Stack Exchange network is licensed via cc-by-sa 3.0[8](Stack Exchange Inc, 2018e).

3.2.5 Related Research

The publication of Stack Exchange Data Dump freely in the Internet has provided immense opportunity for the researchers to use it for their research. Especially this has benefitted the research studies on crowdsourcing and crowdsourced software engineering areas. This section describes in detail, the previous research done using the Stack Overflow data dump related to the topic of this thesis.

This research reviews 21 research papers published from year 2010 to 2017 which were based on Stack Exchange data dump. These papers focus on or more different study areas mentioned below.

- Qualities & Factors for Questions & Answers
- Reputation System
- User Contribution
- User Demographics (Location, Age, Gender)

[8] http://creativecommons.org/licenses/by-sa/3.0/

- User Characteristics
- Topic Analysis
- Question Response Time

Table 3 provides a summary about these reviewed related research papers sorted according to the published year.

Table 3 Related Research *[Source: Author]*

Published Year	Title	Study Area(s)
2010	Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow (Anderson, Huttenlocher, Kleinberg, Leskovec, 2012)	Qualities and Factors for Questions & Answers
2013	Analysis of the Reputation System and User Contributions on a Question Answering Website: StackOverflow (Movshovitz-Attias, Movshovitz-Attias, Steenkiste, Faloutsos, 2013)	Reputation System, User Contribution
2013	Building Reputation in StackOverflow: An Empirical Investigation (Bosu et al., 2013)	Reputation System
2013	Geo-Locating the Knowledge Transfer in Stack Overflow (Schenk, Lungu, 2013)	User Contribution, User Demographics (Location)
2013	Is Programming Knowledge Related To Age? An Exploration of Stack Overflow (Morrison, Murphy-Hill, 2013)	User Demographics (Age), User Contribution
2013	On the Personality Traits of StackOverflow Users (Bazelli, Hindle, Stroulia, 2013)	User Characteristics
2013	StackOverflow and GitHub: Associations Between Software Development and Crowdsourced Knowledge (Vasilescu, Filkov, Serebrenik, 2013)	User Characteristics
2013	Why, When, and What: Analyzing Stack Overflow Questions by Topic, Type, and Code (Allamanis, Sutton, 2013)	Topic Analysis
2014	Gender, Representation and Online Participation: A Quantitative Study of StackOverflow (Vasilescu, Capiluppi, Serebrenik, 2014)	User Demographics (Gender), User Contribution
2014	Min(e)d Your Tags: Analysis of Question Response Time in StackOverflow (Bhat, 2014)	Topic Analysis, Question Response Time
2014	What are developers talking about?An analysis of topics and trends in Stack Overflow (Barua, Thomas, Hassan, 2014)	Topic Analysis

2015	Effects of tag usage on question response time Analysis: Analysis and prediction in StackOverflow (Bhat et al., 2015)	Topic Analysis, Question Response Time
2015	Mining Successful Answers in Stack Overflow (Calefato, Lanubile, Marasciulo, Novielli, 2015)	Qualities and Factors for Questions & Answers
2015	One-day flies on StackOverflow Why the vast majority of StackOverflow users only posts once (Slag, De Waard, Bacchelli, 2015)	User Contribution
2015	The Synergy Between Voting and Acceptance of Answers on StackOverflow, or the Lack thereof (Gantayat et al., 2015)	Qualities and Factors for Questions & Answers
2016	Recognizing Gender of Stack Overflow Users (Lin, Serebrenik, 2016)	User Demographics (Gender)
2016	Text mining stackoverflow: an insight into challenges and subject-related difficulties faced by computer science learners (Joorabchi, English, Mahdi, 2016)	Topic Analysis
2016	Uncovering the Dynamics of Crowdlearning and the Value of Knowledge (Upadhyay, Valera, Gomez-Rodriguez, 2016)	User Characteristics, User Contribution
2017	A Journey of Bounty Hunters: Analyzing the Influence of Reward Systems on StackOverflow Question Response Times (Berger et al., 2017)	Reputation System, Question Response Time
2017	Understanding and evaluating the behavior of technical users. A study of developer interaction at StackOverflow (Ahmed, Srivastava, 2017)	User Characteristics
2017	What Do Developers Use the Crowd For? A Study Using Stack Overflow (Abdalkareem, Shihab, Rilling, 2017)	User Characteristics, Topic Analysis

Table 4 summarizes the studied phenomenon and findings of the reviewed related research papers categorized by the study area.

Table 4 Summary of Related Research *[Source: Author]*

Study Area	**Research Focus / Findings**
Qualities & Factors for Questions & Answers	• Relationship between reputation & response time and the chance of choosing an answer (Anderson, Huttenlocher, Kleinberg, Leskovec, 2012) • Factors affecting the question asker's vote (Gantayat et al., 2015) • Presentation, time and affect have an impact on the successful answer (Calefato, Lanubile, Marasciulo, Novielli, 2015)
Reputation System	• Activities which speed up reputation building (Bosu et al., 2013) • Participation contributions related to reputation (Joorabchi, English,

	Mahdi, 2016)(Movshovitz-Attias, Movshovitz-Attias, Steenkiste, Faloutsos, 2013) • High reputation users mostly provide answers while low reputation users mostly ask questions. However high reputation users ask more questions in average than low reputation users (Movshovitz-Attias, Movshovitz-Attias, Steenkiste, Faloutsos, 2013) • Study on predicting the response time as per the reputation gain (Berger et al., 2017)
User Contribution	• Analysis on how reputation affects the contribution (Movshovitz-Attias, Movshovitz-Attias, Steenkiste, Faloutsos, 2013) • Analysis on user learning affects the contribution (Upadhyay, Valera, Gomez-Rodriguez, 2016) • Study on methods to increase user contribution (Slag, De Waard, Bacchelli, 2015) • Study on how location affects user contribution (Schenk, Lungu, 2013) • User age affects the reputation (also the contribution) (Morrison, Murphy-Hill, 2013) • Men contribute more than women (Vasilescu, Capiluppi, Serebrenik, 2014)
User Demographics (Location, Age, Gender)	• User contribution is highest in Europe and North America. Then Asia which is mostly signified by India; Oceania contributes not as much as Asia but more than South America and Africa combined (Schenk, Lungu, 2013) • Guessing gender from data (Lin, Serebrenik, 2016) • Reputation increases with age (Morrison, Murphy-Hill, 2013) • Men contribute more than women (Vasilescu, Capiluppi, Serebrenik, 2014)
User Characteristics	• Users with higher reputation are more demonstrative compared to average and less reputed users (Bazelli, Hindle, Stroulia, 2013) • Stack Overflow participation has a relationship with GitHub participation of the users (Vasilescu, Filkov, Serebrenik, 2013) • Identified some misconducts in Stack Overflow by analysing the behaviour of users in humanistic perspective (Ahmed, Srivastava, 2017) • Developers use crowd based knowledge to support development tasks and to collect user feedback (Abdalkareem, Shihab, Rilling, 2017) • Newcomers and advanced users have a tendency to gain less knowledge than average users (Upadhyay, Valera, Gomez-Rodriguez, 2016)
Topic Analysis	• Topic modelling analysis which links question concepts, types, and code (Allamanis, Sutton, 2013) • Study on topics and trends (Barua, Thomas, Hassan, 2014) • Identified troublesome topics for the users (Joorabchi, English, Mahdi, 2016) • Topic affects question response time (Bhat et al., 2015) • Identified topics with good help and less help (Abdalkareem, Shihab, Rilling, 2017)
Question Response Time	• Study on predicting the response time as per the reputation gain (Berger, Hennig, Bocklisch, Herold, Meinel 2017) • Topic affects question response time (Bhat, Gokhale, Jadhav,

Pudipeddi, Akoglu 2015)
- Topic affects question response time (Bhat et al., 2015)

3.3 Scope & Research Questions

This section covers the formation of the scope and research questions of this thesis by reviewing the existing literature.

In their survey paper, Zhao et al. has identified three main perspectives for future crowdsourcing research (i.e., participant's perspective, organization's perspective, and crowdsourcing system's perspective). They further mention future directions related to participant's perspective as below.

- Motivation to participate
- Participant's behaviour (Zhao, Zhu, 2014)

They raise following two research issues to be addressed regarding participant's behaviour.

1. Crowd's effort and quantity of contribution.
2. Processes of crowdsourcing. (Zhao, Zhu, 2014)

In this thesis the focus is set on participant's perspective in the direction of participant's behaviour addressing the research issue of "Crowd's effort and quantity of contribution".

Furthermore the scope of this thesis is set towards crowdsourced software engineering. As mentioned in section 3.2.2, Stack Overflow is the main website in Stack Exchange network which is for this area. Therefore this research will use only Stack Overflow related data from the Stack Exchange data dump.

As summarized in section 3.2.5 there has been numerous research conducted on top of the Stack Exchange Data Dump related to this area. As displayed Table 4, the subject areas "User Contribution" and "User Demographics" can be identified as the related research area. The "Topic Analysis" also can be taken as it resembles what users are interested in.

There is only one research done so far to tackle issue, "how user location affects the contribution?" using Stack Exchange Data Dump. Schenk et al. in 2013 in their research has found out that contribution is highest in Europe and North America. Then Asia which is mostly represented by India; Oceania contributes not as much as Asia but more than South America and Africa combined together. However they base their research on the

transfer of knowledge. Specifically who (country) raises the question and who (country) answers it. They also insist that the Stack Overflow have become a global phenomenon. (Schenk, Lungu, 2013) However, it will be beneficial also to perform a comprehensive study on the user distribution across the globe with respect to their contribution and reputation. The understanding of the global participation will help makers of crowdsourcing applications and software companies in setting their global strategies. Therefore the first research question can be derived as below.

Research Question 1: How users are distributed globally with respect to their contribution and reputation?

Further only one research has been carried out to answer the question, "how user's age affects the contribution?" based on Stack Exchange Data Dump. Morrison et al. has found out that there is a positive correlation between age and reputation in Stack Overflow (Morrison, Murphy-Hill, 2013). They have used reputation to resemble programming knowledge. Therefore if the findings can be challenged and verified by another criteria such as number of posted questions and answers will be worthwhile. The results of such analysis will benefit the software industry to identify the age groups with highest influence. Hence the second research question can be derived as below.

Research Question 2: How user contribution changes with respect to their age?

Stewart et al. discuss about the participation inequality of the crowd in their paper by conducting a research inside global company IBM. They have witnessed a more reasonable 33-66-1 distribution different from the 90-9-1 rule for Outliers-Contributors-Super Contributors ratios (Stewart, Lubensky, Huerta, 2010). As Schenk et al. suggests it would be interesting to find whether such patterns exist in other crowdsourcing communities as well. The identified patterns will create a window for the makers of crowdsourcing applications and organizations who benefit from crowdsourced software engineering to re-evaluate their approach. So the third research question can be derived as below.

Research Question 3: Can we classify crowd into three groups: super contributors, contributors, and outliers?

There has been few research carried out in order to answer the question "What users are talking about?" (I.e. Topic Analysis) using Stack Exchange Data Dump. However only Barua et al. has approached to analysis of topics and trends (Barua, Thomas, Hassan, 2014). They have used a statistical topic modelling technique called Latent Dirichlet allocation (LDA), to automatically determine the main topics existing in user dialogs. Since this research is carried out on Stack Overflow data in the period from June 2008 to September 2010, it will be interesting to see the latest trends by analysing the newest data. Further these findings can be compared with other sources like TIOBE Programming Community Index[9] and Stack Overflow Developer Survey[10]. The results will help anyone interested in software industry to get a picture about the popular subject areas and their trends. Further the tags specified in Stack Overflow data can be used to identify the topics in this research. Hence we can derive the fourth research question as below.

Research Question 4: What are popular topics and their trends in different categories such as Programming Languages, Frameworks and Databases the crowd interested in?

Another observation from the previous research is that user contribution is directly measured through the reputation value calculated by Stack Exchange Reputation System. Reputation measurement can also be manipulated by users who plays around with the gamification methods of Stack Overflow (Ahmed, Srivastava, 2017). Therefore it is not good to take the reputation as the only measure of knowledge of users. However in this research the number of questions and answers posted will be also used to represent the contribution. Hence the number of answers posted can be taken as a representor for user knowledge.

When comparing these measurements across users, there is a need of normalization of the figures according to the length of membership for the users. For example Morrison and Murphy-Hill has used the Reputation per Month without just taking Reputation as the

[9] https://www.tiobe.com/tiobe-index/
[10] https://insights.stackoverflow.com/survey/2017

measurement in their research (Morrison, Murphy-Hill, 2013). Similarly number of answers posted per month and number of questions posted per month can be used in this research in addition to the reputation.

3.4 Technology

This section covers a review of literature on technology and tools which falls within the scope of this research.

3.4.1 Big Data

The word "Big Data" has been first introduced in 1990s. In a New York Times article, Lohr gives credit to John Mashey for either introducing or making the term popular (Lohr, 2013). Currently big data has become very popular and even become a hype word as many people use it even for marketing purposes. Regardless of its popularity the term has been used very vaguely in the beginning lacking a formal definition (Ward, Barker, 2013).

Therefore several researchers has come up to derive a proper definition for the term big data. Greco et al. has defined big data as follows after reviewing large number of big data research.

"Big Data is the Information asset characterised by such a High Volume, Velocity and Variety to require specific Technology and Analytical Methods for its transformation into Value." (Greco, Grimaldi, 2016, p. 131)

The initial characteristic about this definition is that it does not quantify what is high volume, velocity or variety. Therefore it is subjective to the context of application. On the other hand, one can think of big data as data which require special (big data) technology and methods to process them. The usage of conventional databases, processing methods, analytical tools may not be suitable due to limitations or sometimes cannot be entirely used on this kind of data.

3.4.2 Apache Spark

Apache Spark[11] has as its architectural base as the resilient distributed dataset (RDD). RDD is a read-only multiset of data items spread over a cluster of computers, which is maintained in a fault-tolerant way. Spark been developed at University of California, Berkley's AMPLab has been first introduced by Zaharia et al. in 2012. Then Spark codebase was provided to the Apache Software Foundation for maintenance. Zaharia et al. boasts that Spark beats Apache Hadoop[12] by up to twenty times in iterative applications and can be used interactively to query hundreds of gigabytes of data. (Zaharia, Chowdhury, Das, Dave, 2012)

Spark RDDs can be used via a language-integrated APIs in Java, Scala, Python, and R together with an optimized engine which has provisions for general execution of graphs (Meng et al., 2015). Zaharia et al. introduces Spark as an open source computing framework which combines streaming, batch, and interactive big data jobs that can be used to create novel applications (Zaharia et al., 2016).

As per Zaharia et al. the main benefits of Spark are,

- Ease of application development due to unified API
- The ability to efficiently combine processing tasks; as Spark can execute various functions over the same data, frequently in memory.
- Ability to support creating novel that were not possible with earlier systems. (Zaharia et al., 2016)

Further they mention that *"The very nature of 'big data' is that it is diverse and messy; a typical pipeline will need MapReduce-like code for data loading, SQL-like queries, and iterative machine learning."*, to promote Spark Framework.

