Predicting Programming Language Preferences from Big5 Personality Traits

By

Badrun Nessa Antu and ID: 012203027

Submitted in partial fulfilment of the requirements
of the degree of Masters of Science in Computer Science and Engineering

May 28, 2024

Department of Computer Science and Engineering
United International University

APPROVAL CERTIFICATE

The thesis titled "**Predicting Programming Language Preferences from Big5 Personality Traits**", Submitted by Badrun Nessa Antu, Student ID: 012203027, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Masters of Science in Computer Science and Engineering on May 2024.

BOARD OF EXAMINERS

____

Dr. Md. Saddam Hossain Mukta
Associate Professor — **Supervisor**
Department of Computer Science and Engineering
United International University

____

Dr. Dewan Md. Farid
Professor — **Head Examiner**
Department of Computer Science and Engineering
United International University

____

Dr. Mohammad Nurul Huda
Professor and Head of Dept. — **Examiner I**
Department of Computer Science and Engineering
United International University

____

Mr. Nahid Hossain
Assistant Professor and Program Coordinator — **Examiner II**
Department of Computer Science and Enginerring
United International University

____

Dr. Md. Motaharul Islam
Professor and Director of MSCSE — **Ex-officio**
Department of Computer Science and Enginerring
United International University

Declaration

I do hereby declare that the research works embodied in this thesis/project entitled "**Predicting Programming Language Preferences from Big5 Personality Traits**" is the outcome of an original work carried out by Badrun Nessa Antu under my supervision.

I further certify that the dissertation meets the requirements and the standard for the degree of [MSc/ BSc] in Computer Science and Engineering.

Dr. Md. Saddam Hossain Mukta
Associate Professor, CSE Department

Abstract

This thesis explores the relationship between the Big Five personality traits (BPT) and programming language preferences. We randomly collect data of a total of N=820 Twitter (currently X) and Stack overflow (SO) users. By analyzing the social media activity and Stack Overflow profiles of users, we aim to predict their preferred programming languages based on their BPT. We cross-link the data between Twitter and SO profiles. Then, we collect user features (i.e., users' BPT, word embedding of tweets, etc.) from Twitter and programming preferences (i.e., programming tags, reputation, question, answer, etc.) from SO. Then, we apply various machine learning (ML) and deep learning (DL) techniques to predict users' programming language preferences from their BPT. We also investigate a few interesting insights about Twitter and SO platforms and how reputation, question asking/replying associated with user's BPT. The results indicate a significant predictive capability, achieving an accuracy rate of 78%. This demonstrates that personality traits, as captured by the BPT, can be a strong indicator of programming language preference. The findings suggest potential applications in personalized learning environments, career guidance, and team composition in software development projects. This research contributes to the growing body of knowledge at the intersection of psychology and computer science, highlighting the impact of personality on technical choices. Future work could expand on these findings by exploring additional personality frameworks and incorporating larger and more diverse datasets.

Acknowledgements

This work would have not been possible without the input and support of many people over the last trimester. I would like to express my gratitude to everyone who contributed to it in some way or other.

First and foremost, I am profoundly indebted to my supervisor, Dr. Md. Saddam Hossain Mukta, Associate Professor of Computer Science and Engineering, who has given me invaluable guidance, insightful feedback, and unwavering support. His expertise and encouragement have been instrumental in shaping this thesis and helping me navigate the numerous challenges encountered along the way in the fields of Social Netwok Analysis, Psychological Behavior, Machine Learning & Deep Learning.

I am also sincerely grateful to the members of my thesis committee for their constructive criticism, suggestions, and contributions that have significantly improved this work.

Last but not the least, I owe to my family including my parents for their unconditional love and immense emotional support.

Publication List

The main contributions of this research is in preparation in journals as mentioned in the following list:

Journal Articles

1. Badun Nessa Antu, Nasreen Azad, Iftekharul Abedeen, Md. Saddam Hossain Mukta and Najmul Islam, "Analysis of Developers' Preferences and Community Behavior from Big5 Personality Traits", "Software: Practice and Experience", April 2024. [Submitted]

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview of the Study

In this study, we employ a novel strategy by computing users' Big5 personality traits from tweets and users' programming preferences from Stack Overflow. Then, we build a novel classification model to predict users' programming language preferences from their Big5 traits derived from tweets. Though we can predict users' personalities from Stack Overflow itself, the questions and answers largely reflect the programming context. On the other hand, users frequently express their thoughts, ideas, and intimate information on social media [2, 3]. Thus, identifying Big5 personality traits on social media is likely to produce accurate results. To the best of our knowledge, this study presents the first approach that exploits the fusion of two sites, i.e., Twitter and Stack Overflow, to predict users' programming language preferences. By using our approach, we can predict a user's programming language preferences from their tweets only, though they do not have any Stack Overflow profile.

Programming languages are media that help developers to translate ideas into performing a specific task. These languages can be tagged by using different topics such as *C, C++,PHP, Java script, Java, Python, C#, .net, Ruby,* and *Android*. Different programming languages may serve similar purposes, but performance of these languages may vary in terms of safety, convenience, and response towards the developers. Some developers like flexibility of Ruby, while others prefer Java for its strictness. Scripting languages such as HTML/CSS are easy to learn and require less effort to make a program. Besides, weakly typed languages such as JavaScript are more portable than other languages. Some like event driven languages like Java/Swing to create complex programs by using quick methods.

The users who contribute to the Stack Overflow (SO) usually recognized by their reputation score [4]. To gain a high reputation score, users generally put in continuous effort and should have high technical expertise. We analyze how users' reputations are linked to their Big5 personality traits and their questions and answers. Then, we compute users' Big5 personality scores by using IBM Watson personality insights API. We also reasonably narrow down the programming tags into four class labels (i.e., oop, web-pro, database, mobile) that are popular according to Stack Overflow statistics.

1.2 Motivation

Software development has become a crucial component of many sectors in the current digital era, spurring innovation and technical improvement [5]. Developers are faced with a multitude of choices when choosing the best tool for their projects due to the constantly growing variety of programming languages. To optimize software development workflows, improve team cooperation, and produce software that meets high standards, it is imperative to comprehend the factors that impact programming language preferences. Personality qualities have long been acknowledged as important determinants of how people behave and make decisions. The Big Five personality traits offer an accurate framework for describing the unique characteristics of human personalities [6]. Barric et al.[7] investigated the possible correlations between personality traits and various aspects of people's professional lives, such as career choices, job performance, and attitudes toward work. Manuel and Helmut [8] explored the complexities of software professionals' personalities and offered insightful advice on how to best manage team dynamics and task distribution for better software engineering results.

In the field of software engineering (SE), developers show interest in learning different programming languages. Studies [9, 10, 11, 12] show that developers' *Big5 personality traits* have strong influence on their programming skill, style, and performance. In this research, we are the first to predict developers' programming language preferences from their *Big5 personality traits* derived from their social media usage, i.e., tweets.

Karimi et al. [9] find a strong relationship between users' *Big5* traits and their programming styles, programming partners, and methods of problem solving. Gnambs [8] finds association between personality traits and programming aptitude (i.e., generation of new algorithm, architecture, and unconventional thinking). Enumerate the objectives in clear and specific terms. Bazelli et al. [11] analyze users' questions and answers from *Stack Overflow* to find association between their *personality traits* and frequently tagged programming topics. In the literature, we find no study that discovers direct connection between *Big5* personality traits and programming language preferences.

Through the results of this research, we hope to close the knowledge gap between software engineering and personality psychology, promote multidisciplinary cooperation, and deepen our understanding of how programming language selection affects human-computer interaction [13].

1.3 Objectives

Through a combination of predictive modeling and empirical analysis, this effort aims to create a framework for forecasting programming language preferences according to individual personality factors. It is possible to improve developer happiness, team relationships, and software development processes all around with such a framework. In summary, we have the following contributions:

- We first cross-link between Twitter and Stack Overflow social networking sites to predict users' programming language preferences from tweets.
- We have also first found a correlation between psychological traits and developers' programming language preferences.
- We build prediction models for the programming language preference of developers' based on their Big5 personality traits.
- We found associations between users' psychological traits and their community behavior, such as questioning, answering, and reputation.

1.4 Section Organization

In this section, an overview of our workflow is given. Section 2 describes the background and related work. Section 3 is all about the data collection for our work. Section 4 is about methodology. Section 5 contains findings and analyses. Section 6 contains implications and discussion. Section 7 presents the concluding remarks.

Chapter 2

Background and Related Works

2.1 Preliminaries

In this chapter, we first describe the preliminary steps that are relevant to our study. Then we also present related works to understand the modern works in this study.

2.1.1 Big5 Personality

Big5 personality or big5 factors make personality of a person. It is defined by different groups of psychologist. Researchers on personality traits have agreed with the trait theory and recommended the five-factor model as the most accurate way to evaluate personality [1, 14, 15, 16]. The Big5 personality traits—also known as openness to experience, conscientiousness, extraversion, agreeableness, and neuroticism—are used to define personality in terms of the model [17]. Big5 Personality was defined by researchers as follows:

Openness: Out of the five personality traits, openness (also known as openness to experience) stresses creativity and insight the most [18]. High-openness personality traits frequently have a diverse range of interests. Openness encompasses a variety of interest and information-seeking behaviors, as well as intellectual curiosity, inventiveness, and originality [19]. They have a natural curiosity for the outside world and other people, and they enjoy learning new things and having new experiences. Additionally, those who score highly on these personality traits often exhibit greater creativity and adventure.

Conscientiousness: Conscientiousness is a personality trait characterized by high levels of thinking, effective impulse control, and goal-directed behaviors [18]. Park et al [20] discussed that individuals with high conscientiousness are typically responsible, cautious, patient, well-organized, focused, meticulous, and achievement-oriented. Conscientious people are organized, considerate of others' feelings, and deadline-conscious.

Extraversion: Excitability, friendliness, talkativeness, assertiveness, and a high level of emotional expressiveness are characteristics of extraversion as a personality trait[18]. Karlsen et al. [21] discussed that extraversion is the personality attribute that has been linked to a variety of leadership qualities the most frequently. As a personality trait, extraversion is characterized by extroversion, friendliness, talkativeness, assertiveness, and

Figure 2.1: Big5 personality traits proposed by Goldberg et al. [1]

a high level of emotional expressiveness.

Ageeableness: Trustworthiness, benevolence, friendliness, affection, and other prosocial traits are included in the agreeableness personality feature [18]. Individuals with high levels of agreeableness are more likely to be cooperative, whereas those with low levels of this personality trait are more likely to be aggressive and occasionally even manipulative.

Neuroticism: Sadness, irritability, and emotional instability are characteristics of neuroticism, a personality trait [18]. High neurotic people frequently experience mood swings, anxiety, impatience, and melancholy. People who score lower on this personality trait tend to be more emotionally stable and resilient.

These characteristics are thought to be stable, constant, and primarily inherited in origin [15].

2.1.2 Twitter and Stack Overflow

Twitter and Stack Overflow are the most important platforms for our research.

Twitter: Twitter is a social networking site where users can send brief messages, or "tweets," and engage in conversation with others by "liking," "retweeting," and "replying" to their tweets. With millions of active users exchanging news, opinions, and personal updates in real-time.

Twitter, which was first introduced in 2006, has grown to become one of the biggest and most prominent social media platforms in the world. Businesses, organizations, and public personalities also use Twitter to communicate with their audience, share news and information, and take part in online discussions in addition to individual users.

Stack Overflow [1]: Software engineers can ask questions and receive answers on Stack Overflow. It was introduced in 2008 and has since grown to be one of the biggest online communities for developers and programmers to learn, share their expertise, and solve programming-related issues.

[1] https://stackoverflow.com/

Users can ask questions about coding issues or particular technologies, and other community members, many of whom are skilled engineers, can respond with advice. Users can vote on the best responses, leave comments, and get reputation points in addition to answering questions.

The feature-rich platform Stack Overflow offers a variety of tools and information for software engineers. Its essential characteristics including the following:

- *Question and Answer section:* The main feature of Stack Overflow is its Question and Answer section (Q&A), which enables users to post queries about coding issues or particular technologies and receive responses from other users in the community.
- *Reputation system:* Stack Overflow offers a reputation system that compensates members for their community contributions, such as publishing high-quality questions and answers, commenting on postings, and voting on answers.
- *Tag system:* Stack Overflow uses tags to organize questions and make it simpler for users to discover pertinent information. Users can browse popular tags to find new questions to answer, or they can search for questions using certain tags.
- *Voting system:* Stack Overflow offers a voting system that enables users to upvote the most helpful and correct responses, making it simpler for other users to access that information.
- *Commenting:* Users may leave feedback on questions and answers to add context, pose further queries, or offer clarification.
- *Badges:* Stack Overflow provides a range of badges to recognize users for their contributions, including responding to questions, leaving comments on articles, and hitting specific reputation score milestones.
- *Syntax highlighting:* Multiple programming languages are supported by Stack Overflow's syntax highlighting feature, which makes it simpler for users to read and comprehend code snippets in responses.
- *Notification system:* Stack Overflow features a notification system that notifies users when new responses to or comments on their questions are posted, or when those users are mentioned in another post.
- *Mobile app:* The Stack Overflow platform is accessible via a mobile app that enables users to engage with the community from their smartphones or tablets.