The Spark Core consists of modules for task scheduling, memory management, fault recovery, interacting with storage systems. RDD programming model which providing distributed collections of objects and functions is also belongs to Spark Core. However Spark also features assortment of higher-level libraries which can be utilized for different applications. This unified Spark stack is illustrated in Figure 6. (Zaharia, Karau, Konwinski, Wendell, 2015)

[11] https://spark.apache.org/
[12] http://hadoop.apache.org/

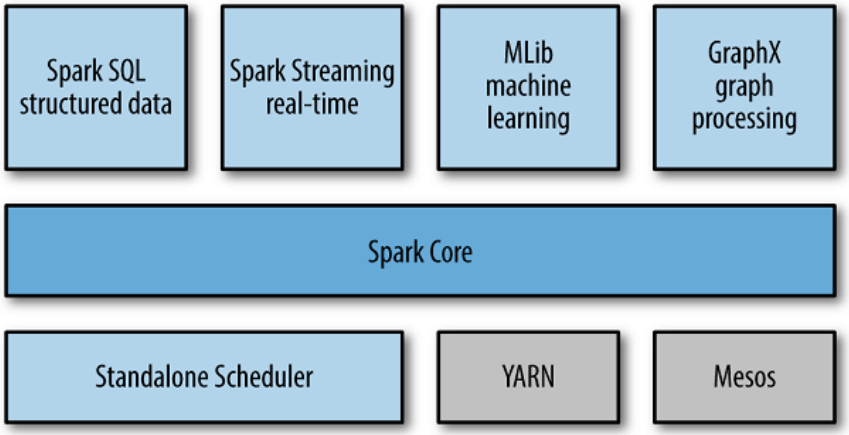

Figure 6 The Spark Stack ***[Source: Zaharia et al., 2015]***

Spark supports loading and saving of data in variety of file formats. Some common formats supported are text files, JSON, CSV, SequenceFiles, Protocol buffers and Object files. These formats can be unstructured, semi structured, or structured. (Zaharia, Karau, Konwinski, Wendell, 2015)

Further there are free libraries for parsing and querying different formats of data. Especially the Databricks Inc.[13] has created libraries for CSV and XML formats. The packages spark-xml[14] and spark-csv[15] developed by Databricks Inc. can be used to read files in local or distributed filesystem as Spark Data Frames.

There is also the possibility to connect Spark to many widespread database using Hadoop connectors or custom Spark connectors. Four main connectors are JDBC, Cassandra, HBase and Elasticsearch. Therefore MySQL database can be connected through JDBC using a MySQL JDBC Connector driver. Then by using Spark SQL, the data can be loaded to special type of RDD called SchemaRDD. SchemaRDD knows the schema of the rows and therefore can store data efficiently than normal RDDs. They also provide the capability to run SQL queries. Further Spark SQL provides facility to integrate SQL and regular Python/Java/Scala code, with the ability to join normal RDDs and SQL tables. (Zaharia, Karau, Konwinski, Wendell, 2015)

[13] https://databricks.com
[14] https://github.com/databricks/spark-xml
[15] https://github.com/databricks/spark-csv

Spark SQL supports parallelism by allowing to specify the number of partitions[16]. This can be very useful when reading from a database table which has millions of records. Especially when there is a need to perform the aggregated operations and joins on top of the loaded data. Using Spark with a traditional database can also make the queries run multiple times faster. Rubin has done an experiment in 2016 to prove this using MySQL and Spark (Rubin, 2016).

The reputed market research company Forrester has also evaluated Spark framework in some of their reports. They have identified Spark as powerful and promising as early as 2015 (Gualtieri et al., 2015). In a report in 2017 they have observed that most of predictive analytics and machine learning vendors have moved from a Hadoop to Spark because of the availability of machine learning libraries and faster in-memory processing (Gualtieri, Sridharan, Kisker, Austin, 2017).

3.4.3 Python

Python[17] is an interpreted high-level programming language for general-purpose programming. However it is also a popular choice for scientific computing. According to TIOBE Programming Community Index[18] it is the 4th most popular language in the world as of March 2018. Respectively Java, C and C++ are much popular than Python. (TIOBE software BV, 2018a)

Python has libraries such as Numpy, Matplotlib, Scipy, scikit-learn and pandas to for data visualization and analysis. Numpy stands for Numerical Python and provides functionality to perform complex mathematical operations on the data. Numpy assists to develop Gaussian distribution and normal distribution on data. Matplotlib is a library which can be used to visualise data in the forms of graphs, plots or histograms. Its sub-module pyplot is a good tool for scatter plot of your data on a 2d graph. The scikit-learn library offers a set simple and efficient tools for data mining and data analysis. Python's sub-modules offers flexibility and a lot of customization power to explore and understand data making it one of the best and most used language in data science and machine learning.

[16] http://spark.apache.org/docs/latest/sql-programming-guide.html#jdbc-to-other-databases
[17] https://www.python.org/
[18] https://www.tiobe.com/tiobe-index/

Spark has its API named PySpark[19] exclusively for Python. PySpark exposes the Spark programming model to Python. This facilitates Python scripts to be executed via Spark engine. Therefore this empowers user with facilities of both technologies.

3.4.4 Data Visualization

Data visualization includes the construction and analysis of the visual representation of data. In this context data means *"information that has been abstracted in some schematic form, including attributes or variables for the units of information"* (Friendly, 2009, p. 2). The main objective of data visualization is to present information clearly and efficiently through graphical means. This most often leads to identify things which cannot be easily observed using conventional reports, tables, spreadsheets etc.

The free Oracle Data Visualization Desktop[20] software provides facility to easily create rich, interactive visuals by connecting to various data sources. The data sources can be CSV files, Excel files or databases etc.

[19] http://spark.apache.org/docs/2.1.0/api/python/pyspark.html
[20] http://www.oracle.com/technetwork/middleware/oracle-data-visualization/downloads/oracle-data-visualization-desktop-2938957.html

4 Practical Part

The practical work carried out in this research is described in this chapter. The subchapters are based on the first four steps mentioned by Fayyad et al. for Knowledge Discovery in Databases as illustrated in Figure 7. The Interpretation / Evaluation phase will be covered in chapter 5.

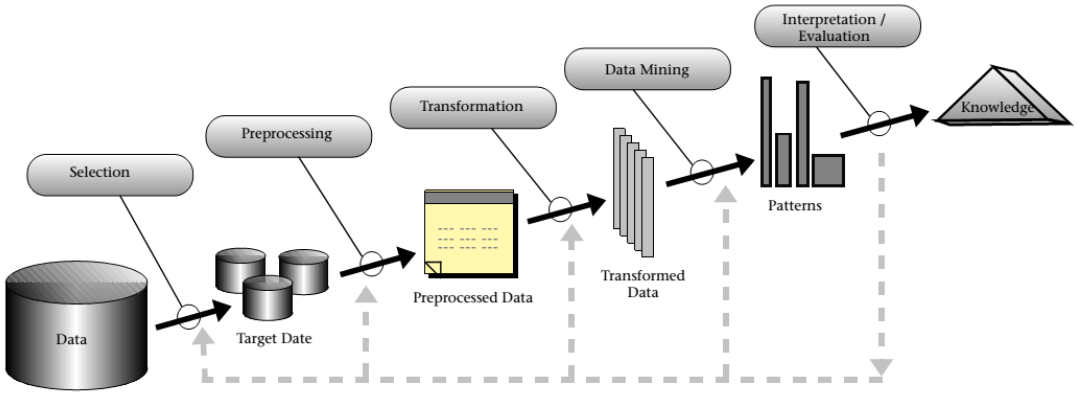

Figure 7 An Overview of the Steps That Compose the KDD Process ***[Source: Fayyad et al. , 1996]***

4.1 Selection

The Stack Exchange Data Dump which was publicly shared in The Internet Archive consisted of 353 zip files for different websites in the Stack Exchange network. However since this research focused only on Stack Overflow website data, the files related to that has been downloaded. This data dump is frequently being updated. The dump which was downloaded has been published on 8th December 2017. However the latest data it has is until 3rd of December 2017.

The following zip files were downloaded from the archive which is relevant for this research.

- stackoverflow.com-Tags.7z
- stackoverflow.com-Users.7z
- stackoverflow.com-Posts.7z

Then these files were extracted and the following relevant xml files were retrieved for further analysis.

- Tags.xml (4.32 MB)
- Users.xml (2.36 GB)

- Posts.xml (56.3 GB)

Since Users.xml and Posts.xml files are very large, these files cannot be opened by a simple text editor for inspecting.

4.1.1 Schema of the Data

The data dump also has shared a text file named readme.txt which specifies the schema of the xml files. Since Users.xml and Posts.xml cannot be opened by a simple text editor as mentioned in 4.1, the schema information were taken from the readme.txt. The schema of the concerned data is illustrated as an Entity Relationship diagram in Figure 8.

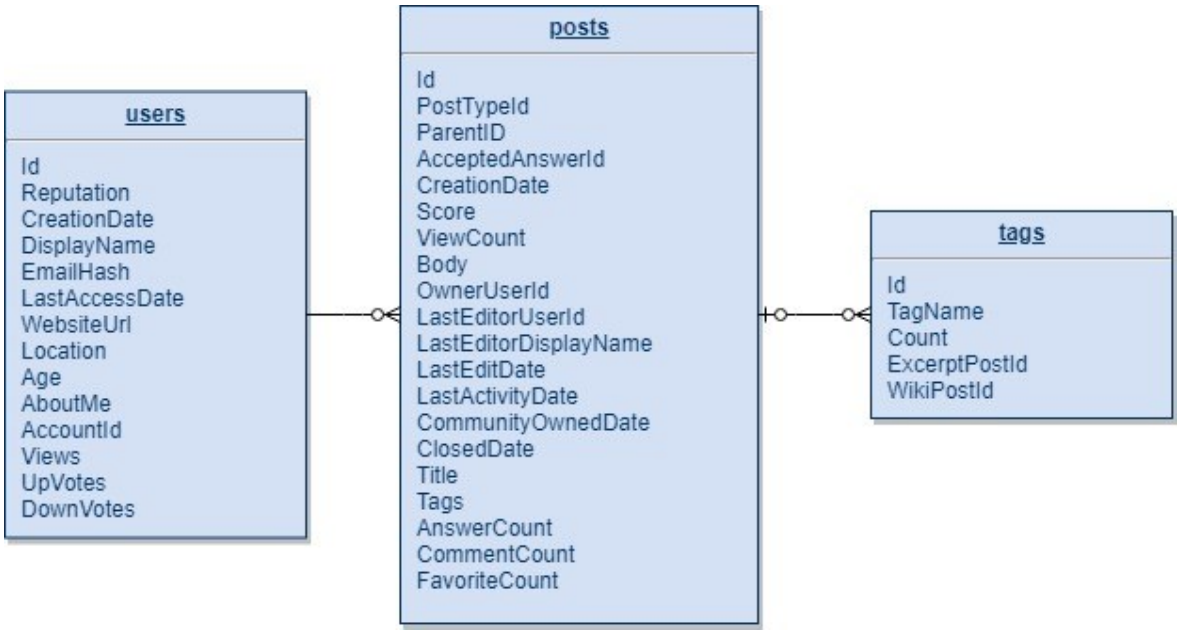

Figure 8 ER Diagram of the Original Schema ***[Source: Author]***

However for the analysis of this thesis, all the columns were not required. The presence of needless data affects the performance of mining tasks. Therefore the fields Body, LastEditorDisplayName and Title were not selected for the analysis.

4.2 Pre-Processing

Initially the structure of the xml files were studied in this phase. The structure for Tags.xml is presented as an example (refer Figure 9). It has the xml root tag as “tags”. Then all the fields for a specific item is specified as repeating child xml element named “row”. However the fields for a specific tag like “.net” is represented as xml attributes for each “row” element. Therefore this xml is represented as a flat xml, as it uses attributes for the data representation.

```
<?xml version="1.0" encoding="utf-8"?>
<tags>
  <row Id="1" TagName=".net" Count="261481" ExcerptPostId="3624959" WikiPostId="3607476" />
  <row Id="2" TagName="html" Count="710104" ExcerptPostId="3673183" WikiPostId="3673182" />
  <row Id="3" TagName="javascript" Count="1519901" ExcerptPostId="3624960" WikiPostId="3607052" />
  <row Id="4" TagName="css" Count="508419" ExcerptPostId="3644670" WikiPostId="3644669" />
  <row Id="5" TagName="php" Count="1147808" ExcerptPostId="3624936" WikiPostId="3607050" />
</tags>
```

Figure 9 Excerpt from Tags.xml *[Source: Author]*

As mentioned in section 3.4.2 the spark-xml package can be used with Spark to read xml files in local or distributed filesystem as Spark Data Frames. But the use of xml attributes in the xml files has created a limitation in this regard. The specific issue here is that spark-xml fails to process the self-closing tags. For an example each "row" element in Tags.xml is represented with a self-closing tag which has only attributes without any tag values. This issue is a known issue for spark-xml[21] but not has been fixed at the time of this research. But the spark-xml community has suggested to use a feature called "explode" in spark-xml as a workaround for this issue[22] to break the xml attribute values.

The "explode" feature is dependant on the computing memory. Therefore "explode" feature cannot be used for large files which doesn't fit the memory of the computer if you are running Spark in a single machine (non-cluster mode). Since this research is carried out running Spark in a single computer with 4GB memory, the xml structure used in the data dump has prevented the ability of using xml files directly with Spark and spark-xml. Further, since xml is not a breakable file structure such as csv, the large xml files cannot be directly loaded into Spark for data mining.

Therefore the researcher had to load the raw data into another format which Spark can utilize its in-memory processing and parallelization power. As discussed in section 3.4.2, a relational database which Spark supports was used to store data in this regard. Therefore MySQL database was chosen and used for this purpose. In order to write the scripts, Python language is selected based on its features discussed in section 3.4.3.

Then Python scripts were written to load data from xml to MySQL database for each xml file. These Python scripts were then executed using spark-submit script which is located in Spark's bin directory. Since "explode" feature cannot handle large files, XML

[21] https://github.com/databricks/spark-xml/issues/92
[22] https://github.com/databricks/spark-xml/pull/149

files are broken into small files dynamically in the scripts. The script used to load data from User.xml into a MySQL table is included in Appendix I as an example.

Three tables, namely tags, users and posts were created in MySQL database and populated using data from the xml files. The Table 5 shows the number of records loaded into respective MySQL tables.

Table 5 Number of Records Loaded into MySQL Tables *[Source: Author]*

MySQL Table Name	Number of Records
tags	50812
users	7,408,959
posts	38,360,000

4.3 Transformation

Conversion of the data into appropriate forms is a necessary phase before starting data mining activities. This section describes the specific data transformation routines performed in this research.

4.3.1 Extraction of Country Name from the Location

The initial transformation task performed is the extraction of country name from the location of a user. This is an attribute construction routine. In the raw data, location is not specified in any standard format. It is a free text field and most of the users has left it blank (unspecified). Only 1,461,297 users have specified their location out of 7,408,959 total users loaded into MySQL. That is roughly 19.7% of total users.

Table 6 Some Example Location Texts *[Source: Author]*

United States	Albury, Australia	San Francisco, CA
Illinois	Salt Lake City, UT, United States	Europe
That is Classified	HeLL	USA
England	United Kingdom	Great Britain

Extraction of country from the location is not straightforward as users has specified location in different formats. Some different types of location texts are shown in Table 6 to show this situation. Therefore a special python programme was implemented to extract the country accurately with the help of a free and open source third party Python library named

geodict[23]. The geodict library can pull location information from unstructured text. In this research geodict library has been used with some customizations to extract country from the location field. It is also observed that most of the users in United States, have specified only up to state name as location and some countries have multiple names. These situations had to be specially addressed in this transformation.

The location of 1,172,495 users were identified and saved in a new database table named user_countries with the schema specified below. This is 15.83% from all users and 80.24% of all the users who have specified their location.

user_countries (Id, Country, Age, CreationDate, LastAccessDate, Reputation, AccountId)

4.3.2 Aggregation

Aggregation is another transformation routine carried out on the data. The level of aggregation chosen has been varied as per the research question under analysis. For an example in order to analyse the country wise user participation, the user data had to be grouped by country name. Then the measures like country wise total number of users, average reputation for users and average age were calculated.

Especially since tables such as users and posts have millions of data, Spark with Python API was chosen for this purpose by leveraging the suitability mentioned in sections 3.4.2 and 3.4.3. As loading the whole table into memory cannot be done due to the size, the partition aware loading feature of Spark was utilized. Then groupBy function and other built-in aggregate functions like count, avg in Spark were used to perform aggregation. Finally the aggregated data has been merged and saved into a single CSV file as per the data mining requirements. A python script which has been used to aggregate country wise user data is included in Appendix II.

The Table 7 summarizes the aggregation activities carried out for each research question.

Table 7 Aggregation Activities ***[Source: Author]***

No.	Research Question	Aggregations	Aggregated Functions Used
1	How users are distributed globally with respect to their	• Posts on User	• Count of Questions per

[23] https://github.com/petewarden/geodict

	contribution and reputation?	• Users on Country	User • Count of Answers per User • Count of Users • Average Reputation • User per 1000 Capita
2	How user contribution changes with respect to their age?	• Posts on User • Users on Age based on Joined Year	• Count of Questions per User • Count of Answers per User • Count of Users • Average Number of Answers per Age • Average Number of Questions per Age • Average Reputation per Age
3	Can we classify crowd into three groups: super contributors, contributors, and outliers?	• Posts on User • Posts on User Id	• Count of Questions per User • Count of Answers per User
4	What are popular topics and their trends in different categories such as Programming Languages, Frameworks and Databases the crowd interested in?	• Distinct Tags in Posts • Distinct Tags in Posts per Year	• Count of occurrences of distinct tags in posts

Grouping posts by user has been used in many of the analyses. It is used to calculate the number of questions and answers posted by individual users. The result of this aggregation has been saved as a CSV file as shown in Figure 10.

```
OwnerUserId,Questions,Answers
136059,1,238
109554,6,21
173355,4,58
52051,67,76
192044,112,58
156892,77,27
184212,14,20
```

Figure 10 Extract from User Q&A Counts CSV File *[Source: Author]*

Similarly all the necessary aggregated data required for research questions were generated with the help of the Python scripts executed on Spark engine. Therefore after this phase multiple summarized CSV files were created.

4.3.3 Merging

Merging data from different sources also has been carried out in the research. Data aggregated in section were sometimes needed to be merged in order to perform data mining. As mentioned in section 3.4.2, Spark provides facility to join RDDs regardless of how data is loaded into the RDDs. This feature is used to get further attributes of users by joining relevant RDDs. For example the data from User Q&A Counts CSV File is merged with user table in MySQL to get the age specific Q&A counts for users as displayed in Figure 11.

```
_Id,_Age,OwnerUserId,Questions,Answers
4414510,27,4414510,10,2
4415422,17,4415422,1,0
4416749,26,4416749,4,5
4419908,40,4419908,3,33
4420484,21,4420484,5,0
4428123,23,4428123,8,7
```

Figure 11 User Q&A Counts with Age *[Source: Author]*

Further the world population data for year 2015 published by United Nations, Population Division were merged with aggregated country wise data to calculate the "users per 1000 capita" figure for each country (United Nations, Department of Economic and Social Affairs, 2017). All merged results were saved as CSV files to be used in data mining.

4.4 Data Mining

In this phase appropriate data mining techniques were selected and used in order to search for patterns of interest.

4.4.1 Summarization

The major data mining technique used in this research is summarization. This basically involves providing a more compact representation of the data set, including visualization.

For the numerical data, descriptive summary statistics helps a lot in understanding the distribution of data. Spark comes with in-built functions for statistical analysis. For an example, the function “describe” in Spark returns a DataFrame containing information such as number of non-null entries (count), mean, standard deviation, and minimum and maximum value for each numerical column. These Spark functions were used appropriately to come up with the descriptive statistics for numerical data.

In some cases the scikit-learn along with Numpy and Scipy Python packages were also utilized to extensively study the distributions of the data. For an example the boxplots and histograms were created using scikit-learn library to learn more about the data. Especially this was used to identify the outliers and extreme values.

Further, the summarized data in the form of CSV files were loaded into Oracle Data Visualization Desktop (ODVD) software to create visualizations in order to identify patterns. The ODVD software has been used as a sandbox to play around with the data by representing those data in different graphs for pattern seeking as well. Dynamic filtering of data to be loaded into visualizations has been used extensively when mining for patterns. A screenshot of ODVD software is displayed in Figure 26 in Appendix IV.

4.4.2 Clustering

Clustering technique has been basically used to answer the research question 3, which is focused on classifying crowd into three groups based on their contribution. The scikit-learn python library has been used for this purpose as per the discussion based on section 3.4.3. Even though ODVD also has a clustering solution, the unified tool stack in Spark was much attractive and helpful for the clustering.

Among many other clustering algorithms, the K-Means clustering algorithm seems to generally have high efficiency (Shah, Jivani, 2013)(Indhu, Porkodi, 2018). Since the clustering should be done on top of over 3.8 million data points (38,360,000), the efficiency of the algorithm is critical. Therefore K-Means algorithm has been used to identify the presence of any clusters. The Python script created for this purpose is included in the Appendix III.

Initially random sample from the original dataset is taken for cluster analysis. The distributions of number of questions posted by users and number of answers provided by users were further studied using Python libraries like Scipy, Numpy and Matplotlib to prune the dataset for final cluster identification.

5 Results and Discussion

5.1 Global User Distribution and Contribution

The research question "How users are distributed globally with respect to their contribution and reputation?" is analysed in this section.

As mentioned in section 4.3.1, the country names of 1,172,495 users of Stack Overflow (15.83% from total users) were identified. The analysis in this section is based on this subset of users.

5.1.1 Distribution of Users across Globe

The geodict library could identify 240 country names. Out of these, 205 country names were identified in the subset under analysis. However top 50 countries sorted in the descending order of user count are presented in Table 8.

Table 8 Top 50 Countries with Users *[Source: Author]*

Country	Count	Cluster	Country	Count	Cluster
UNITED STATES	256470	5	VIET NAM	8359	2
INDIA	214574	5	ROMANIA	8012	2
UK	74955	4	BELGIUM	7683	2
GERMANY	39550	4	SWITZERLAND	7406	2
CANADA	37576	4	ARGENTINA	7277	2
FRANCE	30470	4	SINGAPORE	7168	2
CHINA	30164	4	PORTUGAL	7103	2
AUSTRALIA	22434	3	IRELAND	6906	2
RUSSIAN FEDERATION	22070	3	DENMARK	6846	2
BRAZIL	20070	3	SRI LANKA	6508	2
PAKISTAN	18661	3	JAPAN	6352	2
NETHERLANDS	18170	3	MEXICO	6327	2
INDONESIA	14055	3	NEW ZEALAND	6191	2
UKRAINE	13391	3	MALAYSIA	6179	2
POLAND	13027	3	TAIWAN	5693	2
BANGLADESH	12825	3	NORWAY	5475	2
SPAIN	12364	3	NIGERIA	5288	2
PHILIPPINES	12288	3	GREECE	5121	2
ITALY	12194	3	AUSTRIA	5070	2
SWEDEN	11928	3	COLOMBIA	4765	2
IRAN	11862	3	SOUTH KOREA	4708	2
SOUTH AFRICA	9198	2	CZECH	4405	2

			REPUBLIC		
ISRAEL	9002	2	FINLAND	4251	2
TURKEY	8697	2	NEPAL	4148	2
EGYPT	8527	2	BULGARIA	4134	2

As observed United States and India has marginally very high number of users which is more than 200,000. Collectively they represent 40% of total users. They are categorized as countries in Cluster 5. Cluster 4 countries have users between 30,000 and 75,000. UK, Germany, Canada, France and China belongs to this category. Even though China has the world's highest population, its participation is not matching with the population. It must be due to language issues. This can be same for Russian Federation. Another notable observation is there are only 78 countries with more than 1000 identified users. Cluster 2 represents countries with more than 3000 and only some of them are in top 50 list. Cluster 1 represents countries with less than 3000 users which is not even included in the Table 8.

The number of users from each country may depend on the population. Therefore the above representation does not make a clear idea how the global software professionals are attracted to participate in crowdsourcing. Therefore above data has been merged with world population data for year 2015 published by United Nations, Population Division (United Nations, Department of Economic and Social Affairs, 2017). Then users per 1000 capita figure has been calculated for each country for further analysis.

The map in the Figure 12 displays how users per 1000 capita changes across the globe and the Table 9 presents the top 50 countries with users per 1000 capita in descending order. The main observation compared with user count ranking is United States falling down to 17th position while India does not even qualify in top 50. However UK shows consistency in both and the biggest (population wise) country having highest participation. Iceland becomes the number one even though it does not even have sufficient users to be listed in the first list. The main conclusion that can be derived is that most European countries have higher participation per capita generally. The countries like New Zealand, Singapore, Israel, Canada and Australia are also among the high participating countries.

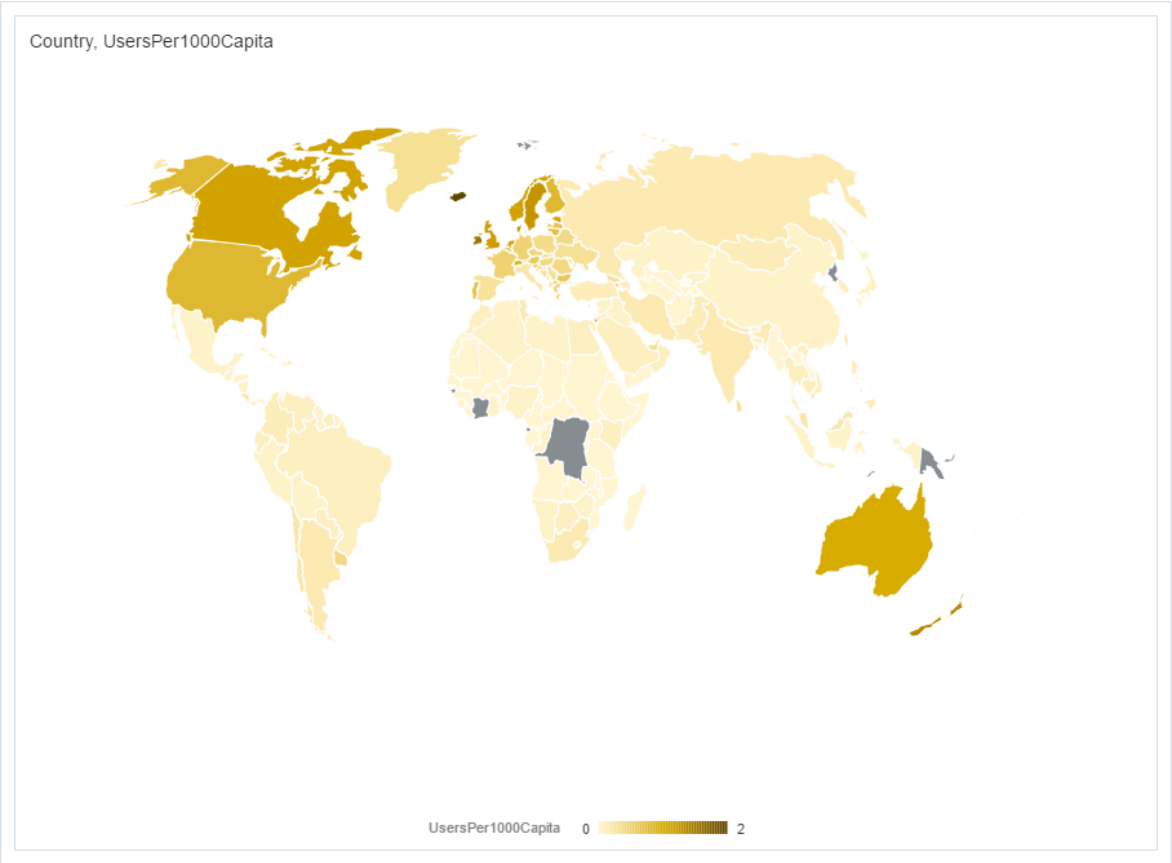

Figure 12 Users per 1000 Capita *[Source: Author]*

Table 9 Top 50 Countries with Users per 1000 Capita *[Source: Author]*

Country	UsersPer1000Capita	Country	UsersPer1000Capita
ICELAND	1.91677	CROATIA	0.537297
MALTA	1.585535	CYPRUS	0.484933
IRELAND	1.469328	GERMANY	0.484042
NEW ZEALAND	1.341631	FRANCE	0.472717
SINGAPORE	1.29497	HONG KONG	0.462205
SWEDEN	1.221685	GREECE	0.456507
DENMARK	1.203439	MACEDONIA	0.438127
UK	1.146152	ARMENIA	0.416531
ISRAEL	1.116244	CZECH REPUBLIC	0.415419
NETHERLANDS	1.072704	ROMANIA	0.403087
NORWAY	1.052918	BELARUS	0.395961
CANADA	1.045238	URUGUAY	0.37942
ESTONIA	1.008119	HUNGARY	0.372039
LUXEMBOURG	0.959874	SLOVAKIA	0.359604
AUSTRALIA	0.942623	POLAND	0.34044
SWITZERLAND	0.890169	GEORGIA	0.322154
UNITED STATES	0.801646	SRI LANKA	0.314183
FINLAND	0.775452	SERBIA	0.312271

LITHUANIA	0.718981	UNITED ARAB EMIRATES	0.299968
PORTUGAL	0.68177	UKRAINE	0.299859
LATVIA	0.6815	COSTA RICA	0.285575
BELGIUM	0.680638	SPAIN	0.266479
SLOVENIA	0.679106	BOSNIA AND HERZEGOVINA	0.257921
AUSTRIA	0.584192	TAIWAN	0.242402
BULGARIA	0.575975	ALBANIA	0.238767

5.1.2 Contribution Related to Country

The user contributions in the means of average reputation per user, average number of questions and answers posted per user from each country has been analysed in this section. The Table 10 summarizes the rankings of countries which falls into top 20 of each category and has more than 500 users along with Russian Federation and India for their significance. The cells in blue background colour displays the ranks within 20 while cells with pink background displays rankings greater than 20 for the respective category. The global map for reputation is included in Appendix IV as supplementary reference.

Table 10 Country Rankings for Contribution *[Source: Author]*

Country	Reputation Rank	Answer Rank	Question Rank
SWITZERLAND	1	1	6
UK	2	4	5
GERMANY	3	3	14
SWEDEN	4	10	13
GUATEMALA	5	55	97
MALTA	6	15	3
ISRAEL	7	2	1
AUSTRIA	8	6	15
NORWAY	9	14	9
NETHERLANDS	10	5	21
AUSTRALIA	11	12	16
NEW ZEALAND	12	13	18
FINLAND	13	11	49
CZECH REPUBLIC	14	7	4
BULGARIA	15	8	38
DENMARK	16	18	7
UNITED STATES	17	22	35

SLOVENIA	18	16	2
CANADA	19	25	24
SLOVAKIA	20	9	20
POLAND	21	17	25
BELGIUM	22	19	10
LATVIA	23	28	17
IRELAND	24	30	11
ITALY	27	23	8
PERU	32	20	55
RUSSIAN FEDERATION	35	38	54
CYPRUS	44	36	19
LEBANON	53	50	12
INDIA	64	58	56

As reputation and answer ranking relates to knowledge sharing, respectively Switzerland has become top country in both of the rankings while closely followed by UK and Germany. Sweden, Austria and Israel are among top 10 of both of the rankings with most of other European countries. New Zealand, Austria and Canada contributes much as well.