These characteristics make Stack Overflow a crucial tool for anyone trying to learn more about programming or solve specific coding issues, in addition to the platform's sizable and vibrant developer community.

2.2 Literature Review

In recent years, questions and answer (Q&A) websites have gained popularity for their accessibility, diversity and high quality content. They hold a vast audience from researchers to general users. *Quora* which is one of the biggest Q&A websites, introduced broad targeting that helped them reach 300 million unique users on a monthly basis. On the other hand, *Stack Overflow* is another acclaimed Q&A website in the developer community. Almost 6,300 questions are being asked per day. Until now, they have 11 million users, with an active interaction of millions of users on a daily basis. Since Stack Overflow posts are written in natural language, comprehensibility metrics and textual context analysis tools can potentially provide us with valuable information on what makes a posting perceived as trustworthy by the community.

Dehnadi [22] conducts an experiment on novice programming language students to solve a basic programming problem. The main target of the experiment is to observe the mental model of the novice students on a given programming problem. The experiment is being observed in two iterations, as before and after learning about the topic. From the experiment, the author detects three groups of students. In the first experiment, 44% of students consistently used the same model for all the questions, 39 percent students inconsistently used different models for different questions and 8 percent students did not answer. In the second experiment, almost everyone used the same model for all the questions. This test works as a reasonable predictor for the students who are likely to have easier success in introductory programming.

Gnambs [12] discusses about different personality traits and their correlation with programming aptitude. The study performs a meta-analysis on 1,695 participants. Among the personality traits, general mental ability ($p = .29$, $z = 4.71$, $p<0.001$) and openness ($z = 7.60$) are the most important for creating successful programs. Programmers with conscientiousness ($z = 2.11$) are less prone to make errors while programming. Moreover, extraversion ($z = 2.75$) also acts as another contributor for successful programming execution. Though agreeableness ($z = .18$) and neuroticism ($z = -1.21$) could not be associated with programming aptitude. The study concludes that, only three of the five personality traits - conscientiousness, openness and introversion (opposite to extroversion) directly impact one's programming aptitude. The other two traits, agreeableness and neuroticism are not correlated with one's programming aptitude.

Golbeck et al. [23] determine users' personalities from their tweets using Big5 personality traits. Twitter is one of the most widely used social media sites where people regularly share their thoughts and feelings. The authors made a Twitter app with 45 questions from Big5 personalities to observe their initial personality traits and later recruited 50 users for the final test. Afterward, the tweets are analyzed with a psycholinguistic text analyzer called Linguistic Inquiry and Word Count (LIWC). Gaussian Process and ZoreR regression analysis are being performed to predict the personality scores of the users. In another study, Quercia et al. [24] categorized Twitter users based on the Big5 personality traits by analyzing their social media data. The users are divided into listeners (follow many users), popular (followed by many users) and highly-read (often listed in others reading lists) categories based on their number of followers and following, listed counts and two different influential scores. Their study shows that, listeners and popular users generally have high extraversion and low neuroticism scores. Whereas, highly-read users tend to obtain a high openness score. Moreover, influential users are more organized about their life; thus, they place a high value on conscientiousness.

Michele et al. [25] discovered that automatic personality features can be utilized to forecast everyday behavior, such as reading preferences and purchasing tendencies. This is just one example of how this strategy can be used successfully. This technology has the potential to impact everyone around the globe with further development. In another study [26], the Environmental Concern Scale, used to gauge participants' attitudes and concerns about the environment, the General Ecological Behavior Scale, and the Self-Reported Pro Environmental Behavior Scale, used to gauge participants' pro-environmental activities, were all completed by 100 people from an online participant pool, according to the information we had. On the other hand, Kim et al. [27] they studied what motivates people to undertake such self-presentation through visual images, specifically through self-photographic photos, and how demographic characteristics, personality traits, and psychological needs could predict these particular forms of behaviors. Social media platforms provide an ideal arena for selective and positive self-presentation. As stated by Bargh and McKenna [28], "The impact of the Internet on interpersonal connections, such as those with friends and family, has been the subject of some of the most disputed research on the social consequences of the Internet." This begs the question of what types of people rely on these online social media platforms for their relationships with others[29]. Nowadays, social networks are typically used to facilitate the development of new offline relationships or to sustain existing ones. Joining social media networks on the Internet helps millions of individuals around the world connect with one another. In contrast, Amichai-Hamburger and Vinitzky [30] in their analysis of the profiles of Facebook users, concluded that Facebook use is associated with psychological traits. Amichai et al. [31] measured the effects of time constraints, the interactivity of online media, demographic profile, and cognitive demands (such as personality) on Internet use. They discovered that there is a complex association between cognitive needs and Internet use that takes into account things like career profile and accessibility to Internet links. According to a study conducted by Nihan et al. [32], the majority of young people (9 out of 10) participate in social media forums and spend 4 to 6 hours a day (or 50% of their free time) on social networking sites.

Salleh et al [33] find the impact of personalities and qualities, specifically openness, conscientiousness, and neuroticism, on the effectiveness of pair programming (PP). Their findings shed important light on the broader implications of understanding preferences in programming language choices by revealing how individual differences impact academic success. Darcy et al. [34] conduct a study that explores the complex interaction between individual variations, such as personality and programming expertise, and programmer performance. Age and ability show up as important variables, but personality and domain-specific elements do not seem to have much of an impact. Amin et al. [35] investigate the BPT and knowledge-gathering behaviors that have an effect on programmer's creativity. This study shows that while neuroticism has a negative effect on creativity intention, openness to experience, extraversion, and conscientiousness traits have a positive association.

Authors	Big5 vs. Programming preference	Twitter & SO fusion	Big5 vs. Repu.	Big5 vs. Q/A
Hannay et al.	×	×	×	×
Salleh et al.	×	×	×	×
Rahman et al.	×	×	×	×
Amin et al.	×	×	×	✓
Bazelli et al.	×	×	×	✓
Papoutsoglou et al.	×	×	×	✓
Our proposed	✓	✓	✓	✓

Table 2.1: Comparative analysis between our proposed and related frameworks.

Prior studies [12] investigates aspects of programming like aptitude, creativity, effectiveness, etc.

2.3 Gap Analysis

In light of the above discussion, our study differs from previous studies in the following aspects:

- No study investigates programming language preferences based on their Big5 personality traits derived from social media usage.
- Our study first makes a fusion between Twitter and Stack Overflow sites to cross link multiple platforms.
- Question-answering systems have become integral to information retrieval and community-driven knowledge sharing. While these systems have been extensively studied, the precise nature of the connection between user questions and the answers provided remains under explored. This gap analysis aims to illuminate this crucial aspect and assess its implications.
- Besides Big5 personalities, we use BERT embeddings to elevate the accuracy of tweet analysis. This advancement marks a significant step forward in improving the quality of insights extracted from social media content and offers a promising avenue for further research and development.

Chapter 3

Dataset

3.1 Data Collection

Data collection is the most challenging part of our study. First, we collect users randomly from *Stack Overflow* by searching the questions with the most popular tags from the users' IDs. We also search for these users on Twitter under the same user name as Stack Overflow. We find 962 users who have the same user names on both Twitter and SO. We confirmed it by observing users' profile photos, common shared websites, professional affiliations, and locations. We also notice that 55 users are not active on Twitter, and the tags of 87 users do not match our class labels. Thus, we discard the data of these users due to the lack of adequate information for predicting BPT. Our final sample size reaches a total of 820 (= 962 - 55 (inactive users) + 87 (irregular tags)). Then, we extract their programming preferences by using the Python *Stack Exchange API* implementation package [1]. Next, we also collect users' tweets by using the Python *tweepy* implementation package. Then, we compute users' Big5 personality scores by using *IBM Personality Insight API*. Arnoux et al. [36] show that *IBM Personality Insight API* outperforms the state-of-the-art techniques for personality computation. The underlying model of the API integrates word embedding features with Gaussian process regression. The API also requires eight times less data than the previous approaches. After computing users' Big5 personality scores, we store these data as independent variables and users' *Stack Overflow* 3 best tags as dependent variables.

We observe that different users use different tags for the same language. For example, c++11, c++14 and c++17 tags are used for c++. We remove these tags and replace them with the main programming language, i.e., C++. When related programming languages are combined, the dataset shows interesting patterns that give a thorough picture of developer preferences in the software development environment. The carefully gathered data from the Stack Overflow Developer reveals trends that provide insightful information about the dynamic process of language adoption. A distinct hierarchy becomes evident when we combine programming languages that are comparable and look at the resulting

[1] https://api.stackexchange.com/docs

No. of users	820
Total number of tweets	1,494,072
Average number of tweets	1,835
Maximum number of tweets of a user	3,247
Minimum number of tweets of a user	9
Total number of words in tweets	21,962,414
Average number of words in tweets	26,981
Maximum number of words in a tweet of a user	42,123
Minimum number of words in a tweet of a user	68

Table 3.1: Statistic of our Dataset

dataset. JavaScript is the most notable language, ranking first out of all the combined languages. From figure 3.1, the matching bar, which is much taller than the others, shows how widely used JavaScript is in online applications. This conclusion is consistent with the language's ongoing dominance among developers.

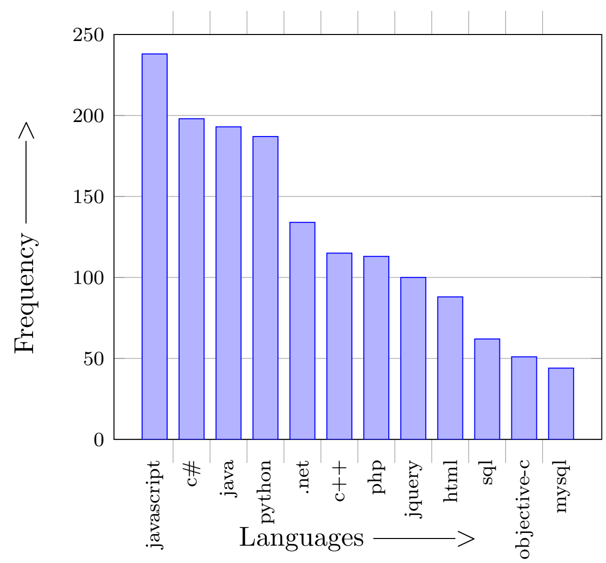


Figure 3.1: Most frequent languages

Positioned as the second most used language, C# is quite prevalent. The upward trend of C#'s bar indicates its continued relevance in ecosystems that are centered around Microsoft. This result is consistent with the language's well-established use in Microsoft technology application development. Java continues to have a prominent position, demonstrating its continued use in a variety of applications. Java's stability and adaptability are demonstrated by the height of the corresponding bar in the chart, particularly in enterprise-level applications. Python highlights its broad popularity and expanding influence across multiple domains, with a rising trend in its bar. This rise is in perfect harmony with Python's readability, adaptability, and wide range of applications.

A more complex comprehension of programming language preferences is made possible by the bar chart's visual portrayal. For those involved in the software development community, the rise of Python, the stability of Java, the growth of C#, and the importance of JavaScript all contribute to a picture of changing developer preferences.

Class label	Related Tags	Instances
oop	.net, c#, f#, asp.net, vb.net, java, python, swing, c++	900
web-pro	php, ruby, perl, jquery, javascript, node.js, angularjs, reactjs, typescript, json, django	670
mobile	kotlin, objective-c, swift, android, ios	222
database	sql, sql-server, mysql, oralce, postgreSQL	116

Table 3.2: Class Labels.

We only select the first three tags that developers usually assign to their profiles. We see that many developers provide only one tag; therefore, we do not find three different tags for many users. Some programming languages are similar in terms of structure. Therefore, we categorize the languages with similar structure, syntax and capability by a broader class label. For example, *.net* has relevance with *C#, f#, asp.net, vb.net and xml*. These languages use *Microsoft* backed object-oriented platform; we also label these languages as *oop* along with other object-oriented languages. We also categorize several tags such as *php, ruby, perl, jquery, javascript, node.js, angularjs, reactjs and typescript, html, css, json and django* as web-pro categories. We reasonably make broader categories of four different class labels: *oop, web-pro, database, mobile* from the most popular tags. Table 3.2 shows that we have a total 1,908 tags for a total of 820 users. We find that oop has the highest number of tags (900), while the database has the lowest number of tags (116).

3.2 Balancing Dataset

Table 3.2 shows the data distribution of different classes in our dataset. However, the table shows that our dataset has a class imbalance problem [37]. Our class distribution of oop, web-pro, mobile and database has a total of 900, 670, 222 and 116 instances, respectively. We observe that the oop class has a total of 900 instances, whereas the database has only 12% of instances in comparison to the total size. Studies [38, 39] discuss that the majority classifier would achieve an incredible 99.99% accuracy in a dataset with an imbalance ratio of 9999:1 and this high accuracy is misleading because all the classifier is doing is speculating on the majority class in each case. Buda et al. [40] found that class imbalance significantly weakens the classification performance. This constraint emphasizes that when the class distribution is skewed, it is difficult to classify instances. Studies [41, 42] show that oversampling is the most common technique for reducing class imbalance problems. Bader-El-Den et al. [43] use oversampling minority class representations inside the ensemble to effectively tackle the class imbalance problem.