However India and Russian Federation has less contribution despite their high population. The other important observation is most of the countries who are reputed and good answer providers are also good at asking questions. However Italy, Ireland, Latvia, Lebanon are basically question askers but not answer providers. Meanwhile Finland, Netherlands and Bulgaria has higher reputation and answering rate, but not asking many questions.

5.1.3 Discussion

In both user participation and contribution European countries along with Israel, Australia, Canada and New Zealand are highlighted from the rest of the world. These findings were cross evaluated by comparing with the ICT Development Indexes of countries provided by United Nations (United Nations International Telecommunication Union, 2017). This indexed is calculated based on the 11 Information and Communications Technologies (ICT) indicators, grouped in three clusters: access, use and skills. The major difference found was the underperformance of crowdsourcing activities of countries like South Korea and Japan which enjoys good global ICT rankings. This situation can be further proven by comparing the findings with the IMD World Digital Competitiveness Ranking 2017 (IMD

World Competitiveness Centre, 2017). Even though this must be further analysed, one reason can be the language barrier. Presence of some other popular alternatives to Stack Overflow also can be also another reason. Under presence of China and Russian Federation can be also due to this.

5.2 User's Age and Contribution

The research question "How user contribution changes with respect to their age?" is analysed in this section.

Only 665,301 out of 7,408,959 total users have specified their age in Stack Overflow. That is roughly 9% from total users. Mean age of a user is 30.9 years with a standard deviation of 8.6. This is very close to the mean age of 30.3 years and standard deviation of 8.2 what Morrison et al. has found out in their research in 2013 (Morrison, Murphy-Hill, 2013). This implies that the age distribution of the users of Stack Overflow has not changed from 2013 to 2017.

Further analysis on the distribution of the age of users revealed the presence of outliers and extreme values. Some people have stated ages even higher than 80, 90 years which can be quite abnormal. Therefore study has been done using boxplot and histograms (refer Figure 14 and Figure 14) to find out the best range for our analysis. As per the observations, it is decided that the focus range should be age between 14 and 50 years.

There are 645,180 observations in this range which has omitted only 20,121 extreme values. Mean age of the filtered data set is 30 years with a standard deviation of 6.7. Standard deviation is improved from 2 units as a result of pruning outliers.

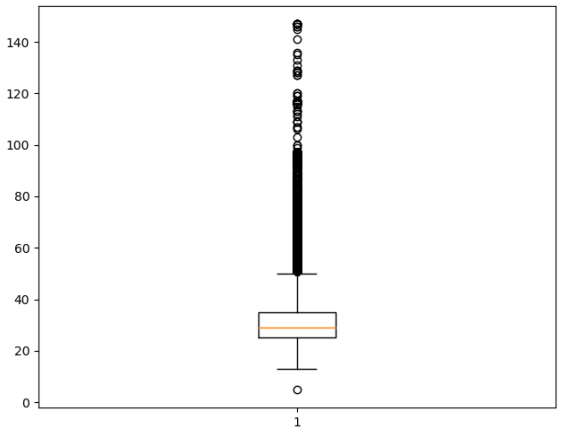

Figure 13 Boxplot of User Age ***[Source: Author]***

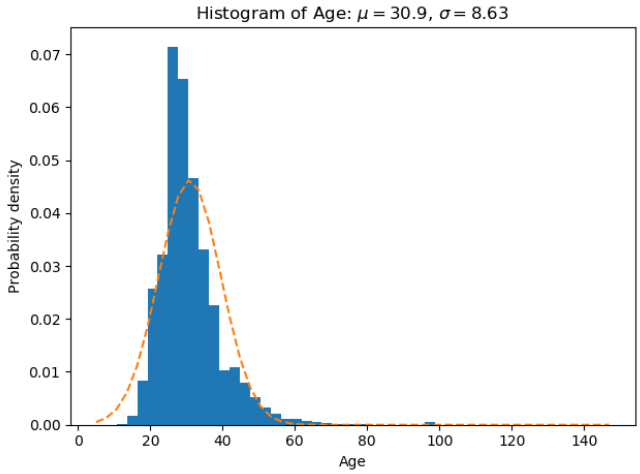

Figure 14 Histogram of User Age *[Source: Author]*

The filtered datasets were then loaded into the ODVD to identify more patterns through visualizations. As discussed in section 3.3 three measures are utilized to represent user contribution. Those are Average Reputation per Month, Average Questions per Month and Average Answers per Month.

As can be seen from Figure 15, there is a strong positive correlation (R^2 = 0.87) between age and average reputation per month. When age increases the reputation also increases. For an example a 40 years old user has slightly over 3 times higher reputation than a 25 year old user. A line can be fitted using linear regression for this as below.

$$\text{Average Reputation per Month} = 0.429(\text{Age}) - 5.1$$

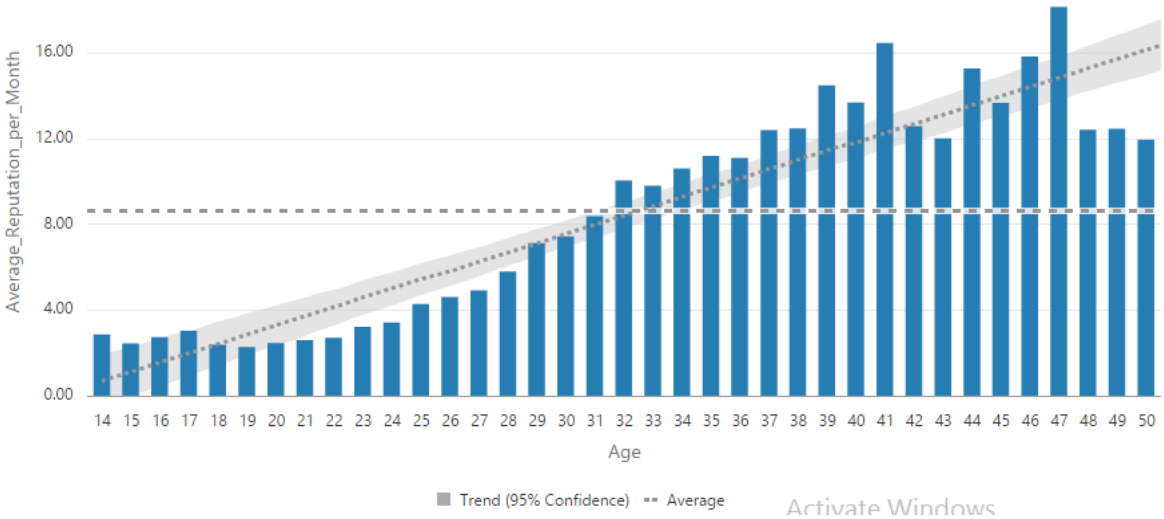

Figure 15 Average Reputation per Month by Age *[Source: Author]*

Further, the average number of answers posted per month also increases with the age as per Figure 16 with very strong positive correlation ($R^2 = 0.92$). For an example a 40 years old user generally posts 2 times higher than a 25 year old user. A line can be fitted using linear regression for this as below.

$$\text{Average Answers per Month} = 0.0087(\text{Age}) - 0.0483$$

Therefore this strongly supports the argument that the software development knowledge increases with the age. Users should be well equipped with knowledge to provide many answers. However currently there is no data processed in this study to get the answers which are marked as correct. But most of the questions in Stack Overflow has more than one answer since particularly in software development, one can solve the same problem in different ways.

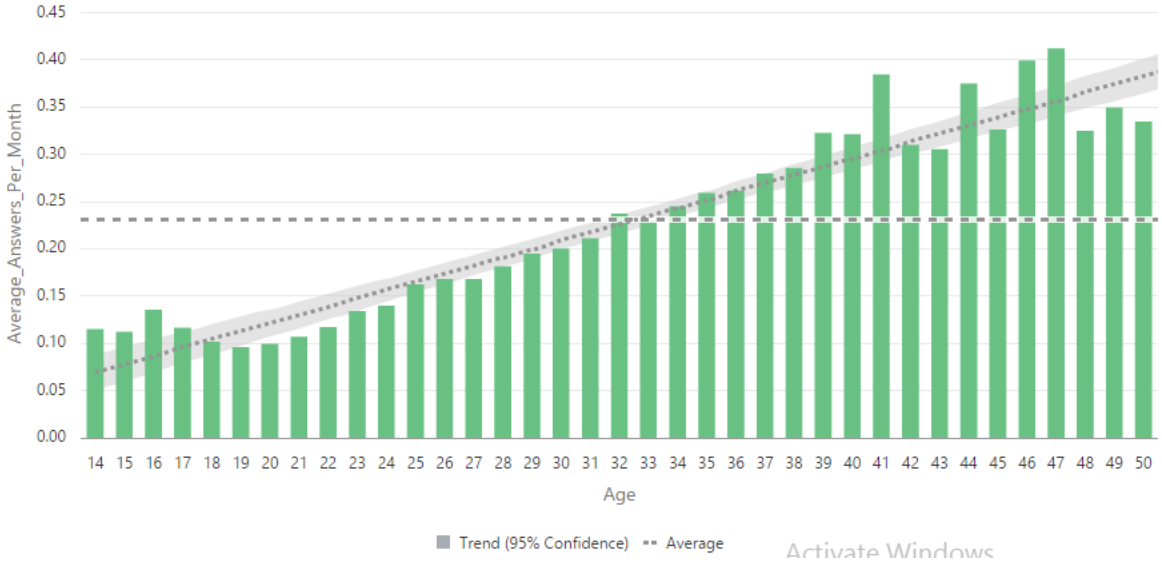

Figure 16 Average Answers per Month by Age *[Source: Author]*

However as shown in Figure 17, the average number of questions posted decreases when users are getting older. This is not decreasing by a large margin but it has medium to strong correlation ($R^2 = 0.72$). But the pattern is interesting as the people younger than 35 years have slightly higher tendency to ask things from the community than the older people. A line can be fitted using linear regression for this as below.

$$\text{Average Questions per Month} = -0.001(\text{Age}) + 0.101$$

This also shows that older people may have knowledge to perform their tasks without much outside support. One can also argue that older people do not like to ask something

due to their prestige and ego. But since this is an online community with no face to face contact, that argument lacks context.

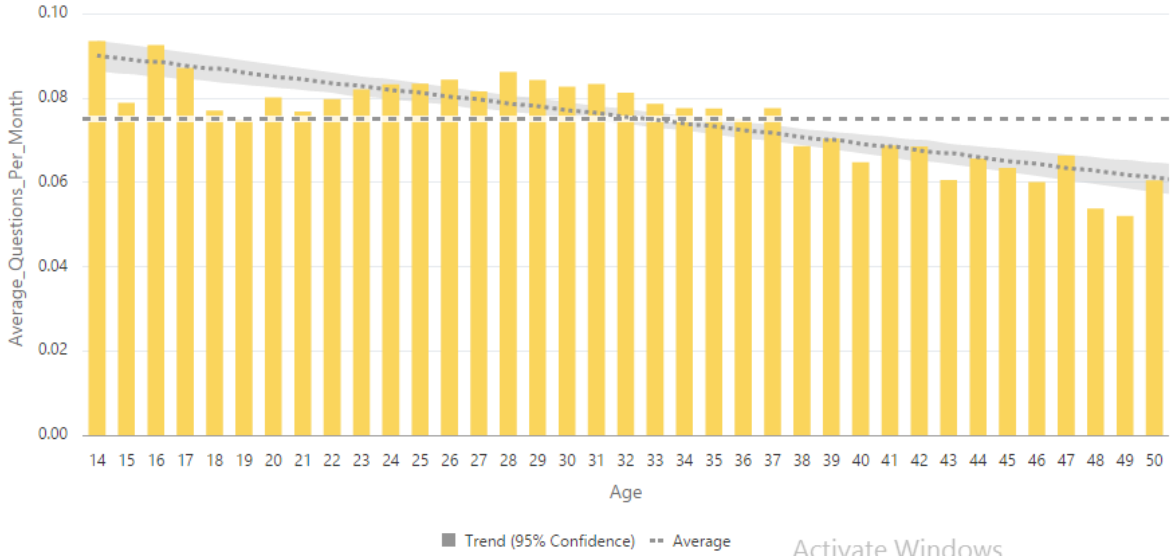

Figure 17 Average Questions per Month by Age *[Source: Author]*

Additionally, it is also observed that the reputation gain and answering rate start to exponentially increase when users pass 5 years of membership in Stack Overflow. Until 5 years, the reputation gain and answering speed for a user is quite low and not improving much. But the rate of asking is slightly improving with the time. The graphs for these observations can be found in Appendix IV.

5.3 User Clusters based on Contribution

The research question "Can we classify crowd into three groups: super contributors, contributors, and outliers?" is analysed in this section.

The counts of questions and answers posted by each user is derived from the posts data as also illustrated in Figure 10 in page 42. Basic statistics about the contribution of users are displayed in Table 11. It is observed that the variation of both of these variables are very high with large standard deviation values. Histograms were also studied and both were right skewed and concentrated zero since most of the users have not posted a single question or answer.

Table 11 Basic Statistics about User Q&A Counts *[Source: Author]*

	Mean	Standard Deviation	Minimum	Maximum	99th Percentile
Number of Questions	30.9	8.6	0	2274	30
Number of Answers	120	2306.4	0	40215	49

This scenario can be further realized from the pie chart in Figure 18. 54% of the users in Stack Overflow has not posted a single question or an answer. 14% of users are active contributors been posting both questions and answers. While 10% of users have only answered and 14% have asked questions. It can be also mentioned that while quarter of users (24%) provides answers, about three quarter (78%) of users just get benefits.

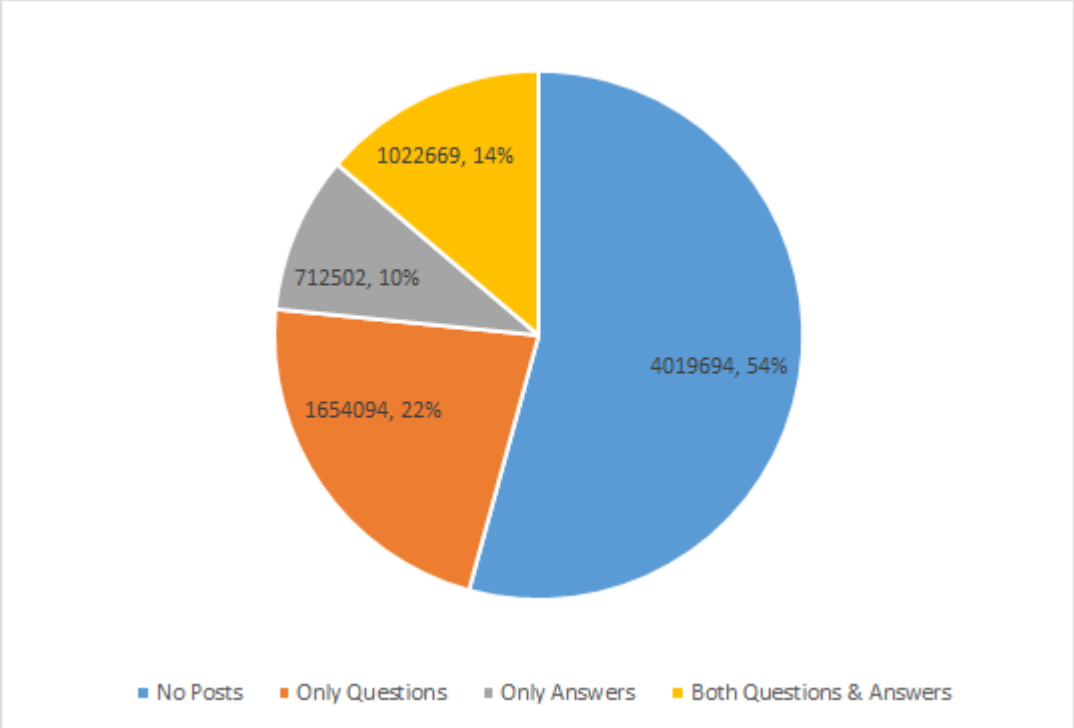

Figure 18 Q&A Count Patterns *[Source: Author]*

Then k-means clustering was used to further identify that it can be identifies three clusters similar to research done by Stewart et al. in an organizational crowdsourcing platform (Stewart, Lubensky, Huerta, 2010). However meaningful three clusters could not be found due to the effect from extreme values as displayed in Figure 19. Number of questions are in x axis and number of answers are in y axis.

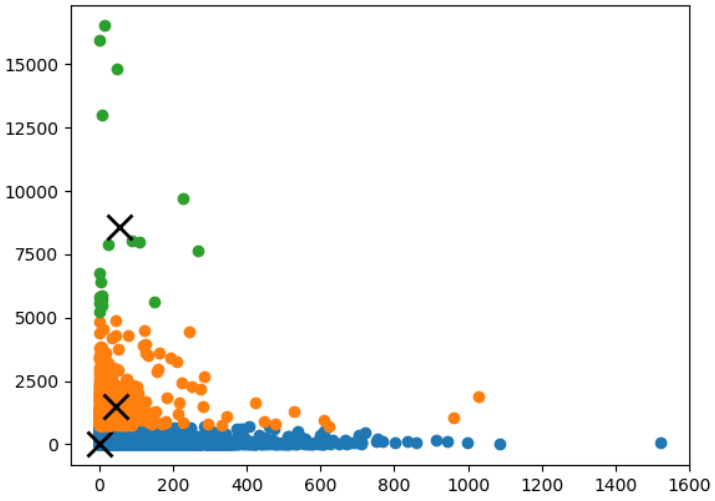

Figure 19 Clusters before Filtering *[Source: Author]*

Then the 99th percentile for both number of questions and answers were calculated and taken as the maximum boundary for the cluster analysis. Number of answers were filtered to be between 0 and 49 and number of questions were filtered to be between 0 and 30. Then the k-means algorithm could come up with the clusters shown in figure with this filtered dataset. Number of questions are in x axis and number of answers are in y axis.

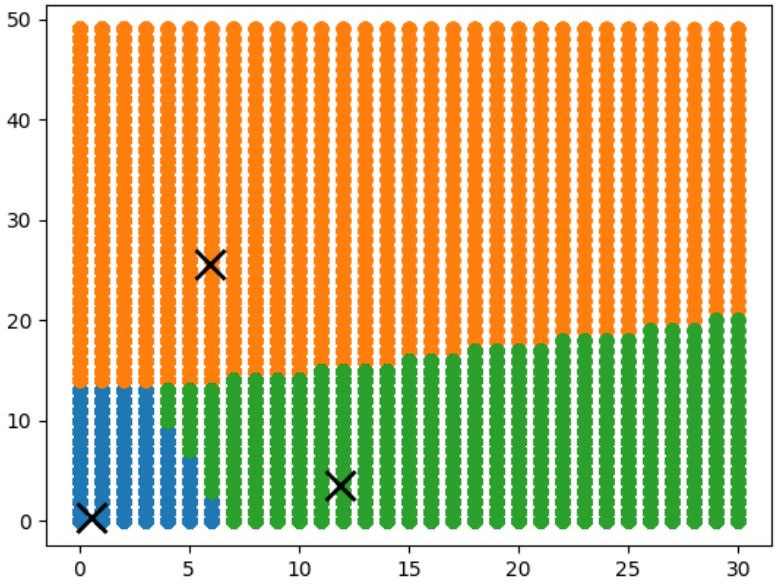

Figure 20 Clusters after Pruning Outliers *[Source: Author]*

Here it can be clearly identified that there are three clusters like, “One-timers”, “Question Askers” and “Answer Providers”. One-timers are the group with minimum level of contribution. Question Askers tend to ask more questions than answers while vice versa for Answer Providers. Table 12 summarizes details about these identified user groups. 93.8% of users are One-timers while another 4.5% are Question Askers. Only 1.7% of users are real knowledge sharers who are Answer Providers.

Table 12 Details of Identified Clusters *[Source: Author]*

Cluster	Centroid (Average)		Count	Percentage (%)
	Number of Questions	Number of Answers		
One-timers	0.6	0.4	6,827,425	93.8
Question Askers	11.9	3.6	329,854	4.5
Answer Providers	6.0	25.6	123,876	1.7

The presence of three clusters according to Stewart et al. could be verified in case of Stack Overflow (Stewart, Lubensky, Huerta, 2010) . But the actual participation percentages of each cluster goes together with what Nielsen observes in his research in 2006 related to in Social Media and Online Communities (Nielsen, 2006).

“In most online communities, 90% of users are lurkers who never contribute, 9% of users contribute a little, and 1% of users account for almost all the action.” (Nielsen, 2006)

Therefore this observation is still valid in 2017 with the largest crowdsourcing Q&A community related to software development.

5.4 Topics & Trends

The research question “What are popular topics and their trends in different categories such as Programming Languages, Frameworks and Databases the crowd interested in?” is analysed in this section.

When a user posts a question on the Stack Overflow website, he can provide number of tags (keywords) related to the topic area. Usually this contains the programming language, framework or DBMS the question is about. In this research, all the questions posted in Stack Overflow until 3rd of December 2017 were analysed. The tags and their

aggregated counts were derived annually for a 4 years period from 2014 to 2017. Then most popular tags were filtered and visualized according to the category (i.e. programming language, framework).

There are around 7.4 million registered users in Stack Overflow at the time of this analysis and about 22 million posts were taken into consideration in this research for the selected period. Therefore the findings should be significant to depict the general view of global software industry. Global amount usage of a technology may affect the number of questions asked about that. Hence this figure can be taken as a metric to evaluate the popularity of technologies. However the number of questions for a topic can be also depend on some other factors like the difficulty and complexity, lacking of help material etc. Therefore it is good that these results can be cross validated with some other popularity rankings as well.

The TIOBE Programming Community Index[24] and Stack Overflow Developer Survey[25] results can be used for cross validating the results. TIOBE Index ranking is mainly based on the hits of the most popular search engines for each technologies (TIOBE software BV, 2018b). Meanwhile Stack Overflow rankings are based on a survey done based on 64,000 software developers worldwide (Stack Exchange Inc, 2018f).

Figure 21 shows all-time favourite topics (topics with more than 50,000 tag references in the questions) as a word cloud. The size of the topic is scaled as per the popularity. The all-time popular topics among users include; JavaScript, Java, C# PHP, Android, jQuery, Python, HTML and C++ respectively

[24] https://www.tiobe.com/tiobe-index/
[25] https://insights.stackoverflow.com/survey/2017

Figure 21 Topic Cloud *[Source: Author]*

5.4.1 Programming Languages

The top 20 most popular programming languages in TIOBE Index for March 2018 were chosen for this analysis (TIOBE software BV, 2018a). Figure 22 shows the trends of these programming languages from 2014 to December, 2017 derived from this research. The main observation we can make is the steep rise of popularity of Python programming language. This can be attributed to the popularity of Data Science and Big Data application and Pythons use in these area. Python is also a popular as web development language. And we can observe a slight decline of popularity for JavaScript, Java, PHP, C# and C++.

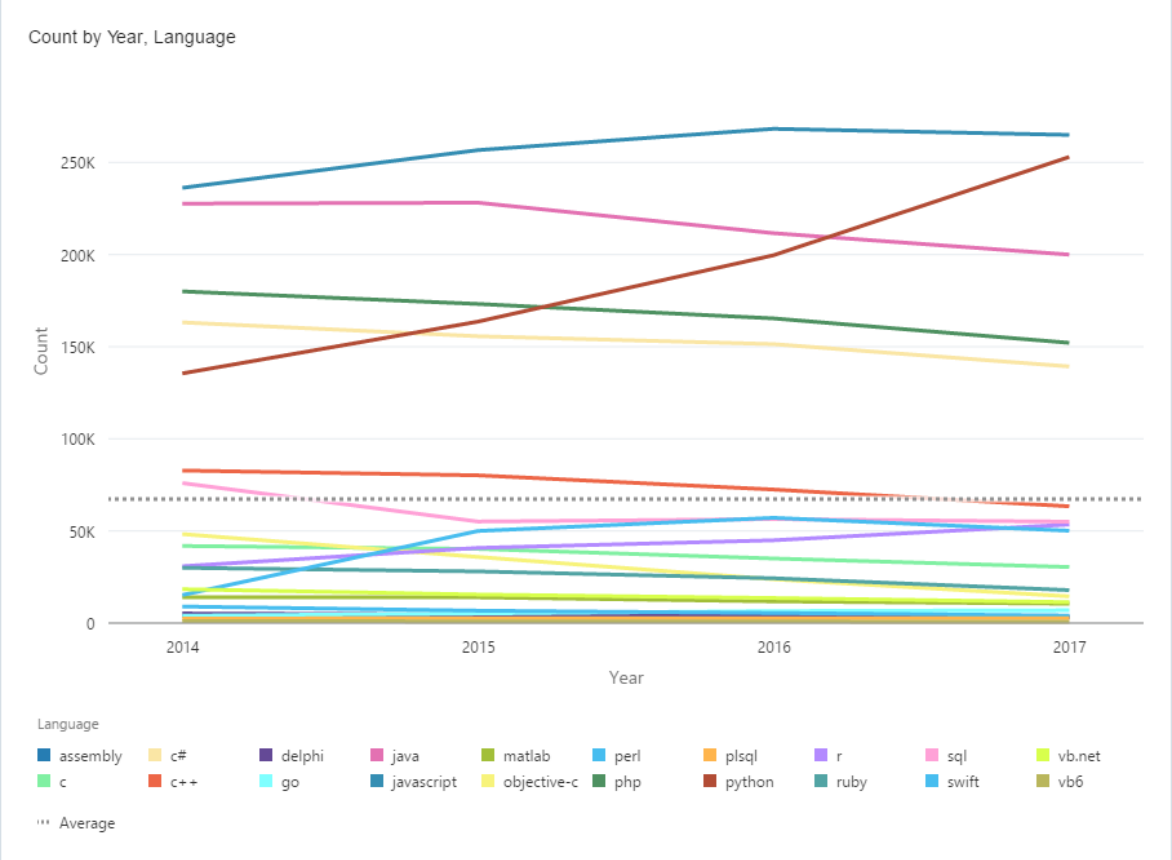

Figure 22 Language Trends *[Source: Author]*

Figure 23 displays the trends in TIOBE Index. The rise of popularity of Python and decreasing trends of other mentioned languages can be observed here as well. However in TIOBE Index JavaScript falls in 8th position. Therefore JavaScript making top in Stack Overflow has to be further analysed. One can suspect that the lack of official documentation and support can draw users to depend on the crowd of experts for help. Apart from that the other notable observation is the low appearance of C and Visual Basic .Net tags even though they are in top ten of TIOBE Index. This can be also due to the presence of good documentation for C (books) and Visual Basic .Net (MSDN).

However according to the Stack Overflow Developer Survey 2017, JavaScript is the most popular among developers. Then SQL, Java C# and Python follows up respectively. The focus towards JavaScript can be justified as 72.6% of the survey respondents were web developers while desktop application developers consists of 28.9% and 23% mobile developers (Stack Exchange Inc, 2018f). Therefore the bias towards JavaScript is mainly due to the fact that the Stack Overflow users are been mostly web developers.

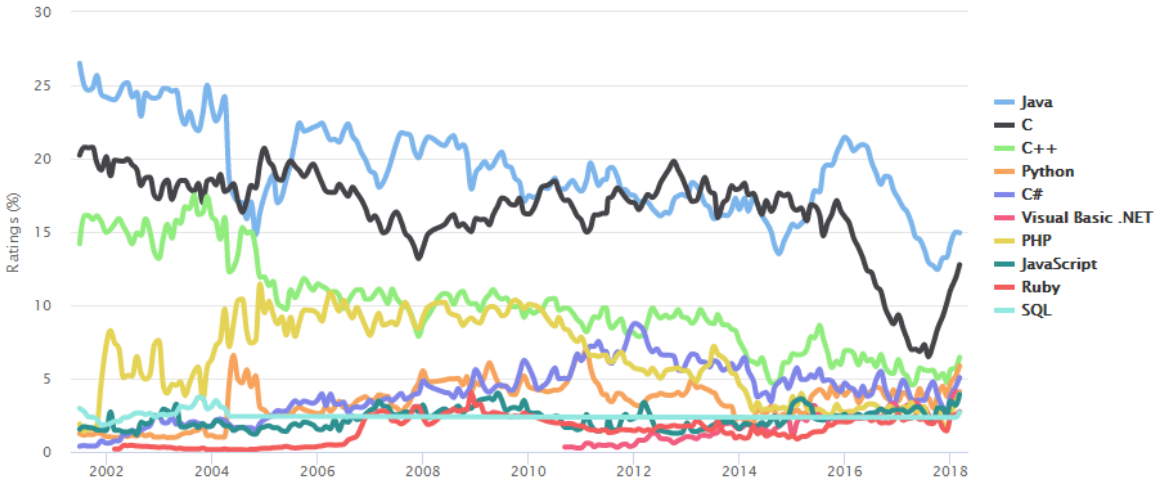

Figure 23 TIOBE Index for Programming Languages ***[Source: TIOBE software BV, 2018]***

The slight downward trends of some languages can be attributed to the saturation effect as well. When questions are already posted and answered, there is no need to put a new question. This can be tackled if the number of views per question also counted for the analysis. Another solution can be to visualise the running totals of total questions posted per each tag.

5.4.2 Frameworks

The top 9 frameworks listed in Stack Overflow Developer Survey 2017 were taken into consideration in this analysis. The Figure 24 shows the trends of the frameworks.

Users were mostly interested in JavaScript frameworks. JQuery seems to be the mostly questioned framework by a large margin. AngularJS follows JQuery, and then React. The cross-platform JavaScript runtime environment Node.js is also a popular topic for the community, while Ruby-on-Rails is the mostly discussed web application development framework. One must not forget the .Net Core environment as it is also in top 10.

However both JQuery and AngularJS has a steep downward trend. It seems React is gaining popularity in the community as a JavaScript framework by competing with the latter. Both Ruby-on-Rails and .Net Core also has a downward trend. But Node.js is becoming even popular with a consistent phase.

The big data platform Hadoop has a consistent popularity throughout the period. But Apache Spark since its inception has gained much popularity in this category even surpassing Hadoop in 2016.