We address the class imbalance problem by using the Synthetic Minority Oversampling Technique (SMOTE) [44]. By using SMOTE, we create artificial instances for minority classes in order to balance the distribution of classes in the dataset. Please note that SMOTE is an oversampling approach [45], that compensates for the imbalance problem without adding prejudice in favor of the dominant class by producing synthetic samples. We use the python imbalance-learn implementation package to balance our dataset.

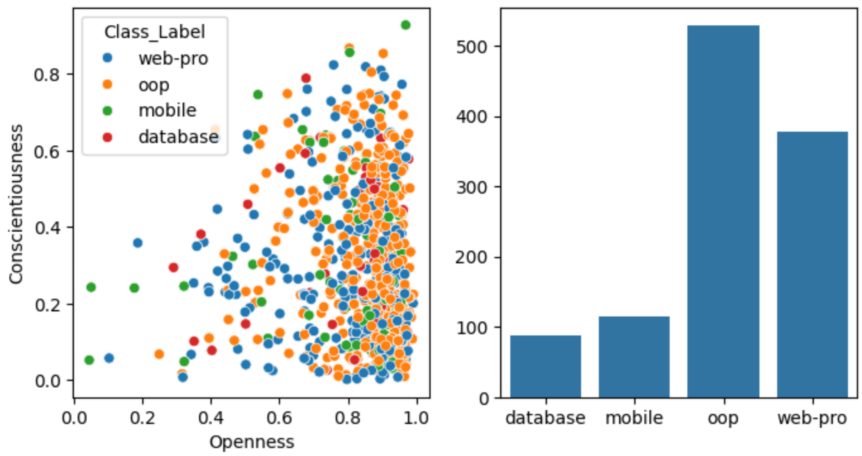


Figure 3.2: Class label distribution before using SMOTE.

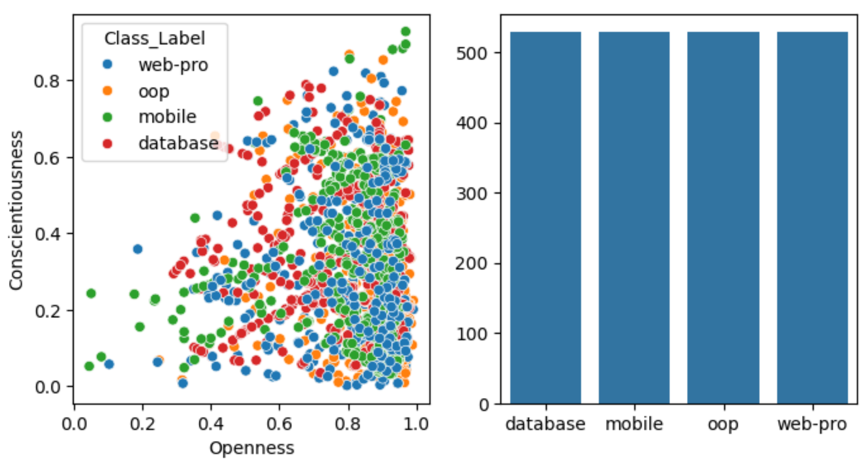


Figure 3.3: Class label distribution before using SMOTE.

Figures 3.2 and 3.3 show two bar charts showing the visual representation of the class labels before and after applying SMOTE technique. These visuals demonstrate how the dataset went from being unbalanced to being balanced, highlighting the usefulness of SMOTE in improving the dataset's suitability for reliable machine learning model training.

Class label	Before SMOTE	After SMOTE (% of oversampling)
oop	900	0%
web-pro	670	134.33%
database	116	775.87%
mobile	222	405.40%

Table 3.3: Data distribution before and after oversampling.

Table 3.3 gives a summary of the distribution of the data both before and after the Synthetic Minority Over-sampling Technique (SMOTE) oversampling. A more balanced distribution of instances among the various classes was achieved by applying the SMOTE, and this can enhance the performance of machine learning models.

Chapter 4

Methodology

This chapter describes different steps of our methodology. Then, we describe how we build our model. We also describe the explainability of our models. Finally, we measure the performance of our models.

4.1 Environment Setup

Bhavitha et. al.[46] described the fundamental usage of machine learning. We have used the Scikit Learn machine learning Python implementation package to use different classification models. We use the following things for classification models and deep learning-based classifiers:

Dataset: The dataset is the essential part of our research. We collect data from Twitter and Stack Overflow.

Csv file: All the data we use in our research is based on the file extension of CSV (Comma Seperated Values).

Platform and Programming language: Google Colab or Colaboratory is our platform where we run our codes. Python is popular for machine learning. So, we use Python for our programming language.

4.2 Methodology

To predict developers' programming preferences, we perform the following steps:

- *Cross linking between sites:* We manually cross-link between user profile of Stack Overflow and Twitter. Then, we collect developers' programming tags and tweets from both Stack Overflow and Twitter, respectively.

- *Computing Big5 Personality:* We compute Big5 personality traits by using IBM Personality Insight API.

- *BERT embedding:* We use BERT embeddings extracted from tweets of the users.

Structure of methodology: We select Big5 traits as independent variables and programming tags as dependent variables. Then, we apply multiclass classification to predict users' programming preferences from Big5 traits.

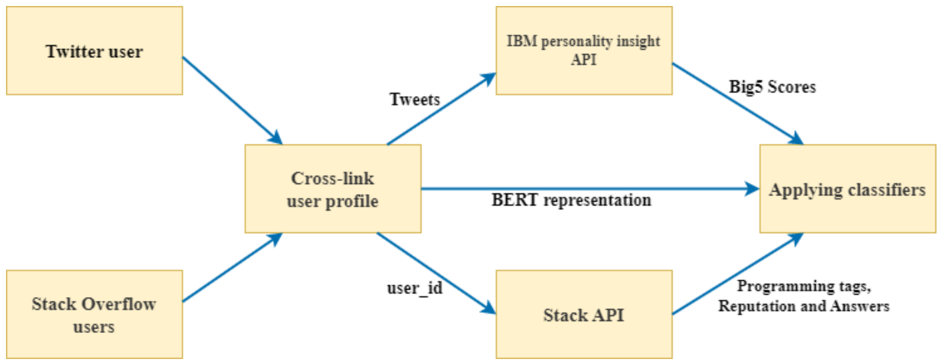


Figure 4.1: Methodology of programming language preferences from Big5 traits

4.2.1 Feature Selection

To predict programming language preference from BPT, we first select important features for applying conventional machine learning (ML) techniques. In addition to this, we compute BERT features from users' tweets for applying deep learning based models to predict programming preferences from BPT.

Features from Big5 traits

We find out relevant BPT that might affect a user's programming language preferences. Towards this, we compute a multi-level discriminant analysis [47] by using *SPSS* where BPT are predictors and users' programming language preferences are dependent variables. BPT computed from *IBM Personality Insight API* are *opennes, conscientiousness, extraversion, agreeableness and neuroticism*

	oop	web-pro	database	mobile
Openness	53.673	33.982	30.890	61.319
Conscien.	42.149	21.799	21.272	28.255
Extra.	-0.037	18.370	6.737	-1.907
Agree.	0.674	0.784	0.329	0.410
Neuro.	0.069	0.292	0.591	0.521

Table 4.1: Fisher's LDA function coefficient between users' Big5 traits and their programming preferences

For selecting important personality traits, we use *Fisher's linear discriminant analysis (LDA)* [48] Discriminant analysis finds correlation between independent variables and dependent variables having more than two class labels Fisher's LDA function scores are proportional to the coefficients of multiple linear regression with dependent variables, i.e., programming language preferences. Consequently, larger (> 1.0) score predictors are more accurate predictors [49].

The best selected features are openness, conscientiousness, extraversion.

Features from BERT

The use of BERT embeddings obtained from user tweets offers a special chance to improve social media data analysis. But to maximize model performance, interpretability, and computing efficiency, good feature selection is essential. We can use Word2Vec, but there are some reasons why we use BERT.

Why we use BERT embedding: We use BERT (Bidirectional Encoder Representations from Transformers) [76] based feature extraction over our tweets. BERT captures word meanings in context and understands word semantics based on the surrounding words. BERT also handles polysemy (multiple meanings of words) and captures nuances in language better. BERT models exist for multiple languages, making it a valuable choice for multilingual applications. The use of BERT embedding obtained from user tweets offers a special chance to improve social media data analysis.

Our approach to feature selection of BERT embbeding: Our initial approach involved using the entire embedding layer as features, considering the holistic representation provided by BERT. This approach offers the advantage of exploiting all the captured nuances in the text.

4.2.2 Building Models

In this section, we build a classification model to predict programming language preferences. We randomly split the dataset by 20% and 80% for learning weights and training the model respectively. We outline the methodology and steps undertaken to construct a predictive model that utilizes Big5 personality scores extracted from users' tweets and their reputations, with the objective of predicting programming tags. The tags have been categorized into different classes, such as "oop", "web-pro", "database", "mobile" and serve as the class labels for our classification task.

Algorithm selection: Due to the categorical nature of our target variables (class labels) and the need to handle multiple labels for each observation, we considered several classification algorithms for our modeling approach.

Building machine learning models

The following ML algorithms were evaluated:

- ***Support Vector Machines (SVMs):*** SVMs are versatile and effective for binary and multi-class classification tasks [50]. In our case, we leveraged SVM to address the task of multi-class classification, where each observation can belong to multiple classes simultaneously. We consider SVM due to its flexibility and effectiveness in handling multi-class classification problems. The SVM model was trained on our dataset, which included feature vectors derived from BPT scores [51], with each observation associated with one or more programming tags as labels.

- ***Random Forest (RF):*** RF is a powerful ensemble learning technique that has gained widespread popularity in the field of ML due to its ability to deliver robust, accurate, and versatile predictions [52]. Its versatility and outstanding performance make it a formidable choice for a broad spectrum of real-world applications.

- ***K-Nearest Neighbors (KNN):*** KNN classifier is built with five neighbors. The model is trained and then applied to predict outcomes based on unobserved data. This model is implemented by using the scikit-learn implementation package in Python. The model makes use of the KNN algorithm's capacity to classify data points according to how close they are to other points in feature space in order to precisely assign class labels to input instances [53]. For categorization challenges in machine learning applications, this method provides an adaptable and comprehensible answer [54].

Building Deep Learning Model

We also apply deep learning methods by using features extracted from BERT embedding and Big5 data with convolutional neural networks (CNN) and feedforward neural networks (FFN) [55]. These advanced frameworks offer improved performance for classification tasks.

CNN Model Architecture: In recent years, 2D convolutional neural networks (2D CNN) have been frequently employed in image processing. CNN models have also been used for text classification tasks and have shown promising results in sequence-based methods [56]. The 1D CNN is suitable for identifying specific patterns within text sequences, hence capturing complex relationships. Because pooling layers offer translation invariance, the model can concentrate on the existence of particular characteristics rather than their exact placement [57, 58]. In order to create higher-level representations at subsequent levels, the model combines low-level details it extracts from earlier layers to create hierarchical features [59]. The details of the deep learning architecture are as follows: two one-dimensional convolutional layers with a filter size of 64 and a kernel size of 3 make up the CNN model. A one-dimensional max-pooling layer follows each of these layers. These layers extract features from the input vectors. We use a fully connected neural network layer with 128 neurons to convert the retrieved features. After that, we predict the input sequence labels using a sigmoid [60] output layer. We use the activation function: ReLU [61] for all other layers. We split the data sets for training and testing by 80% and 20%, respectively. We utilize binary cross-entropy as the loss function.

FFN Model Architecture: We also add our FFN model architecture, which consists of two layers:

Layer	Hypermarameters
1D convolution	fl = 64, kr = 3, af = ReLU
1D Max-pooling	Pool-size = 2
1D convolution	fl = 64, kr = 3, af = ReLU
1D Max-pooling	Pool-size = 1
Dense	units = 128, af = ReLU
Output	unit = 4, af = sigmoid

Table 4.2: 1D CNN architecture for our experiment (fl - filter size, kr = kernel size, af = activation function)

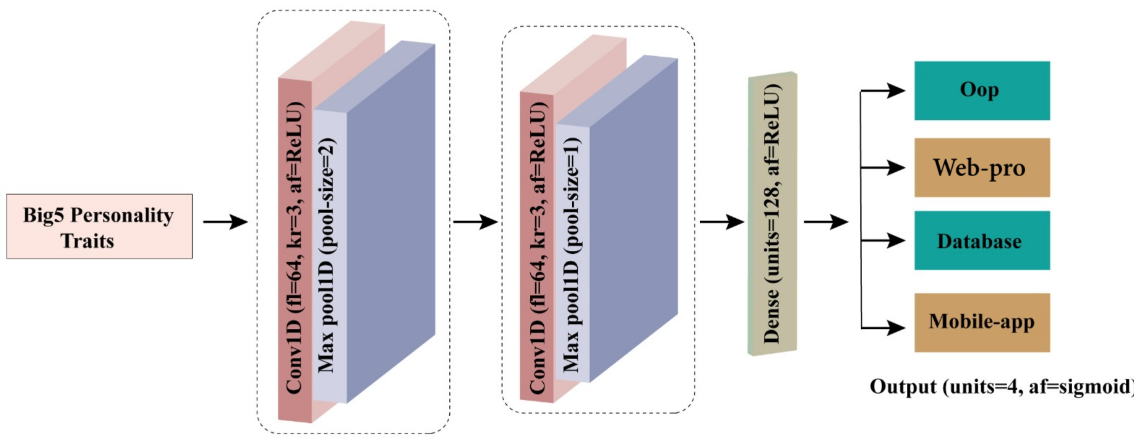


Figure 4.2: 1D CNN Architecture of our work

- Input Layer (Dense Layer): This layer takes Big5 scores and BERT embedding (768 dimensions) as input data. The ReLU activation function [62] introduces non-linearity, enabling the model to learn complex patterns in the data.