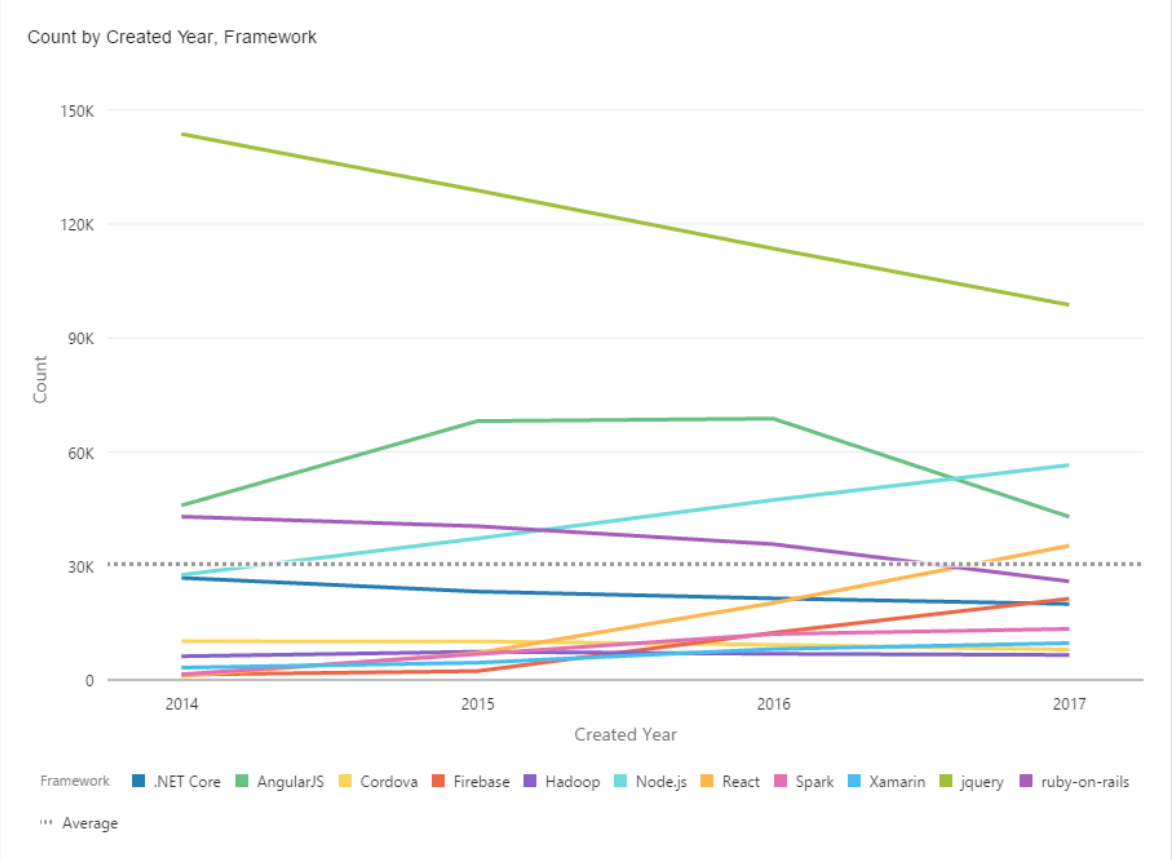

Figure 24 Framework Trends *[Source: Author]*

5.4.3 Databases

The top 8 databases listed in Stack Overflow Developer Survey 2017 were taken into consideration in this analysis. The Figure 25 shows the trends of the databases. Notable observations are the decline of popularity of MySQL and rise of MongoDB and PostgreSQL. However still top place belongs to MySQL by a large margin, which is then followed by SQL Server, MongoDB and PostgreSQL. The findings tallies with Stack Overflow Developer Survey 2017 results except for SQLLite which shares equal popularity with PostgreSQL in the developer survey. The downward trend of MySQL can be attributed to the saturation effect of questions. The rise of the NoSQL database MongoDB must be due to its popularity with the big data applications and web development.

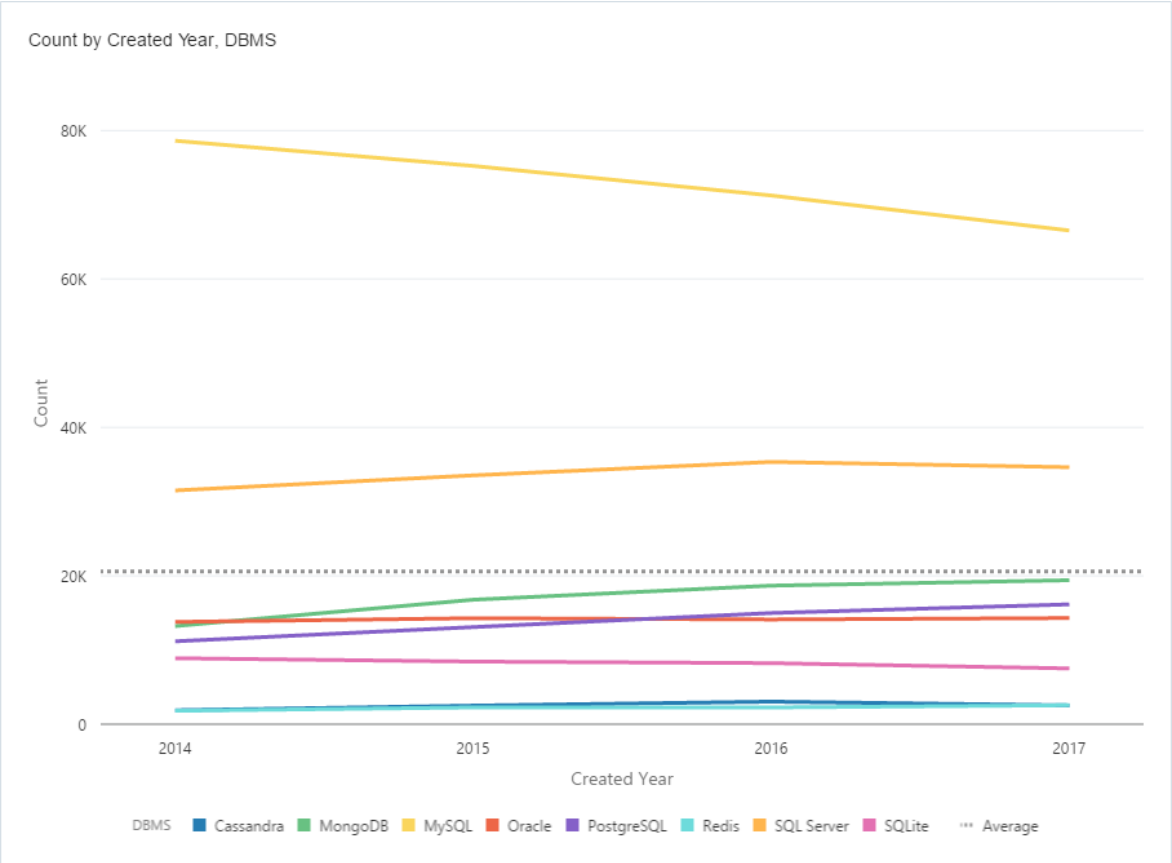

Figure 25 DBMS Trends *[Source: Author]*

6 Conclusion

Crowdsourcing has become a popular global phenomena with invent of Internet technologies, ICT infrastructure and accessibility. Therefore lot of software professionals also use Crowdsourcing platforms for various activities related to software engineering. The Q&A website Stack Overflow in Stack Exchange Network is one among those websites which is daily used by millions of users. Therefore Stack Overflow data could reveal important patterns in global crowdsourcing beneficial for software industry. The aim of this study was to perform data mining on Stack Exchange data, to discover some of these patterns. Main focus of this research was on following four areas.

1. Global user distribution and contribution
2. Contribution related to user age
3. Classify users with regard to their involvement
4. Identify popular topics with trends

All the phases namely selection, pre-processing, transformation, data mining, interpretation and evaluation were carried out in the process of pattern discovery for this study.

Big data analytic techniques were used for data mining activities using Apache Spark with Python language. The xml structure of the Stack Exchange data dump is not compatible with spark-xml library to process directly. Therefore workaround solution has to be used to load xml data into a relational database system as an intermediary persistence. Even though the XML must have been in the correct format, due to the computer memory restrictions the large xml files cannot be processed in-memory by Spark. This is mainly due to the fact that XML is not a breakable file format. Apart from this, Apache Spark along with Python and its libraries has provided a simple yet powerful framework for data analysis tasks carried out in this research.

One limitation of this research has been the usage of single computer (4GB RAM) with Apache Spark for data analytics. It could have been beneficial if Apache Spark has been used in cluster mode which can be more powerful with the help of parallel processing and more computing memory.

The user friendliness and features of ODVD were very useful for interpreting results and identifying patterns. ODVD's advanced data analytics features such as automatic clustering of data, identifying trend lines were also utilized. However for the advanced

cluster analysis, scikit-learn Python library has been used as it provides more features and controls.

The results on Global User Distribution and Contribution, clearly show that although majority of the users are from USA and India. However in both participation and contribution aspects, European countries along with Australia, Canada and New Zealand has higher rankings. It is also noted the less rankings of Japan, South Korea, Russian Federation, Brazil and China. Since these countries represent huge portion of world population, further studies should be carried out to find factors for this phenomena.

There is a common perception in society that younger people are have higher knowledge in contemporary or modern technologies. However the results from this study challenges this view as it could be revealed that younger people ask more questions than older people, while vice versa for answers. Also the reputation metric in Stack Overflow also increases with age. This is an important finding for software industry. But however this should be further verified by additional data. Most of the aged users in Stack Overflow can be technology enthusiastic personnel. So these users may sometimes do not represent the general picture in the software industry.

Further, users could be classified as “one-timers” (93.8%), “question askers” (4.5%) and “answer providers” (1.7%). This proves that almost all of the users of a crowdsourcing platform are just observers while only few people actively contributes. This finding will very useful for Stack Overflow itself to rethink about their strategy to motivate users to contribute. However so far Stack Overflow is the most popular and respected community on its kind. But one could also note that the reputation score in Stack Overflow does not solely depend on questions and answers posted but various other activities like voting, commenting, reviewing etc. That can be a reason for success of Stack Overflow.

Finally, popularity and trends of different programming languages, databases and frameworks are also identified with the help of tag counts. These results can be biased based on the developer profiles of Stack Overflow users. For an example if Stack Overflow has more web developers than desktop developers, the community may ask many questions related to web technologies. So a future research can be carried out to find the developer profile distribution of Stack Overflow users.

7 References

ABDALKAREEM, Rabe, SHIHAB, Emad and RILLING, Juergen, 2017. What Do Developers Use the Crowd For? A Study Using Stack Overflow. *IEEE Software*. 2017. Vol. 34, no. 2, p. 53–60. DOI 10.1109/MS.2017.31.

AHMED, Tanveer and SRIVASTAVA, Abhishek, 2017. Understanding and evaluating the behavior of technical users. A study of developer interaction at StackOverflow. *Human-centric Computing and Information Sciences*. 2017. Vol. 7, no. 1, p. 1–19. DOI 10.1186/s13673-017-0091-8.

ALLAMANIS, Miltiadis and SUTTON, Charles, 2013. Why, when, and what: Analyzing stack overflow questions by topic, type, and code. *IEEE International Working Conference on Mining Software Repositories*. 2013. No. Table I, p. 53–56. DOI 10.1109/MSR.2013.6624004.

ANDERSON, Ashton, HUTTENLOCHER, Daniel, KLEINBERG, Jon and LESKOVEC, Jure, 2012. Discovering value from community activity on focused question answering sites. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12* [online]. 2012. P. 850. DOI 10.1145/2339530.2339665. Available from: http://dl.acm.org/citation.cfm?doid=2339530.2339665

AROLAS, Enrique Estellés and LADRÓN-DE-GUEVARA, Fernando González, 2012. Towards an Integrated Crowdsourcing Definition. *Journal of Information Science*. 2012. Vol. 32, no. 2, p. 1–16.

ATWOOD, Jeff, 2009. A Theory of Moderation - Stack Overflow Blog. [online]. 2009. [Accessed 15 March 2018]. Available from: https://stackoverflow.blog/2009/05/18/a-theory-of-moderation/

BARUA, Anton, THOMAS, Stephen W. and HASSAN, Ahmed E., 2014. *What are developers talking about? An analysis of topics and trends in Stack Overflow*. ISBN 1066401292.

BAZELLI, Blerina, HINDLE, Abram and STROULIA, Eleni, 2013. On the personality traits of StackOverflow users. *IEEE International Conference on Software Maintenance,*

ICSM. 2013. P. 460–463. DOI 10.1109/ICSM.2013.72.

BERGER, Philipp, HENNIG, Patrick, BOCKLISCH, Tom, HEROLD, Tom and MEINEL, Christoph, 2017. A Journey of Bounty Hunters: Analyzing the Influence of Reward Systems on StackOverflow Question Response Times. *Proceedings - 2016 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2016*. 2017. P. 644–649. DOI 10.1109/WI.2016.0114.

BHAT, Vasudev, 2014. Min (e) d Your Tags : Analysis of Question Response Time in StackOverflow. . 2014. No. Asonam, p. 328–335.

BHAT, Vasudev, GOKHALE, Adheesh, JADHAV, Ravi, PUDIPEDDI, Jagat and AKOGLU, Leman, 2015. Effects of tag usage on question response time: Analysis and prediction in StackOverflow. *Social Network Analysis and Mining*. 2015. Vol. 5, no. 1, p. 1–13. DOI 10.1007/s13278-015-0263-3.

BOSU, Amiangshu, CORLEY, Christopher S., HEATON, Dustin, CHATTERJI, Debarshi, CARVER, Jeffrey C. and KRAFT, Nicholas A., 2013. Building reputation in StackOverflow: An empirical investigation. *IEEE International Working Conference on Mining Software Repositories*. 2013. P. 89–92. DOI 10.1109/MSR.2013.6624013.

CALEFATO, Fabio, LANUBILE, Filippo, MARASCIULO, Maria Concetta and NOVIELLI, Nicole, 2015. Mining successful answers in stack overflow. *IEEE International Working Conference on Mining Software Repositories*. 2015. Vol. 2015–Augus, p. 430–433. DOI 10.1109/MSR.2015.56.

FAYYAD, Usama, PIATETSKY-SHAPIRO, Gregory and SMYTH, Padhraic, 1996. From Data Mining to Knowledge Discovery in Databases. *AI Magazine*. 1996. Vol. 17, p. 37–54.

FRIENDLY, Michael, 2009. *Milestones in the history of thematic cartography, statistical graphics, and data visualization*. 2009.

GANTAYAT, Neelamadhav, DHOOLIA, Pankaj, PADHYE, Rohan, MANI, Senthil and SINHA, Vibha Singhal, 2015. The synergy between voting and acceptance of answers on StackOverflow - Or the lack thereof. *IEEE International Working Conference on Mining Software Repositories*. 2015. Vol. 2015–Augus, p. 406–409. DOI 10.1109/MSR.2015.50.

GRECO, Marco and GRIMALDI, Michele, 2016. A formal definition of Big Data based on its essential features. . 2016. DOI 10.1108/LR-06-2015-0061.

GUALTIERI, Mike, KISKER, Holger, HAMMOND, Jeffrey S., CURRAN, Rowan, FICHERA, Richard, STATEN, James and CHRISTAKIS, Sophia, 2015. *Apache Spark Is Powerful And Promising* [online]. [Accessed 18 March 2018]. Available from: https://www.forrester.com/report/Apache+Spark+Is+Powerful+And+Promising/-/E-RES121127#

GUALTIERI, Mike, SRIDHARAN, Srividya, KISKER, Holger and AUSTIN, Christian, 2017. *The Forrester Wave™: Predictive Analytics And Machine Learning Solutions, Q1 2017* [online]. [Accessed 18 March 2018]. Available from: https://www.forrester.com/report/The+Forrester+Wave+Predictive+Analytics+And+Machine+Learning+Solutions+Q1+2017/-/E-RES129452

GUAZZINI, Andrea, VILONE, Daniele, DONATI, Camillo, NARDI, Annalisa and LEVNAJIĆ, Zoran, 2015. Modeling crowdsourcing as collective problem solving. *Scientific Reports* [online]. 2015. Vol. 5, p. 1–10. DOI 10.1038/srep16557. Available from: http://dx.doi.org/10.1038/srep16557

HOWE, Jeff, 2006. The Rise of Crowdsourcing. *Wired Magazine* [online]. June 2006. Vol. 14, no. 6, p. 1–4. DOI 10.1086/599595. Available from: http://www.wired.com/wired/archive/14.06/crowds_pr.html

IMD WORLD COMPETITIVENESS CENTRE, 2017. *IMD World Digital Competitiveness Ranking 2017* [online]. Available from: https://www.imd.org/globalassets/wcc/docs/release-2017/world_digital_competitiveness_yearbook_2017.pdf

INDHU, R and PORKODI, R, 2018. Comparison of Clustering Algorithm. . 2018. Vol. 3, no. 1, p. 218–223.

JOORABCHI, Arash, ENGLISH, Michael and MAHDI, Abdulhussain E., 2016. Text mining stackoverflow: an insight into challenges and subject-related difficulties faced by computer science learners. *Journal of Enterprise Information Management* [online]. 2016. Vol. 29, no. 2, p. 255–275. DOI 10.1108/JEIM-11-2014-0109. Available from: http://www.emeraldinsight.com/doi/10.1108/JEIM-11-2014-0109

LIN, Bin and SEREBRENIK, Alexander, 2016. Recognizing gender of stack overflow users. *Proceedings of the 13th International Workshop on Mining Software Repositories - MSR '16* [online]. 2016. P. 425–429. DOI 10.1145/2901739.2901777. Available from: http://dl.acm.org/citation.cfm?doid=2901739.2901777

LOHR, Steve, 2013. The Origins of "Big Data": An Etymological Detective Story - The New York Times. [online]. 2013. [Accessed 18 March 2018]. Available from: https://bits.blogs.nytimes.com/2013/02/01/the-origins-of-big-data-an-etymological-detective-story/

LYNCH, Alec, 2010. Crowdsourcing is Not New - The History of Crowdsourcing (1714 to 2010). [online]. 2010. [Accessed 11 March 2018]. Available from: https://blog.designcrowd.com/article/202/crowdsourcing-is-not-new--the-history-of-crowdsourcing-1714-to-2010

MAO, Ke, CAPRA, Licia, HARMAN, Mark and JIA, Yue, 2017. A survey of the use of crowdsourcing in software engineering. *Journal of Systems and Software*. 2017. Vol. 126, p. 57–84. DOI 10.1016/j.jss.2016.09.015.

MENG, Xiangrui, BRADLEY, Joseph, YAVUZ, Burak, SPARKS, Evan, VENKATARAMAN, Shivaram, LIU, Davies, FREEMAN, Jeremy, TSAI, DB, AMDE, Manish, OWEN, Sean, XIN, Doris, XIN, Reynold, FRANKLIN, Michael J., ZADEH, Reza, ZAHARIA, Matei and TALWALKAR, Ameet, 2015. MLlib: Machine Learning in Apache Spark. [online]. 2015. Vol. 17, p. 1–7. DOI 10.1145/2882903.2912565. Available from: http://arxiv.org/abs/1505.06807

MERRIAM-WEBSTER, 2018. Crowdsourcing | Definition of Crowdsourcing by Merriam-Webster. [online]. 2018. [Accessed 11 March 2018]. Available from: https://www.merriam-webster.com/dictionary/crowdsourcing

MORRISON, Patrick and MURPHY-HILL, E, 2013. Is Programming Knowledge Related To Age? *People.Engr.Ncsu.Edu* [online]. 2013. P. 3–6. Available from: http://people.engr.ncsu.edu/ermurph3/papers/msr13.pdf

MOVSHOVITZ-ATTIAS, Dana, MOVSHOVITZ-ATTIAS, Yair, STEENKISTE, Peter and FALOUTSOS, Christos, 2013. Analysis of the reputation system and user contributions on a question answering website. *Proceedings of the 2013 IEEE/ACM*

International Conference on Advances in Social Networks Analysis and Mining - ASONAM '13 [online]. 2013. P. 886–893. DOI 10.1145/2492517.2500242. Available from: http://dl.acm.org/citation.cfm?doid=2492517.2500242

NIELSEN, Jakob, 2006. Participation Inequality: The 90-9-1 Rule for Social Features. [online]. 2006. [Accessed 31 March 2018]. Available from: https://www.nngroup.com/articles/participation-inequality/

RIEDL, Christoph, LEIMEISTER, Jan Marco and KASSEL, Universität, 2010. RATING SCALES FOR COLLECTIVE INTELLIGENCE IN INNOVATION COMMUNITIES : WHY QUICK AND EASY. In: *Thirty First International Conference on Information Systems*. 2010. p. 1–21.

RUBIN, Alexander, 2016. How Apache Spark makes your slow MySQL queries 10x faster (or more) - Percona Database Performance Blog. [online]. 2016. [Accessed 20 March 2018]. Available from: https://www.percona.com/blog/2016/08/17/apache-spark-makes-slow-mysql-queries-10x-faster/

SCHENK, Dennis and LUNGU, Mircea, 2013. Geo-Locating the Knowledge Transfer in Stack Overflow. In: *Proceedings of the 2013 International Workshop on Social Software Engineering*. Saint Petersburg, Russia: ACM. 2013. p. 2–5.

SHAH, Chintan and JIVANI, Anjali, 2013. Comparison of data mining clustering algorithms. *2013 Nirma University International Conference on Engineering (NUiCONE)* [online]. 2013. P. 1–4. DOI 10.1109/NUiCONE.2013.6780101. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6780101

SLAG, Rogier, DE WAARD, Mike and BACCHELLI, Alberto, 2015. One-day flies on StackOverflow - Why the vast majority of StackOverflow users only posts once. *IEEE International Working Conference on Mining Software Repositories*. 2015. Vol. 2015–Augus, p. 458–461. DOI 10.1109/MSR.2015.63.

STACK EXCHANGE INC, 2018a. About - Stack Exchange. [online]. 2018. [Accessed 15 March 2018]. Available from: https://stackexchange.com/about

STACK EXCHANGE INC, 2018b. About - Stack Overflow. [online]. 2018. [Accessed 15 March 2018]. Available from: https://stackoverflow.com/company

STACK EXCHANGE INC, 2018c. What is reputation? How do I earn (and lose) it? - Help Center - Stack Overflow. [online]. 2018. [Accessed 15 March 2018]. Available from: https://stackoverflow.com/help/whats-reputation

STACK EXCHANGE INC, 2018d. Badges - Stack Overflow. [online]. 2018. [Accessed 15 March 2018]. Available from: https://stackoverflow.com/help/badges

STACK EXCHANGE INC, 2018e. Stack Exchange Data Dump : Stack Exchange, Inc. : Free Download & Streaming : Internet Archive. [online]. 2018. [Accessed 12 January 2018]. Available from: https://archive.org/details/stackexchange

STACK EXCHANGE INC, 2018f. Stack Overflow Developer Survey 2017. [online]. 2018. [Accessed 31 March 2018]. Available from: https://insights.stackoverflow.com/survey/2017

STEWART, Osamuyimen, LUBENSKY, David and HUERTA, Juan M., 2010. Crowdsourcing participation inequality. *Proceedings of the ACM SIGKDD Workshop on Human Computation - HCOMP '10* [online]. 2010. P. 30. DOI 10.1145/1837885.1837895. Available from: http://portal.acm.org/citation.cfm?doid=1837885.1837895

TIOBE SOFTWARE BV, 2018a. TIOBE Index | TIOBE - The Software Quality Company. [online]. 2018. [Accessed 18 March 2018]. Available from: https://www.tiobe.com/tiobe-index/

TIOBE SOFTWARE BV, 2018b. Programming Languages Definition | TIOBE - The Software Quality Company. [online]. 2018. [Accessed 31 March 2018]. Available from: https://www.tiobe.com/tiobe-index/programming-languages-definition/

UNITED NATIONS, DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS, Population Division, 2017. *World Population Prospects: The 2017 Revision*.

UNITED NATIONS INTERNATIONAL TELECOMMUNICATION UNION, 2017. ITU | 2017 Global ICT Development Index. [online]. 2017. [Accessed 31 March 2018]. Available from: http://www.itu.int/net4/ITU-D/idi/2017/#idi2017rank-tab

UPADHYAY, Utkarsh, VALERA, Isabel and GOMEZ-RODRIGUEZ, Manuel, 2016. Uncovering the Dynamics of Crowdlearning and the Value of Knowledge. [online]. 2016. No. i. DOI 10.1145/3018661.3018685. Available from: http://arxiv.org/abs/1612.04831%0Ahttp://dx.doi.org/10.1145/3018661.3018685

VASILESCU, Bogdan, CAPILUPPI, Andrea and SEREBRENIK, Alexander, 2014. Gender, representation and online participation: A quantitative study. *Interacting with Computers*. 2014. Vol. 26, no. 5, p. 488–511. DOI 10.1093/iwc/iwt047.

VASILESCU, Bogdan, FILKOV, Vladimir and SEREBRENIK, Alexander, 2013. StackOverflow and GitHub: Associations between software development and crowdsourced knowledge. *Proceedings - SocialCom/PASSAT/BigData/EconCom/BioMedCom 2013*. 2013. P. 188–195. DOI 10.1109/SocialCom.2013.35.

WARD, Jonathan Stuart and BARKER, Adam, 2013. Undefined By Data: A Survey of Big Data Definitions. [online]. 2013. DOI 10.1145/2699414. Available from: http://arxiv.org/abs/1309.5821

ZAHARIA, Matei, CHOWDHURY, Mosharaf, DAS, Tathagata and DAVE, Ankur, 2012. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. *Nsdi* [online]. 2012. P. 2–2. DOI 10.1111/j.1095-8649.2005.00662.x. Available from: https://www.usenix.org/system/files/conference/nsdi12/nsdi12-final138.pdf

ZAHARIA, Matei, FRANKLIN, Michael J., GHODSI, Ali, GONZALEZ, Joseph, SHENKER, Scott, STOICA, Ion, XIN, Reynold S., WENDELL, Patrick, DAS, Tathagata, ARMBRUST, Michael, DAVE, Ankur, MENG, Xiangrui, ROSEN, Josh and VENKATARAMAN, Shivaram, 2016. Apache Spark: a unified engine for big data processing. *Communications of the ACM*. 2016. Vol. 59, no. 11, p. 56–65. DOI 10.1145/2934664.

ZAHARIA, Matei, KARAU, Holden, KONWINSKI, Andy and WENDELL, Patrick, 2015. *Learning Spark: Lightning-Fast Big Data Analytics*. 1st. O'Reilly Media, Inc. ISBN 1449358624 9781449358624.

ZHAO, Yuxiang and ZHU, Qinghua, 2014. Evaluation on crowdsourcing research: Current status and future direction. *Information Systems Frontiers*. 2014. Vol. 16, no. 3, p. 417–434. DOI 10.1007/s10796-012-9350-4.

8 Appendices

Appendix I

This section includes the Python script which is been used to load data from Users.xml file to a table named “users” in MySQL database.

```
from pyspark import SparkContext
from pyspark.sql import SQLContext
from pyspark.sql.types import *

sc = SparkContext('local', 'thesis')
sqlContext = SQLContext(sc)

header = "<?xml version=\"1.0\" encoding=\"utf-8\"?> <users>"
footer = '</users>'
counter = 0
lines = ""

row_type = StructType([ \
        StructField("_AccountId", LongType(), True), \
            StructField("_Age", StringType(), True), \
        StructField("_CreationDate", StringType(), True), \
            StructField("_DisplayName", StringType(), True), \
        StructField("_DownVotes", LongType(), True), \
            StructField("_Id", LongType(), True), \
            StructField("_LastAccessDate", StringType(), True), \
            StructField("_Location", StringType(), True), \
        StructField("_ProfileImageUrl", StringType(), True), \
            StructField("_Reputation", LongType(), True), \
            StructField("_UpVotes", LongType(), True), \
        StructField("_VALUE", StringType(), True), \
        StructField("_Views", LongType(), True), \
        StructField("_WebsiteUrl", StringType(), True)])

my_schema = StructType([ StructField("row", ArrayType(row_type, True),
True)])

log = open("C:\\users_log.txt","w")

with open("D:\\Himesha\\MSc_Informatics\\Thesis\\StackExchange\\08-12-
2017\\Users.xml") as f:
      for line in f:
            counter = counter + 1
            if "<row Id=" in line:
                  lines = lines + line
            if (counter % 1000 == 0):
                  row = header + lines + footer
                  fh = open("C:\\users.xml","w")
                  fh.write(row)
                  fh.close()
                  try:
                        df =
sqlContext.read.format('com.databricks.spark.xml').options(rowTag='users').
schema(my_schema).load('C:\\users.xml')
```

```
                        dest_df = df.selectExpr("explode(row) as
e").select("e.*")
                        # dest_df.show()
                        dest_df.write.format('jdbc').options(

url='jdbc:mysql://localhost:3306/StackExchange',
                                        driver='com.mysql.jdbc.Driver',
                                        dbtable='users',
                                        user='root',
                                        password='root').mode('append').save()
                    except:
                        log.write("Error processing lines in batch contains
line " + str(counter) + "\n")
                        lines = ''
                        continue;
                    lines = ''

if (counter % 1000 > 0):
      row = header + lines + footer
      fh = open("C:\\users.xml","w")
      fh.write(row)
      fh.close()
      try:
            df =
sqlContext.read.format('com.databricks.spark.xml').options(rowTag='users').
schema(my_schema).load('C:\\users.xml')
            dest_df = df.selectExpr("explode(row) as e").select("e.*")
            dest_df.write.format('jdbc').options(
                        url='jdbc:mysql://localhost:3306/StackExchange',
                        driver='com.mysql.jdbc.Driver',
                        dbtable='users',
                        user='root',
                        password='root').mode('append').save()
      except:
            log.write("Error processing lines in batch contains line " +
str(counter) + "\n")

log.close()
```

This script was executed using spark-submit script which is located in Spark's bin directory as below.

```
spark-submit -packages com.databricks:spark-xml_2.11:0.4.1
D:\Himesha\MSc_Informatics\Thesis\SparkTestWork\Final\users.py
```