- Output Layer (Dense Layer): This layer has a single neuron, which makes it suitable for regression tasks (use for BERT embedding with reputation). The sigmoid activation function [63] is used in the output layer, producing a probability-like output that ranges from 0 to 1.

The model is compiled with an Adam optimizer. The training process is monitored using validation data to assess the model's performance over 100 training epochs.

Advantages of choosing the algorithms:

Our study uses a nonlinear approach because of it's complex relationships and data representation. The comparative advantages of each model in the context of our problem are given:

- ***SVM:***

- It can handle both linear and nonlinear data through the use of different kernel [64]. Also it works for multiclass classification by using Scikit-learn SVM module.
- SVM minimize the risk of overfitting by maximizing the margin between classes [65].

- ***Random Forest:***
  - It is an ensemble technique that combines multiple decision trees, gaining high accuracy and robustness against overfitting [66].
  - It naturally handles categorical data and interactions between variables which is important to understand relationship between personality traits and programming languages choices.

- ***K-Nearest Neighbor (KNN):***
  - KNN may provide light on how similar personalities are to one another and possibly identify groups of people who have similar tastes in programming languages.
  - By varying the number of neighbors (k), allowing for different levels of sensitivity to the dataset's characteristics [67].

- ***1D CNN:***
  - 1D CNN is suitable for categorical data, continuous data, audio/text data and sequential data [68]. Ahmed et. al. [69] stated that 1D-CNNs process one-dimensional data, such text or audio, and are comparable to 2D-CNNs.
  - 1D CNNs have the ability to detect complex correlations between programming language choices and personality attributes, potentially revealing subtle patterns that other models would miss.
  - 1D CNNs can handle fresh data effectively after they have been trained, which makes them appropriate for large-scale or real-time prediction problems.

- ***FNN:***
  - It is versatile and approximates nonlinear functions [70]. So, it can identify intricate relationship between personality traits and programming languages.
  - Modeling the relationship between personality traits and programming languages is made flexible by the architecture of FNNs, which may be customized to prioritize particular features or interactions

Explainability of the BPT-based model:

To better understand, how our model predicts the programming preference based on the BPT, we deployed the SHAP Python library [1] which enables us to look into the contribution of the feature sets and what impacts they have on the overall decision. Instead of dealing with them as black boxes, it is important to understand what makes the models behave as they do. Explainability is crucial in making sure the model is not biased or behaves predictably [71]. As our model is a multiclass classifier, we investigate how the model makes decisions for each of the individual classes. Utilizing the Shap Library (SL), graphical representations are rendered in logit space to explain these processes. The log odds $f(x)$, integral to this visualization, are derived from the probability (P) associated with the predicted class for a given sample, calculated as $f(x) = ln(P/1 - P)$. Before, analyzing and going into the details of the model, we examine the feature correlation in our dataset.

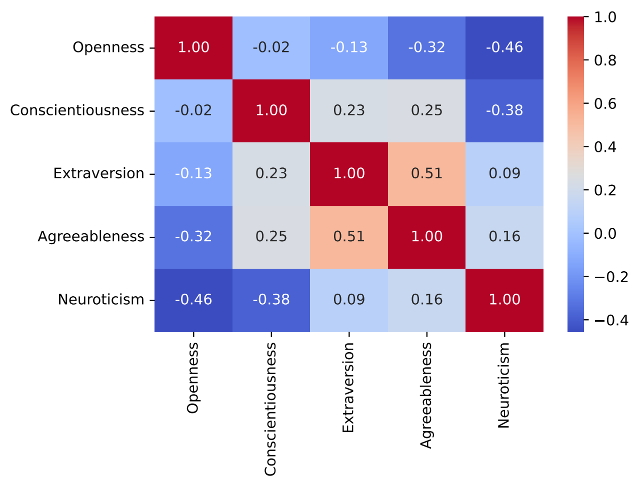
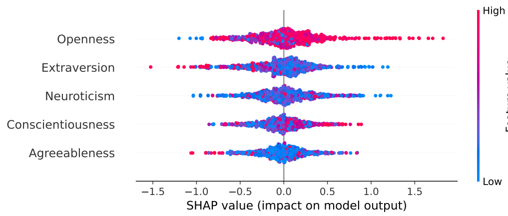


Figure 4.3: Correlation among the personality traits.

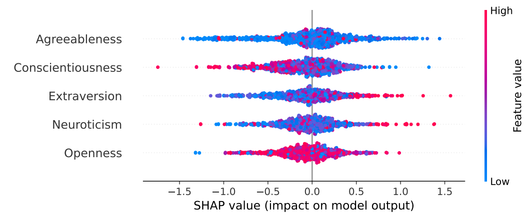
Figure 4.3 depicts the outcomes of this correlation analysis, revealing predominantly minimal correlations among the traits, barring notable exceptions such as the correlation between Openness and Neuroticism, and Extroversion and Agreeableness. The negative correlation observed between openness and neuroticism stems from the inherent incongruence between these traits; individuals characterized by openness embrace change and challenges, whereas those exhibiting neuroticism tend to display excessive caution and hesitancy in decision-making processes. Conversely, the significant positive correlation between extraversion and agreeableness arises from the shared attributes inherent in these traits. Extroversion indicates an individual's ease in social interactions and the ability to effectively convey personal interests and these qualities are congruent with high levels of agreeableness. These findings are consistent with existing literature references [72], further supporting the validity and significance of our observations regarding trait correlations

[1] https://pypi.org/project/pyshp/

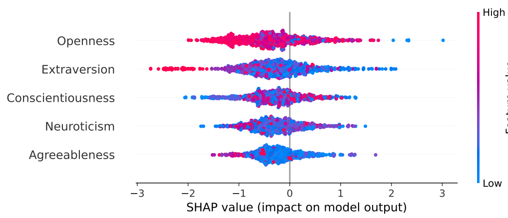
within the context of programming preference prediction.



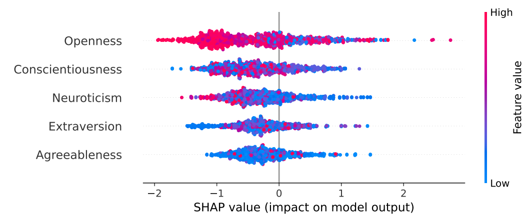
(a) OOP



(b) Web-pro



(c) Mobile



(d) Database

Figure 4.4: Contribution of BPT in the prediction of programming preferences.

To illustrate the methodology behind the model predictions, Bee Swarm charts are employed for each predicted class across the entirety of the dataset. Figure 4.4 illustrates these charts, which depict the extent of influence exerted by trait scores on the log odds of the classes, as indicated by the distance of the data points from the normal line. The density of data points within the charts reflects the frequency of samples contributing to either augmenting or diminishing specific log odds from the normal line (0.0). Furthermore, the coloration of the data points signifies the relative magnitude of the feature values: predominantly red data points positioned to the left and blue ones to the right denote that higher feature values lead to a reduction in log odds, whereas the reverse scenario suggests that lower trait scores diminish the log odds. Conversely, a combination of blue and red data points indicates a weaker correlation between the feature and the prediction of programming preference. The correlation weakens as the distribution of red and blue data points becomes more homogeneous. Notably, the occurrence of alternating bands of high and low feature values within the charts may arise from complex interactions among the features.

The visualization presented in Figure 4.5 illustrates the prediction process for a single sample drawn from the dataset. The notation $E[f(x)]$ represents the average log odds derived from all predictions associated with a specific programming preference. The term $f(x)$ denotes the log odds pertaining to an individual prediction derived from the feature set. Within the visualization, the red and blue rectangles signify Shapley values that respectively augment or diminish the log odds. The length of these rectangles corresponds to the magnitude of deviation induced by the associated feature. Upon inspection of Figure 4.5, we can see that the 'OOP' (Object-Oriented Programming) class exhibits the highest log odds compared to other programming preferences, thereby indicating it as the

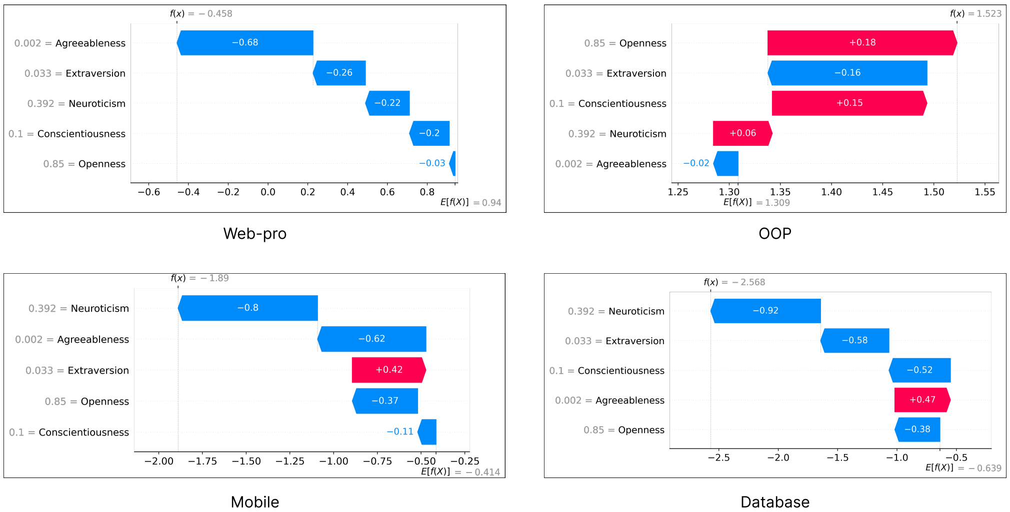


Figure 4.5: Contribution of the feature list in class prediction for one sample.

predicted class for the given sample. Notably, the features ‘Openness’, ‘Conscientiousness’, and ‘Neuroticism’ contribute positively to the log odds, whereas ‘Extraversion’ and ‘Agreeableness’ exert a negative influence.

This gives us context into how the model is making decisions and understanding the relationship between the feature set and the predicted classes. As machine learning algorithms become more sophisticated and integrated into various aspects of our lives, it’s essential to understand not only their predictive capabilities but also how they arrive at their conclusions. Explainability ensures transparency and trustworthiness in the predictive model’s outcomes. Understanding why certain traits are correlated with specific programming preferences allows for more accurate predictions and enables individuals to comprehend the reasoning behind these predictions. Moreover, explainability helps identify any biases inherent in the predictive model or the data used to train it.

Model Evaluation for programming preferences:

- ***Performance Matrix:*** Understanding the performance of classification models in practical applications depends critically on their evaluation [73]. A number of performance criteria are used to evaluate these models’ effectiveness and accuracy. Among these, accuracy (ACC), F1 score, precision, and recall stand out as key performance measures. Each provides distinct information about how well the model is working.

  – ***Accuracy:*** When the number of accurate forecasts is divided by the total number of predictions, accuracy indicates how accurate the model is overall. Accuracy, also known as precision, describes how well a model or technique does its job [74]. Equation number 4.1 shows the computation of the accuracy of a model.

- ***Precision:*** The ratio of actual positive predictions made by the model is known as precision. Precision is computed by the equation 4.2. Precision, also known as positive predictive value (PPV), is a metric used to assess the efficacy and applicability of a method within a database where data science (DS) has been used [74, 75].
- ***Recall:*** Recall, sometimes referred to as sensitivity or true positive rate (TPR) [76, 77], assesses how well the model can locate each positive case in the dataset. A high recall score means that few positive cases are missed by the model, which means it efficiently catches the majority of them. Recall is computed by equation 4.3
- ***F1 score:*** The F1 score offers a fair evaluation of a model's performance since it is the harmonic mean of precision and recall. The characteristic of the harmonic mean of two values is that it is more closely aligned with the smaller value than the bigger one [78]. When a model obtains a high F1 score, it simultaneously gets high recall and high precision. F1 score is calculated by equaltion 4.4

$$ACC = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{4.1}$$

$$precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{4.2}$$

$$recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{4.3}$$

$$F1\ score = \frac{2 \times precision \times recall}{precision + recall} \tag{4.4}$$

Classifier	Precision	Recall	F1-Score	Accuracy
SVM	0.78	0.73	0.75	0.78
RF	0.65	0.63	0.64	0.66
KNN	0.62	0.60	0.61	0.62
1D CNN	0.76	0.73	0.74	0.76
FNN	0.75	0.73	0.74	0.75

Table 4.3: Performance of Big5 traits based predicting programming preference

- ***Result:*** Table 4.3 presents the performance of our prediction model by using traditional ML models, i.e., SVM and RF by using our BPT features. Then, we obtain a performance of 78% by using SVM model. On the other hand, we find word embedding based 1D CNN and FNN models show a performance of 76% and 75% , respectively.

Chapter 5

Findings and Analytics

In this section, we present our findings by analyzing users' Big5 personality traits and their Stack Overflow and Twitter (X) activities. We investigate and answer the following findings/research questions (RQs):

- RQ1: How BPT are associated with developers' programming language preference?
- RQ2: How BPT and developers' community behavior (i.e., question, answer and reputation) are linked with each other?

5.1 RQ1

5.1.1 Motivation:

Developers' BPT likely influences their problem-solving skills, work habits, and overall compatibility with the working environment. The openness trait may have a correlation with mobile platform-based programming languages. Mobile apps are updated often to include new functionality, address bugs that have been identified, or adapt to changes in the surrounding or technological environment [79]. Trabucchi et al. [80] discussed that mobile app technology is producing a growing number of significant inventions based on digital technologies (i.e., cryptocurrency wallet, unity-game-engine, virtual-reality, augmented-reality, and kotlin-coroutines etc.) due to the widespread usage of smartphones—4.88 billion users are anticipated by 2024 (March), according to bankmycell.com. Smartphones are also a great example of Internet of Things (IoT) and big data generators because they are embedded with many sensors (e.g., compass, accelerometer, camera, GPS tracker) that enable users to gather a vast amount of data. Developers have to be creative and curious to explore new technology to update mobile applications.

Developers who are responsible, patient, and goal-oriented are likely to be interested in OOP based programming languages as well. The maintainability of programs can be enhanced when a language fully adopts the OOP idea and incorporates OOP elements [81]. Oop structure helps developers maintain code with classes and different modules to implement software (i.e., abstraction, encapsulation, and polymorphism). The conscientiousness trait might drive developers to maintain and organize the code for reuse and to make it easy to understand.

Developers who have a high extraversion score likely have a correlation with web programming (web-pro). Akimov et al. [82] discussed that e-commerce developers might need to be aware of the technologies (motion UI, serverless-framework, websockets, static site generators, jamstack, etc.) and interaction with the marketing department, IT department, as well as how these organizations collaborate to create and promote the company's e-commerce. Garaizar et al. [83] described some work of web developers in which they use JavaScript to create images or other web assets from scratch and preload them. Extraverts tend to be outgoing people who do well in situations requiring a lot of conversation and engagement. Developers in web development projects are likely to share the content, outcomes, tips, and tricks with others because of their focus on cooperation and customer involvement.

Neuroticism may not have any significant correlation with any platform-based programming language, since negative emotions may reduce working motivation [84, 85].

In the light of the above discussion, we also find an association between developers' patterns of tweets and their SO tags, which can be relatable to their BPT. We observe below that *User_X*'s tweet exhibits a high degree of conscientiousness since it shows concern about issues in society like the climate crisis and a sense of duty. On the other hand, the user likely assigns primary tags that fall under the label of oop.

Conscientiousness vs oop

User_X

Stack Overflow tags: c#, .net, entity-framework and asp.net etc.

Tweets: This is what keeps me awake at night. Coding and IT security are interesting problems. But what really scares me is the climate crisis. It's here. Now.

Besides, the tweet of *User_Y* reveals a great degree of excitement to learn new experiences because the tweet expresses interest in taking part in an intellectually challenging task. Similarly, the user also shows interest in SO with the tags of mobile application development.

Openness vs Mobile

User_Y

Stack Overflow tags: objective-c, ios, iphone and xcode etc.

Tweets: #adventofcode starts tomorrow! I hope you've all got your projects set up and ready to go. Good luck! May you get all 50 stars!

Similarly, *User_Z* praises his/her friend's creation by tweet. By sharing his/her excitement, the user hopes to establish a connection with their social circle in addition to recognizing a friend's accomplishment. This post demonstrates the extraversion personality trait by expressing happiness and engaging in social interactions. The user's stack overflow tags represent their preferences for programming tags that are associated with their BPT.

Extraversion vs Web-pro

User_Z

Stack Overflow tags: angularjs, javascript, html and jquery etc.

Tweets: Really like the finished product. My friend Bruce made this cool bellows for his outdoor fireplace.

5.1.2 Results from FLDA:

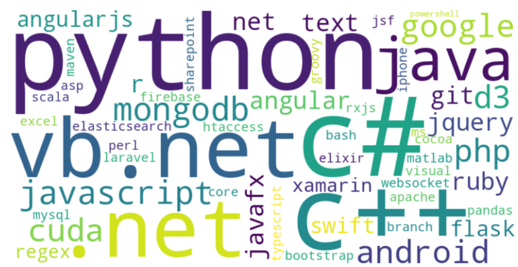
Table 4.1 shows the Fisher's coefficient of LDA between users' BPT and their programming language preferences. However, we find that our best selected features are *openness, conscientiousness* and *extraversion* traits. Openness has a higher correlation with mobile platform-based programming languages than oop, web-pro, and database. Conversely, the conscientiousness trait has a higher correlation with oop than mobile platform based programming languages. Conscientiousness: developers are likely to work with oop (python, java, c#, c++, etc.), because they may feel comfortable working with effective data modeling, implementing safe design patterns, ensuring security with oop features, and having an interest in maintaining code. However, the extraversion trait is highly correlated with web-pro. Developers with a web-pro preference are likely to share their products and ideas with others frequently. However, we find no correlation between BPT and database tags. Since the majority of the developers (i.e., oop, web-pro, and mobile) need assistance with database platforms, they also join conversations in other threads. Fan et al.[86] stated that when learning in open-ended environments, everyone should have knowledge about databases. Therefore, we did not find any specific correlation with any BPT. However, developers with the neuroticism trait have negative emotions like anxiety, stress, loneliness, etc., these psychological states may degrade working performance. In study [12], authors also find similar associations to ours (openness, conscientiousness, and extraversion) when finding correlation with programming aptitudes.

5.1.3 Developers' BPT and their topical interests:

We also explore the popular topics on SO and the prominent personality traits expressed by the participants in those topics. We provide insight into the interactions between the participant's involvement patterns based on their BPT. By combining user-generated tags with the personality trait (especially Openness and Conscientiousness), we can find a pattern by utilizing data visualization techniques, especially word cloud [87]. We may evaluate the contributions and topic interests of the participants through the distinctive view provided by the word cloud, an engaging collection of commonly used tags linked to user IDs. But what distinguishes this study is the use of personality trait data, particularly the 3rd quartile of openness and conscientiousness, which provide the visualization with a deeper understanding of the underlying reasons and user preferences. Figure 5.1

presents the relationship among user-generated content, personality traits, and community interactions. Since the cloud represents the topical interest among the users who have high openness and conscientiousness, we find that android, python, java, etc. are the most popular tags, which are also endorsed by our discovered association. Figures 5.1a and 5.1b show that the users who have higher openness and conscientiousness scores, respectively, prefer these large size words (programming technologies and languages), and those with lower scores prefer small size words (programming technologies/languages)

(a) Word Cloud for Openness (scores above 3rd quartile)



(b) Word Cloud for Conscientiousness (scores above 3rd quartile)

Figure 5.1: Word Cloud as topic of interest by openness and conscientiousness personality traits.

5.2 RQ2

As developers navigate platforms like SO and Twitter and interact with the community, their posts and responses are likely to reflect their behavioral traits [88, 89]. In SO, most of the interactions are through questions and answers, which represent users' technical interest. Whereas Twitter is more of a social platform where people post their personal experiences and thoughts, intimate personal information, daily activities, and their observations on any issue. Studies [23, 90] show that users likely reveal their actual personality traits in social media. Due to the nature of these two platforms, the way of interacting with the respective communities is expected to be different. As the posts and interactions from Twitter are more candid and informal, the analysis of one's profile here is more likely to reflect their comprehensive behavioral traits. Arnoux et al. [36] show that the IBM Personality Insight API outperforms the state-of-the-art techniques for personality prediction [1]. The underlying model of the API integrates word embedding features with Gaussian process regression. The API also requires eight times less data than the previous approaches. Several studies [91, 92] also used the IBM Personality Insight API for personality inference and showed that such text-based inferred personality can still predict real-world behavior. The API requires at least 100 words to start analyzing. We observe that we can compute personality by using the API for users who have, on average, 12–15 tweets. In this section, we explore several objectives to address these findings.

[1]https://cloud.ibm.com/apidocs/personality-insights

- **Objective 1 (O1):** Does BPT affect an individual's propensity to engage in question-and-answer interactions on SO?
- **Objective 2 (O2):** Does a person exhibit the same BPT across Twitter and SO?
- **Objective 3 (O3):** Have BPT any affect on the reputation of an individual in SO?

5.2.1 Observations:

O1:

We explore the association between the BPT and question-and-answer pairs. Table 5.1 shows the association between questions and answers for different personality traits. Openness and conscientiousness are two notable personality traits that show significant associations with the answer a user provides.

Big5 Personality Traits	p value of Ques.	p value of Replying
Openness	0.4396	0.0046*
Conscient.	0.5449	0.0217*
Extravert.	0.46598	0.111
Agreeable	0.1623	0.355968
Neuroticism	0.373679	0.829

Table 5.1: Association with Big5 personality traits and question asking/replying.

On the other hand, the connections between the questions and answers for extraversion, agreeableness, and neuroticism do not achieve statistical significance. These findings raise questions about the complex and multidimensional characteristics of personality evaluation, even though they may point to a lack of relationship between some question-answering pairings and these personality traits.

O2:

In this section, we conduct *Paired T Test* [93] to check if the means of these two scores are significantly different by using Python *stat* package [2]. After computing the BPT scores from Twitter and SO, Table 5.2 shows that all Big5 traits are significantly different from one platform to another. After computing the Big5 scores from Twitter and SO, Table 5.2 shows that all five dimensions of Big5 produce different scores. The mean values for all Big5 personality traits are higher on SO than on Twitter. The t-values for all traits are negative, indicating that the mean values on SO are significantly higher than on Twitter. The p-values for all traits are significant (<0.05), indicating that the differences between the mean values on SO and Twitter are statistically distinguishable. Since our investigation shows a significant difference in the Big5 traits computed from both of the

[2] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_rel.html

platforms, the observation of both SO and Twitter can provide better insight into one's personality traits and in turn their language preference.

Big5 Personality Traits	mean_stack.	mean_Twit.	t-value	p-value
Openness	0.992	0.824	-31.977	3.35E-146*
Conscient.	0.520	0.311	-29.505	7.07E-131*
Extravert.	0.127	0.153	5.503	4.99E-08*
Agreeable	0.003	0.053	16.598	1.52E-53*
Neuroticism	0.114	0.401	37.486	8.75E-180*

Table 5.2: Results of Paired T Test on Big5 Traits between Twitter and Stack Overflow Platforms.

The following formulae were used to determine the t- and p-values:

- ***t-value Calculation:*** The mean difference and standard deviation of the Twitter and Stack Overflow groups are used to compute the t-value for each personality feature. The equation for *t-value* is following:

$$t = \frac{\bar{d}}{s_d/\sqrt{n}} \tag{5.1}$$

  In this equation:

  - $\bar{d}$ is the mean differences between paired observation.
  - s_{d} is the standard deviation of the differences.
  - n is the number of paired observations.

- ***p-value Calculation:*** The p-value denotes the likelihood of a t-value that, under the null hypothesis, is either as extreme or more extreme than the t-value that was observed. The sample size determines the degrees of freedom, which are computed using the t-distribution. The p-value for a two-tailed test calculated by the following formula:

$$\text{p-value} = 2 \times P(T > |t|) \tag{5.2}$$

  Where:

  - T is the t-distribution with $n-1$ degrees of freedom.
  - $|t|$ is the absolute value of the observed t-value.

O3:

We briefly discuss the association between the reputation of SO users and behavior to understand the relationship between them. In SO, users' reputation scores are essentially calculated based on the number of votes they earn for participating in any of the three main Software Engineering (SE) community activities: asking, answering, and modifying questions and answers [94]. The SO website has details on various reputation-related actions, like bounty payouts and reputation limitations. Receiving thumbs-ups from other users for any of these acts entitles a user to reputation points; however, these activities have varying weights. Each of the following actions will get you points: 5, 10, and 2 for revisions [94]. Upvotes, downvotes, accepted answers, bounties, and approved suggested modifications are the main factors that impact reputation scores on SO and indicate the level of trust within the community. For this, we will be exploring each of these contributing factors to the reputation of users in the following sections.

- **Reputation vs. Question:** Reputation is a way of measuring a user's level of participation and contributions in SO. Questions are one of the primary methods by which users seek assistance and information [95]. The relevance and quality of a question can have an impact on one's reputation in SO.