Appendix II

This section includes the Python script which is been used to load data from a table named "users" in MySQL database, aggregate country wise user data and save results into a single CSV file.

```
from pyspark import SparkContext
sc = SparkContext('local', 'thesis')
from pyspark.sql import SQLContext
sqlContext = SQLContext(sc)
```

```
source_df = sqlContext.read.format('jdbc').options(
          url='jdbc:mysql://localhost:3306/stackexchange',
          driver='com.mysql.jdbc.Driver',
          dbtable='user_countries',
          user='root',
              partitionColumn = 'Id',
              lowerBound = 0,
              upperBound = 1200000,
              numPartitions = 6,
          password='root').load()

# Looks the schema of this DataFrame.
source_df.printSchema()

# Groups people by country and calculate aggregates
countsByCountry = source_df.groupBy("Country").agg({"*": "count",
"Reputation": "avg", "Age": "avg"})

countsByCountry.show()

# Saves a single merged csv file
countsByCountry.repartition(1).write.format("com.databricks.spark.csv").opt
ion("header", "true").save("c:\\UserCountry.csv")
```

Appendix III

This section includes the Python script which is been used to identify the clusters using K-Means clustering method in the scikit-learn library.

```
from pyspark import SparkContext
sc = SparkContext('local', 'thesis')

from pyspark.sql import SQLContext
sqlContext = SQLContext(sc)

source_df = sqlContext.read.format("csv").option("header",
"true").option("mode",
"DROPMALFORMED").load("D:\\Himesha\\MSc_Informatics\\Thesis\\StackExchange\
\08-12-2017\\QACount_with_all_users.csv")


filtered_ds = source_df.filter(source_df.Questions <
31).filter(source_df.Answers < 50)

centers = filtered_ds.rdd.map(lambda p: [int(p.Questions),
int(p.Answers)]).collect()

import numpy as np

from sklearn.cluster import KMeans
from sklearn import metrics

import matplotlib.pyplot as plt

data = np.asarray(centers, dtype = np.float32)

k = 3
```

```
kmeans = KMeans(n_clusters=k, random_state=0).fit(data)

labels = kmeans.labels_
centroids = kmeans.cluster_centers_

from matplotlib import pyplot
import numpy as np

for i in range(k):
    # select only data observations with cluster label == i
      ds = data[np.where(labels==i)]
      print("\nCluster " + str(centroids[i,0]) + ", " + str(centroids[i,1])
+ " count: " + str(len(ds)))
    # plot the data observations
      pyplot.plot(ds[:,0],ds[:,1],'o')
    # plot the centroids
      lines = pyplot.plot(centroids[i,0],centroids[i,1],'kx')
    # make the centroid x's bigger
      pyplot.setp(lines,ms=15.0)
      pyplot.setp(lines,mew=2.0)
pyplot.show()
```

Appendix IV

This section includes a set of supplementary figures for further details.

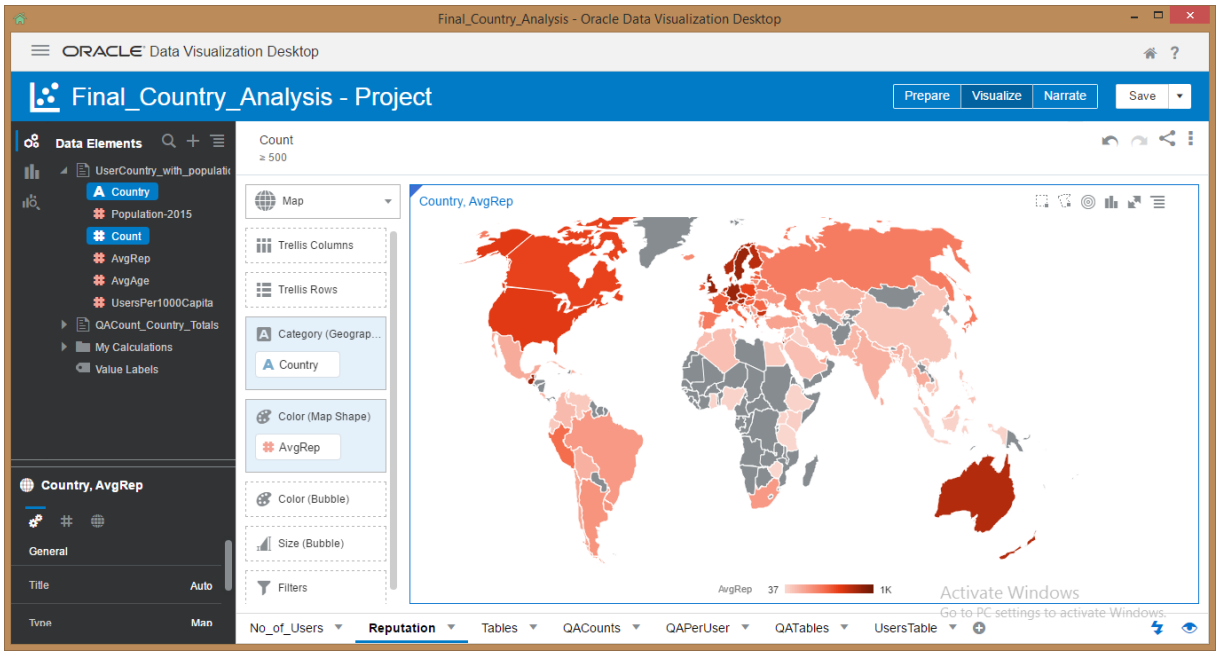

Figure 26 A Screenshot of ODVD Software *[Source: Author]*

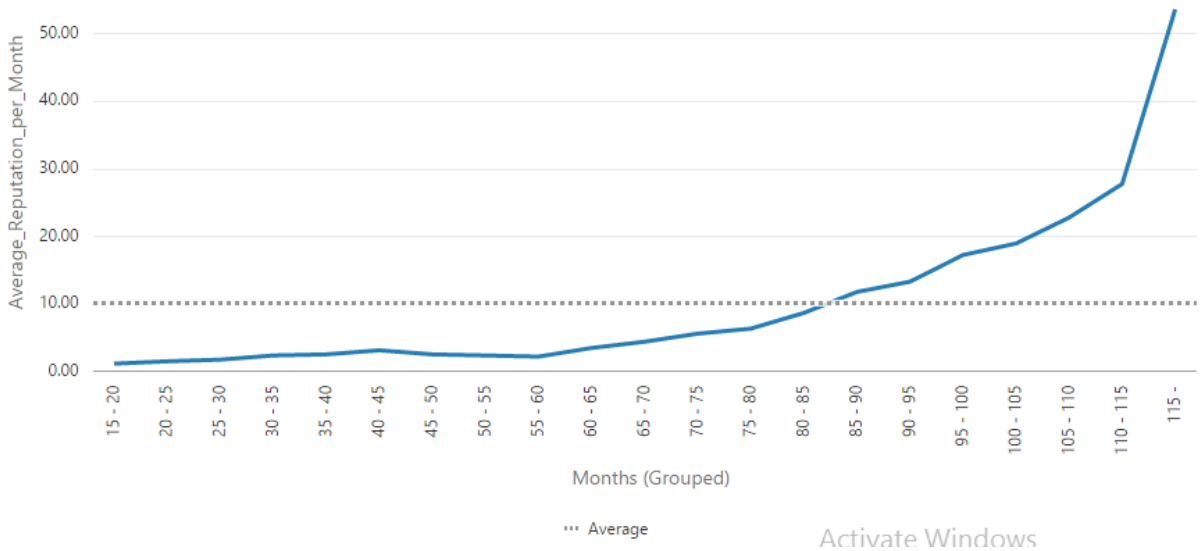

Figure 27 Reputation Gain per Membership Time *[Source: Author]*

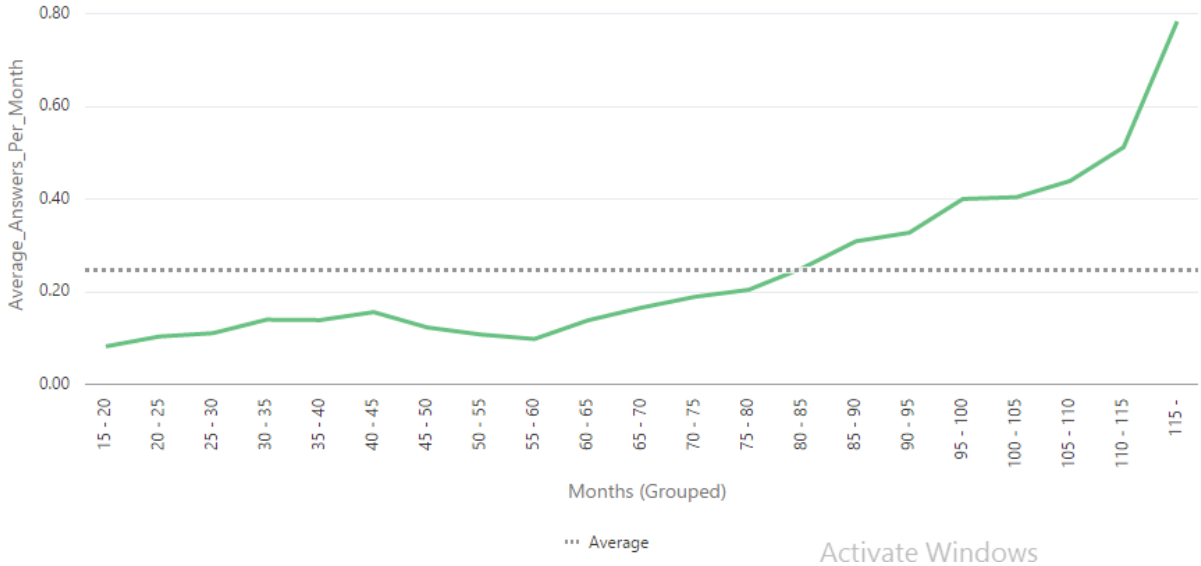

Figure 28 Answering Rate per Membership Time *[Source: Author]*

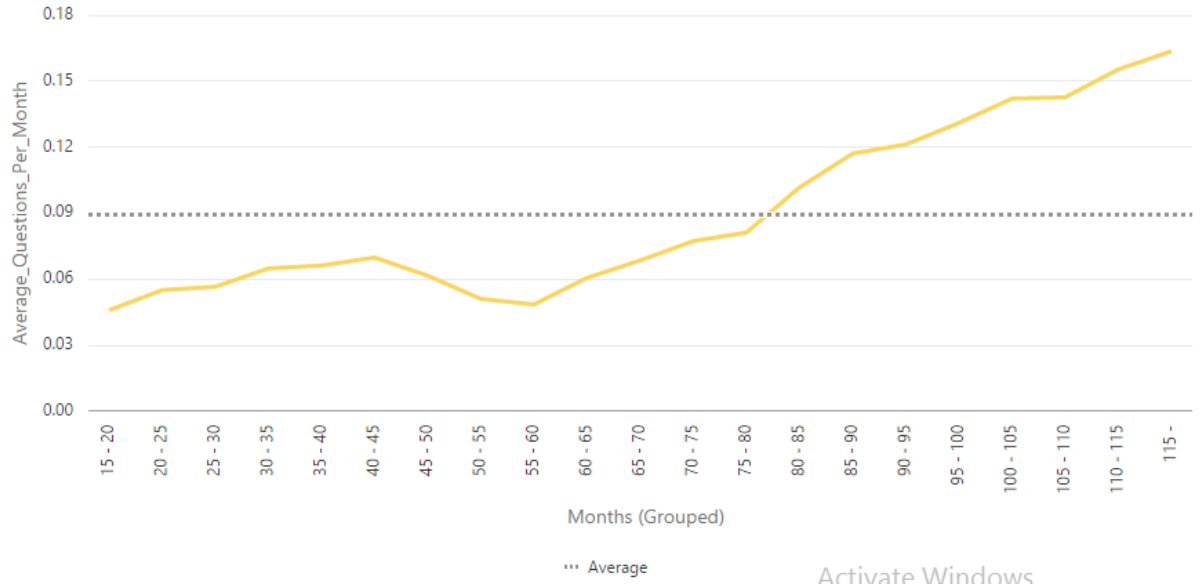

Figure 29 Question Asking Rate per Membership Time *[Source: Author]*

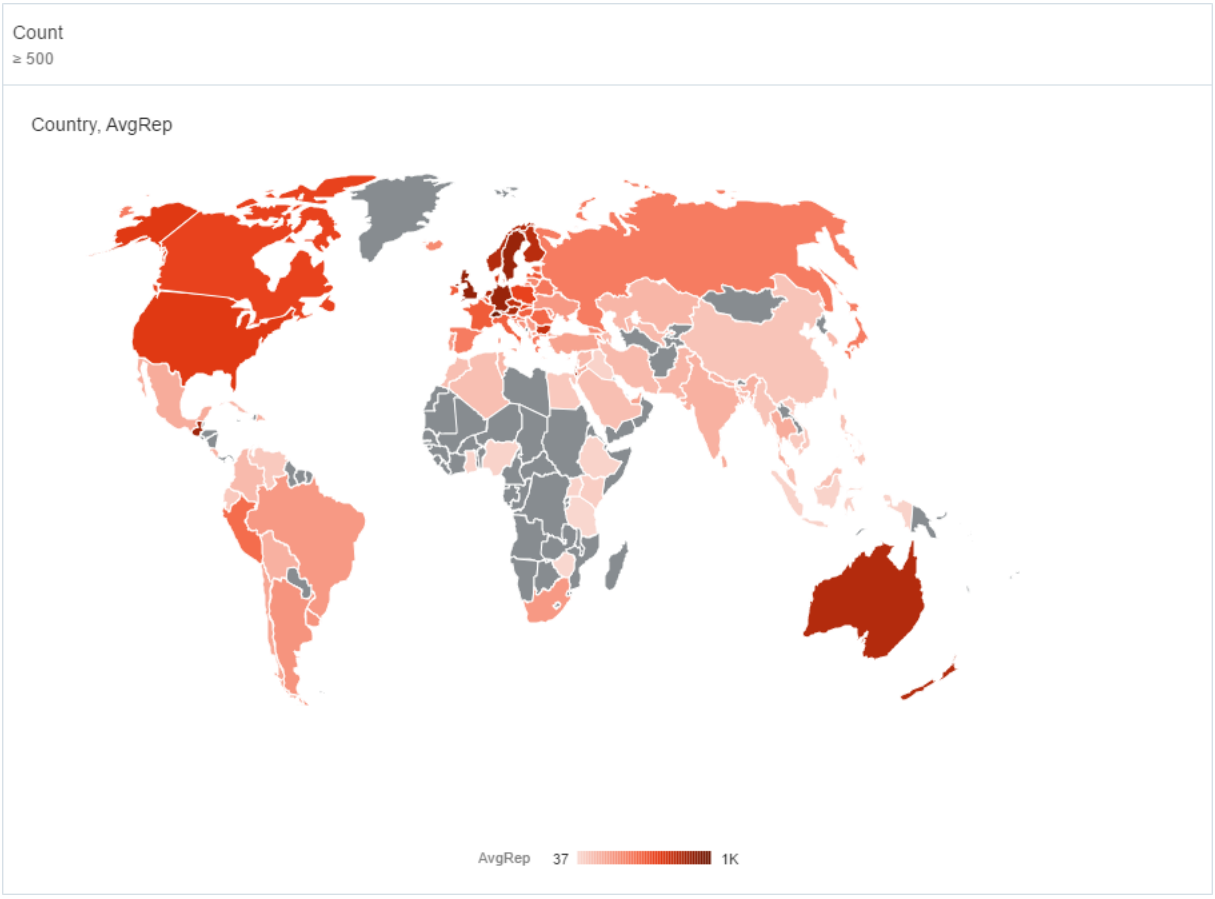

Figure 30 Reputation per Country ***[Source: Author]***