	Reputation_P_value
Question	-0.0227

  Table 5.3: Association of users' reputation and their questions

  Table 5.3 presents that the p-value (<0.05) of reputation and question is significant and they are negatively correlated. A user with a good reputation is more likely to get prompted and answer correctly to their questions. Study [96] shows that questioners may face competence penalties and high humility. The study also shows that most practitioners do not ask questions at every opportunity, and questions reveal what the seeker does not know. Calefatoa et al. [97] mention how a user's reputation can influence the likelihood that their questions will be answered successfully on SO. They discuss that user-generated questions that are successful should be brief, incorporate code samples into the question body, and refrain from using unnecessary capital letters. They also talk about how a user's chances of receiving a satisfactory response to their question on SO are increased when they write their queries in a neutral manner.

- **Question vs. Answer:** From Table 5.4, we can observe that questions and answers are significantly correlated (where p-value <0.05) to each other. Users can ask and respond to questions about programming and software development on SO [98]. SO hosts a wide range of programming languages and technologies, users can post questions, and the community can subsequently respond with answers.

	Question _P_value
Answer	0.02335*

  Table 5.4: Association of users' questions and their answers.

  Zhu et al. [99] discuss the nature of questions in SO in relation to answers. They point out that both askers and answers actively participate in these discussions and that the more answers there are, the more likely participation increases. Lastly, they talk about a significant relationship between the number of answers to a question and the time it takes to respond to it, i.e., more discussed questions get slower answers. Wang et al. [100] discuss the quality of shared answers to questions on SO. They mention that users make more meaningful revisions to their answers on badge-awarding days compared to typical days. They also discuss that users were more likely to make minor answer revisions if they made many minor revisions in a single day.

- **Reputation vs. Answer:** Reputation is an estimation of a user's knowledge and participation in the community, it plays a crucial role in deciding to answer questions on SO. High-reputation users are regarded as more informed and experienced, and their responses are frequently given greater weight and consideration. Pennebaker et al.[101] demonstrated how consistently a user's reputation relates to how well their response was received. From Table 5.5, we observe that users' pattern of answer and their reputation scores are significantly correlated. Users with high reputation scores are likely to have their responses regarded more seriously by other users in SO [95].

	Reputation_P_value
Answer	0.00001114*

Table 5.5: Association of users' reputation and their answers

In SO, users ask questions, give answers, and submit comments created by others [102, 103]. In this way, users earn rewards or reputation scores, which help them introduce themselves in the online community as experts on the topic. Furthermore, people with high reputation ratings are more likely to have their responses voted up, which can make the answer more visible and aid in its user outreach. The SO question-answering system places a lot of emphasis on reputation as well as the quality of answers. While answers are made in responses to user-posted questions, reputation is a gauge of a user's participation and contributions to the website. Although reputation can influence an answer's exposure and legitimacy, it does not ensure that the answer has high merit. A user who filed a question on SO has the option to mark one of the submitted responses as an accepted answer, indicating that this is the most beneficial response. In a similar vein, users can "upvote" answers to indicate how useful they are, and users with a good reputation are typically those who have offered numerous insightful answers [95].

- **Association with BPT and Reputation**

From Table 5.6, we present the correlation coefficients between users' BPT and reputation scores. The correlation coefficient between openness and reputation is approximately 0.08 and the p-value (<0.05) is statistically significant. Our study shows that users with high openness scores likely hold a high reputation in SO.

Big5 Traits	r_value_reputation	p-value_reputation
openness	0.0792	0.02346*
Conscien.	0.0838	0.0164*
Extraversion	-0.0259	0.39901
Agreeableness	-0.0357	0.4143
Neuroticism	-0.0485	0.0147*

Table 5.6: Association of Big5 Personality Traits and users' reputation.

Answers of low quality or irrelevant to the discussion topic usually gets downvoted in SO [104]. Figure 5.2 shows the upvote to downvote ratio of SO users who fall above the third quartile of openness scores. Among the observed users, the majority of them have a large number of upvotes relative to the number of downvotes.

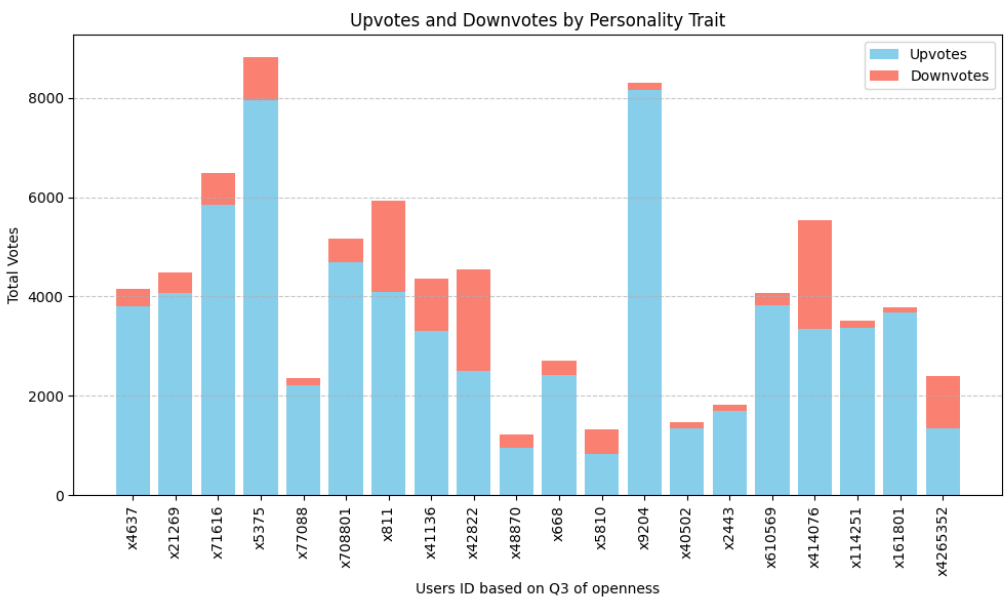


Figure 5.2: Upvote to downvote ratio of posts from users who express more openness

Conscientiousness and neuroticism show similar p-values indicating these behaviors have a significant correlation (< 0.05) to one's reputation in the SO platform. Similar to openness, conscientiousness have a positive correlation which refers users expressing more of these behaviors have a higher possibility of garnering more reputation. On the contrary, neuroticism has a negative correlation to reputation meaning developers who have negative traits such as anger, sadness, agitation, etc. may likely impact negatively on one's reputation. Among the BPTs, agreeableness and extraversion trait do not show any significant connection (p-val > 0.05) to the reputation of an individual in any SO platform.

The findings of this study have a significant impact both online and real world situation because of the growing significance of online reputation. The idea of social capital also considers the connection between a person's social status and their accomplishments or reputation[105]. According to one's social position in the social network, one's social capital is a gauge of how important they are to the community. However, there might be more factors outside only social standing that influence a user's success.

Chapter 6

Implications and Discussions

This section covers our study's main conclusions, the convergence of ML and psychology research, real-world implications and validity for our study.

6.1 Key findings

Developers' language choices are likely to be influenced by their behavioral traits, i.e., BPT. Exploring platforms such as Twitter and SO may be a reflection of the characteristics of developers, suggesting a connection between language preferences and behavior. Different ML algorithms allow us to anticipate the preferred programming languages of developers. These results can enhance our use of technology by clarifying the reasons behind people's language preferences. However, we summarise our key findings as follows:

- Developers who have high openness scores are likely to inclination with mobile platforms, while developers who are high in conscientiousness score lean more toward OOP. While extraversion may be in line with collaborative web projects. The impact of agreeableness and neuroticism seems to be negligible.

- Our article explores how involvement with question-and-answer sessions on SO is influenced by BPT. We discover the importance of openness and conscientiousness traits on interaction dynamics, which is statistically significant. However, extraversion, agreeableness, and neuroticism traits do not exhibit significant association, highlighting the complexity of assessing personality in this situation.

- BPTs represent significant variance on Twitter and SO based dataset. Twitter represents users' comprehensive behavior, i.e., BPT, while SO only shows their behavior in a niche professional community. Twitter is a social media platform intended for general use, therefore is less focused on technical question/answer activities. Though one has the freedom to express one's questions and answers in their own expressions, it is always bounded within the technical knowledge domain. This restricts the expression of emotions and personalities, whereas Twitter allows for a broader range of subjects and opinions, leading to distinct patterns in the behavior.

- We also explore the relationship between the behavior of SO users and their reputation. Reputation and users' BPT, questions, and answers have significant connections that highlight the intricate relationship between personality, conduct, and online reputation. SO answers may get upvoted or downvoted. Though upvotes may act as a reward, the likelihood of getting downvoted can deter people from answering. Developers who are more open-minded like to take on this as a challenge and reply to posted questions. Likewise, someone who is more conscientious pays more attention to details, which can lead to a more in-depth understanding of the domain. As answering posted questions/problems requires mastery of their domain, conscientiousness can explain the higher correlations. Also, in SO the likelihood of someone posting a question is dictated by the encounter of problems that someone may lack understanding of the site. The decision to post a question is more influenced by the situation than one's character traits. This may explain the lack of correlation between the BPT and the likelihood of posting questions in SO.

6.2 Convergence of ML and psychology

The findings of our research work has explored the complex landscape where BPT and users' behavior converge, particularly within the context of software development community (i.e., question, answer, and reputation in SO). Ahmed et al. [106] describe that in the collaborative environment of software development, soft skills like personality traits, social interaction skills, communication, and personal habits increase the likelihood of success of an individual and positively contribute to the project's overall aim. In a nutshell, our exploration of the intersection of psychology and technology is not just theoretical. It is a practical guide for companies aiming to build highly effective software development teams [107]. Organizations can position themselves for success in a fast-moving market by identifying the hidden combination of personality traits, reputation, and domain-specific preferences [108, 109]. In simple terms, this is the foundation for promoting creativity and achieving excellence in software projects, ultimately leading the business to long-term success.

By using effective machine learning methods, our research is in line with the wider movement of interdisciplinary investigations at the interface between computer science and psychology. Our findings are comparable with general social structures, demonstrating the increasing acknowledgement that individual differences influence digital experiences and career results [110, 111]. This study adds new perspectives to the fields of software engineering and HCI (human-computer interaction) [112], while also complementing and expanding upon the larger corpus of work in the fields of machine learning, online behavior analysis, and personality psychology.

6.3 Practical implications

Our investigation for AI-based prediction models for programming language selection discover the relationship between user behavioral attributes, online activity, and technology preferences. We propose a few practical implications related to complicated human nature and preferences across a range of programming platform related context.

- Recruiters, policy makers, job agencies, etc. can analyze users' tweets to understand their inclination towards a programming platform/technology. Since we demonstrate association with BPT, the mentioned stakeholders can also conduct a straight forward survey (i.e., IPIP, TIPI, etc.) for recommending users' to be involved in a suitable platform if available tweets or open-ended writing content are not found.
- Technologies are upgrading on a continuous basis, even new technologies and paradigms are introducing. In our article, we divided the technologies into four basic aspects such as: object construct, web based platforms, technologies for hand held devices and overall data storing and manipulating systems. However, based on our categorization, we can distribute the new technologies to its respective classes if such platforms are found. Therefore, it is also possible to map the new trends and technologies according to our findings.
- Since we discover association with BPT and community activities such as questioning, answering and reputation, our method may also able to play important role to distribute tasks among the team members during project responsibility assignments.
- Since BPT is one of the powerful instruments in psychology research, we can consider these traits as middle layer. For example, we may deploy gamification, metaverse, interactive simulation, etc. to understand users' BPT which can be mapped with programming preferences. There are several studies [113] that predict users' BPT by using a gamification based approach.

6.4 Threats to validity

- *Internal validity:* Internal validity represents the degree to which variables in the research environment may affect the results and interpretation of the information gathered. A possible risk to the internal validity of the suggested technique is related to the data collection process. We have gathered data from Twitter and SO based on various users' tweets and programming tags. For tags, we should consider all possible tags for our research, but we have used selective tags from the users' profiles, which may not give predictions for all languages. It takes careful investigation and interpretation to see how BPT interacts with various aspects of online behavior, such as reputation or question-answer dynamics.

- *External validity:* Our research primarily focuses on Twitter and SO which may limit the insight of other social and online behavioral communities. Distinct user groups, engagement rules, and content categories across platforms may have an impact on the behavior patterns and personality traits that are observed. Our findings are based on specific time duration. However, user preferences and BPT may change over time [114] and with the new development of technologies. This could restrict the applicability of our results.

- *Construct validity:* We make sure that the measurement of personality traits is consistent with accepted psychological theories and constructions by utilizing approved tools or procedures, such as the IBM Personality Insight API. Stack Overflow reputation scores are a good indicator of a user's reliability and value to the development community. Based on clearly specified parameters like upvotes, acceptable responses, and badges, these reputation scores make sure that the measurement is in keeping with the concept of reputation as intended in the context of online platforms. The research is based on well-established frameworks and theories from computer science, sociology, and psychology, including social network theory and the Big Five personality theory. We make sure that the ideas being examined are theoretically significant and pertinent to the research domain by lining up our research questions and hypotheses with these theoretical frameworks.

Chapter 7

Conculsion

This section contains the limitations, future directions, and conclusion of our work.

7.1 Limitation

We acknowledge some limitations of our work. Some other factors beyond BPT may influence developers' programming preferences. Our selective data collection is potentially limiting predictions for all programming languages. Our research is primarily focused on Twitter and Stack Overflow, potentially limiting insights from other online behavioral communities, with findings based on specific time durations and subject to changes in user preferences and technological developments.

7.2 Future directions

Considering the limitations and other factors, we can extend our research into several future avenues.

- *Content type for prediction:* In our article, we mainly predict users' BPT from their writing pattern, i.e., text, used in Twitter by using IBM personality insight API. However, in recent time several studies also use multi modal content type such as image [115], EEG from brain signals [116], gait analysis from video [117], speech [118] and many more to predict users' BPT. We may find association of users' programming preferences extracted from multi modal content type for better comprehension of user behavior.

- *Traits of psychological behavior:* In our study, we mainly find users' programming platform recommendation from the BPT. However, we may consider other important human behavior such as values, self-efficacy, emotion, sentiment to find inclination with their programming platforms and technologies. In recent times, people are also checking how differently their brain works, neurodiversity, can be associated with BPTs, consequently how they are correlated with users' programming language recommendation.

- *Use of cutting edge technologies:* Though we have demonstrated deep learning and explainability of the features, we may still use further advanced deep learning based techniques such as attention, and transformer based approaches in the computational modules. Furthermore, we can use architectural features, i.e., degree centrality, betweenness, clique relationship among the users, similarity among the users, etc. may produce different relationship with users' programming recommendation.

- *Dynamics of BPT:* We can analyze long-term data to evaluate if/how BPT and preferences change/evolve over time. Apart from long-term, short-term studies can also help identify trends and pickup rates.

7.3 Conclusion

Our study has explored the intriguing relationship that exists between users' BPT, and programming languages within the framework of online communities. The various predictive models showed impressive accuracy, highlighting their potential for real-world programming language recommendation applications. Examining the BPT traits revealed how various traits affect patterns of user involvement. We have conducted an analysis of platform differences between SO and Twitter, which revealed subtle aspects of user behavior. Furthermore, the complex interrelationship among inquiring, answering, and reputation on SO has demonstrated the dynamic nature of the knowledge-sharing ecosystem. Furthermore, a more thorough analysis of the complex relationships between the BPTs and online behavioral dynamics could improve our understanding of user interactions.

References

[1] Lewis R Goldberg. The structure of phenotypic personality traits. *American psychologist*, 48(1):26, 1993.

[2] Ryan Boyd, Steven Wilson, James Pennebaker, Michal Kosinski, David Stillwell, and Rada Mihalcea. Values in words: Using language to evaluate and understand personal values. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 9, pages 31–40, 2015.

[3] Mitja D Back, Juliane M Stopfer, Simine Vazire, Sam Gaddis, Stefan C Schmukle, Boris Egloff, and Samuel D Gosling. Facebook profiles reflect actual personality, not self-idealization. *Psychological science*, 21(3):372–374, 2010.

[4] Amiangshu Bosu, Christopher S Corley, Dustin Heaton, Debarshi Chatterji, Jeffrey C Carver, and Nicholas A Kraft. Building reputation in stackoverflow: an empirical investigation. In *2013 10th Working conference on mining software repositories (MSR)*, pages 89–92. IEEE, 2013.

[5] Daniel Smith. The effects of federal research and development subsidies on firm commercialization behavior. *Research Policy*, 49(7):104003, 2020.

[6] Paul T Costa and Robert R McCrae. A five-factor theory of personality. *The five-factor model of personality: Theoretical perspectives*, 2:51–87, 1999.

[7] Murray R Barrick and Michael K Mount. The big five personality dimensions and job performance: a meta-analysis. *Personnel psychology*, 44(1):1–26, 1991.

[8] Manuel Wiesche and Helmut Krcmar. The relationship of personality models and development tasks in software engineering. In *Proceedings of the 52nd ACM conference on Computers and people research*, pages 149–161, 2014.

[9] Zahra Karimi and Stefan Wagner. The influence of personality on computer programming: a summary of a systematic literature review. 2014.

[10] Zahra Karimi, Ahmad Baraani-Dastjerdi, Nasser Ghasem-Aghaee, and Stefan Wagner. Links between the personalities, styles and performance in computer programming. *Journal of Systems and Software*, 111:228–241, 2016.

[11] Blerina Bazelli, Abram Hindle, and Eleni Stroulia. On the personality traits of stackoverflow users. In *2013 IEEE international conference on software maintenance*, pages 460–463. IEEE, 2013.

[12] Timo Gnambs. What makes a computer wiz? linking personality traits and programming aptitude. *Journal of Research in Personality*, 58:31–34, 2015.

[13] Muhammad Usman and Nasir Mehmood Minhas. Use of personality tests in empirical software engineering studies. 2019.

[14] Paul T Costa Jr and Robert R McCrae. Longitudinal stability of adult personality. In *Handbook of personality psychology*, pages 269–290. Elsevier, 1997.

[15] Paul T Costa and Robert R McCrae. *Objective personality assessment*. Springer, 1978.

[16] Robert Hogan. A socioanalytic perspective on the five-factor model. 1996.

[17] Timothy A Judge, Chad A Higgins, Carl J Thoresen, and Murray R Barrick. The big five personality traits, general mental ability, and career success across the life span. *Personnel psychology*, 52(3):621–652, 1999.

[18] Robert A Power and Michael Pluess. Heritability estimates of the big five personality traits based on common genetic variants. *Translational psychiatry*, 5(7):e604–e604, 2015.

[19] Imran Ali. Personality traits, individual innovativeness and satisfaction with life. *Journal of Innovation & Knowledge*, 4(1):38–46, 2019.

[20] Mihye Park and Seongsu Kim. Effects of personality traits and team context on individual innovative behavior (exploitation and exploration). *Sustainability*, 14(1):306, 2021.

[21] Håvard R Karlsen and Eva Langvik. Facet level effects of extraversion on leadership behaviours rated by subordinates. *Cogent Psychology*, 8(1):1930712, 2021.

[22] Saeed Dehnadi. Testing programming aptitude. In *PPIG*, page 9. Citeseer, 2006.

[23] Jennifer Golbeck, Cristina Robles, Michon Edmondson, and Karen Turner. Predicting personality from twitter. In *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*, pages 149–156. IEEE, 2011.

[24] Daniele Quercia, Michal Kosinski, David Stillwell, and Jon Crowcroft. Our twitter profiles, our selves: Predicting personality with twitter. In *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*, pages 180–185. IEEE, 2011.

[25] Michelle X Zhou, Jeffrey Nichols, Tom Dignan, Steve Lohr, Jennifer Golbeck, and James W Pennebaker. Opportunities and risks of discovering personality traits from social media. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pages 1081–1086. 2014.

[26] Tara Wuertz. *Personality traits associated with environmental concern.* PhD thesis, Walden University, 2015.

[27] Ji Won Kim and T Makana Chock. Personality traits and psychological motivations predicting selfie posting behaviors on social networking sites. *Telematics and Informatics*, 34(5):560–571, 2017.

[28] John A Bargh, Katelyn YA McKenna, et al. The internet and social life. *Annual review of psychology*, 55(1):573–590, 2004.

[29] Teresa Correa, Amber Willard Hinsley, and Homero Gil De Zuniga. Who interacts on the web?: The intersection of users' personality and social media use. *Computers in human behavior*, 26(2):247–253, 2010.

[30] Yair Amichai-Hamburger and Gideon Vinitzky. Social network use and personality. *Computers in human behavior*, 26(6):1289–1295, 2010.

[31] Yair Amichai-Hamburger, Or Kaynar, and Adi Fine. The effects of need for cognition on internet use. *Computers in Human Behavior*, 23(1):880–891, 2007.

[32] Nihan Özgüven and Burcu Mucan. The relationship between personality traits and social media use. *Social Behavior and Personality: an international journal*, 41(3):517–528, 2013.

[33] Norsaremah Salleh, Emilia Mendes, and John Grundy. Investigating the effects of personality traits on pair programming in a higher education setting through a family of experiments. *Empirical Software Engineering*, 19:714–752, 2014.

[34] David P Darcy and Meng Ma. Exploring individual characteristics and programming performance: Implications for programmer selection. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 314a–314a. IEEE, 2005.

[35] Aamir Amin, Shuib Basri, Mobashar Rehman, Luiz Fernando Capretz, Rehan Akbar, Abdul Rehman Gilal, and Muhammad Farooq Shabbir. The impact of personality traits and knowledge collection behavior on programmer creativity. *Information and Software Technology*, 128:106405, 2020.

[36] Pierre-Hadrien Arnoux, Anbang Xu, Neil Boyette, Jalal Mahmud, Rama Akkiraju, and Vibha Sinha. 25 tweets to know you: A new model to predict personality with social media. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 472–475, 2017.

[37] Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C Prati, Bartosz Krawczyk, and Francisco Herrera. *Learning from imbalanced data sets*, volume 10. Springer, 2018.

[38] Cheng G Weng and Josiah Poon. A new evaluation measure for imbalanced datasets. In *Proceedings of the 7th Australasian Data Mining Conference-Volume 87*, pages 27–32, 2008.

[39] T Ryan Hoens and Nitesh V Chawla. Imbalanced datasets: from sampling to classifiers. *Imbalanced learning: Foundations, algorithms, and applications*, pages 43–59, 2013.

[40] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259, 2018.

[41] Kushankur Ghosh, Colin Bellinger, Roberto Corizzo, Paula Branco, Bartosz Krawczyk, and Nathalie Japkowicz. The class imbalance problem in deep learning. *Machine Learning*, pages 1–57, 2022.

[42] Haseeb Ali, MN Mohd Salleh, Rohmat Saedudin, Kashif Hussain, and Muhammad Faheem Mushtaq. Imbalance class problems in data mining: A review. *Indonesian Journal of Electrical Engineering and Computer Science*, 14(3):1560–1571, 2019.

[43] Mohammed Bader-El-Den, Eleman Teitei, and Todd Perry. Biased random forest for dealing with the class imbalance problem. *IEEE transactions on neural networks and learning systems*, 30(7):2163–2172, 2018.

[44] Bhagat Singh Raghuwanshi and Sanyam Shukla. Smote based class-specific extreme learning machine for imbalanced learning. *Knowledge-Based Systems*, 187:104814, 2020.

[45] Junnan Li, Qingsheng Zhu, Quanwang Wu, and Zhu Fan. A novel oversampling technique for class-imbalanced learning based on smote and natural neighbors. *Information Sciences*, 565:438–455, 2021.

[46] BK Bhavitha, Anisha P Rodrigues, and Niranjan N Chiplunkar. Comparative study of machine learning techniques in sentimental analysis. In *2017 International conference on inventive communication and computational technologies (ICICCT)*, pages 216–221. IEEE, 2017.

[47] RR Mallios. Predicting class ii mhc/peptide multi-level binding with an iterative stepwise discriminant analysis meta-algorithm. *Bioinformatics*, 17(10):942–948, 2001.

[48] Jian Hu, Fan Su, Xia Ren, Lei Cao, Yumei Zhou, Yuhan Fu, Grace Tatenda, Mingfei Jiang, Huan Wu, and Yufeng Wen. Prediction of carotid plaque by blood biochemical indices and related factors based on fisher discriminant analysis. *BMC Cardiovascular Disorders*, 22(1):371, 2022.

[49] Ahmed Shahriar Sakib, Md Saddam Hossain Mukta, Fariha Rowshan Huda, AKM Najmul Islam, Tohedul Islam, and Mohammed Eunus Ali. Identifying insomnia from social media posts: psycholinguistic analyses of user tweets. *Journal of medical Internet research*, 23(12):e27613, 2021.

[50] Zafar Mehmood and Sohail Asghar. Customizing svm as a base learner with adaboost ensemble to learn from multi-class problems: A hybrid approach adaboost-msvm. *Knowledge-Based Systems*, 217:106845, 2021.

[51] Denis Eka Cahyani and Anas Falih Faishal. Classification of big five personality behavior tendencies based on study field with twitter analysis using support vector machine. In *2020 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pages 140–145. IEEE, 2020.

[52] Anjaneyulu Babu Shaik and Sujatha Srinivasan. A brief survey on random forest ensembles in classification model. In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 2*, pages 253–260. Springer, 2019.

[53] Munwar Ali, Low Tang Jung, Abdel-Haleem Abdel-Aty, Mustapha Y Abubakar, Mohamed Elhoseny, and Irfan Ali. Semantic-k-nn algorithm: An enhanced version of traditional k-nn algorithm. *Expert Systems with Applications*, 151:113374, 2020.

[54] Djamel Djenouri, Roufaida Laidi, Youcef Djenouri, and Ilangko Balasingham. Machine learning for smart building applications: Review and taxonomy. *ACM Computing Surveys (CSUR)*, 52(2):1–36, 2019.

[55] Ahmed R Abas, Ibrahim Elhenawy, Mahinda Zidan, and Mahmoud Othman. Bert-cnn: A deep learning model for detecting emotions from text. *Computers, Materials & Continua*, 71(2), 2022.

[56] Saeed Ghaderizadeh, Dariush Abbasi-Moghadam, Alireza Sharifi, Na Zhao, and Aqil Tariq. Hyperspectral image classification using a hybrid 3d-2d convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:7570–7588, 2021.

[57] Xiaoming Zhao, Zhiwei Tang, and Shiqing Zhang. Deep personality trait recognition: a survey. *Frontiers in Psychology*, 13:839619, 2022.

[58] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.

[59] Moein Heidari, Amirhossein Kazerouni, Milad Soltany, Reza Azad, Ehsan Khodapanah Aghdam, Julien Cohen-Adad, and Dorit Merhof. Hiformer: Hierarchical multi-scale representations using transformers for medical image segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6202–6212, 2023.

[60] Meliboev Azizjon, Alikhanov Jumabek, and Wooseong Kim. 1d cnn based network intrusion detection with normalization on imbalanced data. In *2020 international conference on artificial intelligence in information and communication (ICAIIC)*, pages 218–224. IEEE, 2020.

[61] Xiuyan Li, Rengui Lu, Qi Wang, Jianming Wang, Xiaojie Duan, Yukuan Sun, Xiaojie Li, and Yong Zhou. One-dimensional convolutional neural network (1d-cnn) image reconstruction for electrical impedance tomography. *Review of scientific instruments*, 91(12), 2020.

[62] Edna C Too, Li Yujian, Pius Kwao Gadosey, Sam Njuki, and Firdaous Essaf. Performance analysis of nonlinear activation function in convolution neural network for image classification. *International Journal of Computational Science and Engineering*, 21(4):522–535, 2020.

[63] Grzegorz Dudek. Data-driven learning of feedforward neural networks with different activation functions. In *International Conference on Artificial Intelligence and Soft Computing*, pages 66–77. Springer, 2021.

[64] Vinod Kumar Chauhan, Kalpana Dahiya, and Anuj Sharma. Problem formulations and solvers in linear svm: a review. *Artificial Intelligence Review*, 52(2):803–855, 2019.

[65] Feiping Nie, Zhezheng Hao, and Rong Wang. Multi-class support vector machine with maximizing minimum margin. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14466–14473, 2024.

[66] Víctor Aceña, Isaac Martín de Diego, Rubén R Fernández, and Javier M Moguerza. Minimally overfitted learners: a general framework for ensemble learning. *Knowledge-Based Systems*, 254:109669, 2022.

[67] Liliya A Demidova. Two-stage hybrid data classifiers based on svm and knn algorithms. *Symmetry*, 13(4):615, 2021.

[68] Zhenyu Yuan, Yuxin Jiang, Jingjing Li, and Handong Huang. Hybrid-dnns: Hybrid deep neural networks for mixed inputs. *arXiv preprint arXiv:2005.08419*, 2020.

[69] Adel A Ahmed, Waleed Ali, Talal AA Abdullah, and Sharaf J Malebary. Classifying cardiac arrhythmia from ecg signal using 1d cnn deep learning model. *Mathematics*, 11(3):562, 2023.

[70] Alexey Beskopylny, Alexandr Lyapin, Nikita Beskopylny, and Elena Kadomtseva. Comparison of the efficiency of neural network algorithms in recognition and classification problems. In *E3S Web of Conferences*, volume 224, page 01025. EDP Sciences, 2020.

[71] Donghee Shin. The effects of explainability and causability on perception, trust, and acceptance: Implications for explainable ai. *International Journal of Human-Computer Studies*, 146:102551, 2021.

[72] Michelle A Harris and Ulrich Orth. The link between self-esteem and social relationships: A meta-analysis of longitudinal studies. *Journal of personality and social psychology*, 119(6):1459, 2020.

[73] Rung-Ching Chen, Christine Dewi, Su-Wen Huang, and Rezzy Eko Caraka. Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*, 7(1):52, 2020.

[74] Yishu Bai, Jiachen Tu, Mengzhuo Yang, Liang Zhang, and Peter Denno. A new aggregation algorithm for performance metric calculation in serial production lines with exponential machines: design, accuracy and robustness. *International Journal of Production Research*, 59(13):4072–4089, 2021.

[75] Daniel Berrar et al. Performance measures for binary classification., 2019.

[76] Bradley J Erickson and Felipe Kitamura. Magician's corner: 9. performance metrics for machine learning models, 2021.

[77] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[78] David J Hand, Peter Christen, and Nishadi Kirielle. F*: an interpretable transformation of the f-measure. *Machine Learning*, 110(3):451–456, 2021.

[79] Islem Saidani, Ali Ouni, Md Ahasanuzzaman, Safwat Hassan, Mohamed Wiem Mkaouer, and Ahmed E Hassan. Tracking bad updates in mobile apps: A search-based approach. *Empirical Software Engineering*, 27(4):81, 2022.

[80] Daniel Trabucchi and Tommaso Buganza. Data-driven innovation: switching the perspective on big data. *European Journal of Innovation Management*, 22(1):23–40, 2019.

[81] Aaron Friesen. Designing programming languages for writing maintainable software. 2023.

[82] Oleksandr Akimov, Marta Karpa, Oksana Parkhomenko-Kutsevil, Vasily Kuprichuk, Azad Omarov, kimov, rp, Parhomenko-Kucevil, A marov, and B . Entrepreneurship education of the formation of the e-commerce managers professional qualities. *International Journal of Entrepreneurship*, 25(7):1–8, 2021.

[83] Pablo Garaizar and Ulf-Dietrich Reips. Best practices: Two web-browser-based methods for stimulus presentation in behavioral experiments with high-resolution timing requirements. *Behavior Research Methods*, 51:1441–1453, 2019.

[84] Woojin Park et al. A study of influence of b anker's big5 personality traits on entrepreneurial intention: Mediated effect of digital media utilization capability. *Asia-Pacific Journal of Business Venturing and Entrepreneurship*, 15(2):209–220, 2020.

[85] Abira Reizer, Yael Brender-Ilan, and Zachary Sheaffer. Employee motivation, emotions, and performance: a longitudinal diary study. *Journal of Managerial Psychology*, 34(6):415–428, 2019.

[86] Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *Advances in Neural Information Processing Systems*, 35:18343–18362, 2022.

[87] Ji Wang, Jian Zhao, Sheng Guo, Chris North, and Naren Ramakrishnan. Recloud: semantics-based word cloud visualization of user reviews. In *Graphics Interface 2014*, pages 151–158. AK Peters/CRC Press, 2020.

[88] Tamara Lopez, Thein Tun, Arosha Bandara, Levine Mark, Bashar Nuseibeh, and Helen Sharp. An anatomy of security conversations in stack overflow. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, pages 31–40. IEEE, 2019.

[89] Maria Papoutsoglou, Johannes Wachs, and Georgia M Kapitsaki. Mining dev for social and technical insights about software development. In *2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR)*, pages 415–419. IEEE, 2021.

[90] Danny Azucar, Davide Marengo, and Michele Settanni. Predicting the big 5 personality traits from digital footprints on social media: A meta-analysis. *Personality and individual differences*, 124:150–159, 2018.

[91] Euna Mehnaz Khan, Md Saddam Hossain Mukta, Mohammed Eunus Ali, and Jalal Mahmud. Predicting users' movie preference and rating behavior from personality and values. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 10(3):1–25, 2020.

[92] Zhe Liu, Anbang Xu, Yi Wang, Jerald Schoudt, Jalal Mahmud, and Rama Akkiraju. Does personality matter? a study of personality and situational effects on consumer behavior. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, pages 185–193, 2017.

[93] Henry Hsu and Peter A Lachenbruch. Paired t test. *Wiley StatsRef: statistics reference online*, 2014.

[94] Yonas Woldemariam. Assessing users' reputation from syntactic and semantic information in community question answering. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5383–5391, 2020.

[95] Dana Movshovitz-Attias, Yair Movshovitz-Attias, Peter Steenkiste, and Christos Faloutsos. Analysis of the reputation system and user contributions on a question answering website: Stackoverflow. In *Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 886–893, 2013.

[96] Irina Cojuharenco and Natalia Karelaia. When leaders ask questions: Can humility premiums buffer the effects of competence penalties? *Organizational Behavior and Human Decision Processes*, 156:113–134, 2020.

[97] Fabio Calefato, Filippo Lanubile, and Nicole Novielli. How to ask for technical help? evidence-based guidelines for writing questions on stack overflow. *Information and software technology*, 94:186–207, 2018.

[98] Anton Barua, Stephen W Thomas, and Ahmed E Hassan. What are developers talking about? an analysis of topics and trends in stack overflow. *Empirical Software Engineering*, 19:619–654, 2014.

[99] Wenhan Zhu, Haoxiang Zhang, Ahmed E Hassan, and Michael W Godfrey. An empirical study of question discussions on stack overflow. *Empirical Software Engineering*, 27(6):148, 2022.

[100] Shaowei Wang, Tse-Hsun Chen, and Ahmed E Hassan. How do users revise answers on technical q&a websites? a case study on stack overflow. *IEEE Transactions on Software Engineering*, 46(9):1024–1038, 2018.

[101] Yla R Tausczik and James W Pennebaker. Predicting the perceived quality of online mathematics contributions from users' reputations. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1885–1888, 2011.

[102] Mondal Saikat, Mohammad Masudur Rahman, Chanchal K Roy, and Kevin Schneider. The reproducibility of programming-related issues in stack overflow questions. *Empirical Software Engineering*, 27(3), 2022.

[103] Pradeep Kumar Roy, Jyoti Prakash Singh, and Amitava Nag. Finding active expert users for question routing in community question answering sites. In *Machine Learning and Data Mining in Pattern Recognition: 14th International Conference, MLDM 2018, New York, NY, USA, July 15-19, 2018, Proceedings, Part II 14*, pages 440–451. Springer, 2018.

[104] Zhang Zhang, Xinjun Mao, Yao Lu, Shangwen Wang, and Jinyu Lu. An empirical study on the influence of social interactions for the acceptance of answers in stack overflow. In *2020 27th Asia-Pacific Software Engineering Conference (APSEC)*, pages 425–434. IEEE, 2020.

[105] Ronald S Burt. The network structure of social capital. *Research in organizational behavior*, 22:345–423, 2000.

[106] Faheem Ahmed, Luiz Fernando Capretz, Salah Bouktif, and Piers Campbell. Soft skills and software development: A reflection from the software industry. *arXiv preprint arXiv:1507.06873*, 2015.

[107] Carlos Tam, Eduardo Jóia da Costa Moura, Tiago Oliveira, and João Varajão. The factors influencing the success of on-going agile software development projects. *International Journal of Project Management*, 38(3):165–176, 2020.

[108] Scott Seibert, Jos Akkermans, and Cheng-Huan Liu. Understanding contemporary career success: A critical review. *Annual Review of Organizational Psychology and Organizational Behavior*, 11:509–534, 2024.

[109] Örjan de Manzano and Fredrik Ullén. Domain specific traits predict achievement in music and multipotentiality. *Intelligence*, 89:101584, 2021.

[110] Christian Montag and Jon D Elhai. A new agenda for personality psychology in the digital age? *Personality and Individual Differences*, 147:128–134, 2019.

[111] Seren Başaran and Obinna H Ejimogu. A neural network approach for predicting personality from facebook data. *Sage Open*, 11(3):21582440211032156, 2021.

[112] Chidera Victoria Ibeh, Onyeka Franca Asuzu, Temidayo Olorunsogo, Oluwafunmi Adijat Elufioye, Ndubuisi Leonard Nduubuisi, and Andrew Ifesinachi Daraojimba. Business analytics and decision science: A review of techniques in strategic business decision making. *World Journal of Advanced Research and Reviews*, 21(02):1761–1769, 2024.

[113] Rodrigo Smiderle, Sandro José Rigo, Leonardo B Marques, Jorge Arthur Peçanha de Miranda Coelho, and Patricia A Jaques. The impact of gamification on students' learning, engagement and behavior based on their personality traits. *Smart Learning Environments*, 7(1):3, 2020.

[114] Akif Ahmed, Md Saddam Hossain Mukta, Fardin Muntasir, Shaguffta Rahman, AKM Najmul Islam, and Mohammed Eunus Ali. Can covid-19 change the big5 personality traits of healthcare workers? In *Proceedings of the 7th international conference on networking, systems and security*, pages 12–17, 2020.

[115] Chanchal Suman, Sriparna Saha, Aditya Gupta, Saurabh Kumar Pandey, and Pushpak Bhattacharyya. A multi-modal personality prediction system. *Knowledge-Based Systems*, 236:107715, 2022.

[116] Harshit Bhardwaj, Pradeep Tomar, Aditi Sakalle, and Wubshet Ibrahim. Eeg-based personality prediction using fast fourier transform and deeplstm model. *Computational Intelligence and Neuroscience*, 2021:1–10, 2021.

[117] Yuliang Zhao, Hualin Xing, Xiaoai Wang, Yu Tian, Tingting Sun, Meng Chen, Dannii YL Yeung, Samuel MY Ho, Jianping Wang, and Wen Jung Li. Wearable internet of things gait sensors for quantitative assessment of myers–briggs type indicator personality. *Advanced Intelligent Systems*, 6(3):2300328, 2024.

[118] Le Vy Phan and John F Rauthmann. Personality computing: New frontiers in personality assessment. *Social and personality psychology compass*, 15(7):e12624, 2021.