Popularity of Programming Languages using Stack Overflow Data

Aditya Panchal
A. D. Patel Institute of Technology, India

Abstract— Stack Overflow [1] is a software developer's most popular Q&A platform. The questions posed in Stack Overflow usually contain the code snippet as a medium for sharing knowledge and learning. Stack Overflow users are responsible for properly tagging a question's programming language and assume that the programming language of the snippets is the same as the tag itself. A programming language analysis of questions posted on Stack Overflow is being suggested in this study using Natural Language Processing (NLP) and Machine Learning (ML). The best performance is given by the application of ML software in mixing text and code snippets for a query. These results show that a fragment of only certain lines of source code can be defined in the programming language. We visualize the programming languages feature space to determine the properties of the information within the language-related questions.

Keywords: Natural Language Processing (NLP), Stack Overflow, Machine Learning (ML)

I. INTRODUCTION

Stack Overflow has been used widely in the development of technology over the last decade. Inexperienced developers today rely on Stack Overflow to answer their software development concerns. Through Stack Overflow development there have been changes in the number of programming languages in use. Stack Overflow lists 38 languages in its most common, afraid, and wanted list in its 2018 Developer's Survey. This list lists 100 languages TIOBE Programming Language [2].

Forums such as Stack Overflow use query labels to suit users who can address them. Nevertheless, new users may not correctly mark their posts in Stack Overflow or beginner developers. This leads moderators to vote down and tag the comments, even if the issue is valid and gives the community added value. In some cases, concerns about stack overflow relevant to language programming may lack a language mark for programming. For example, Pandas is a popular Python library that provides data structures and powerful analysis tools but typically does not have a Python tag in its Stack Overflow Questions. This might lead to confusion among developers who are new to the language of programming and who do not know all its common libraries. When comments are immediately tagged with the corresponding programming languages, the issue of missing language tags could be solved.

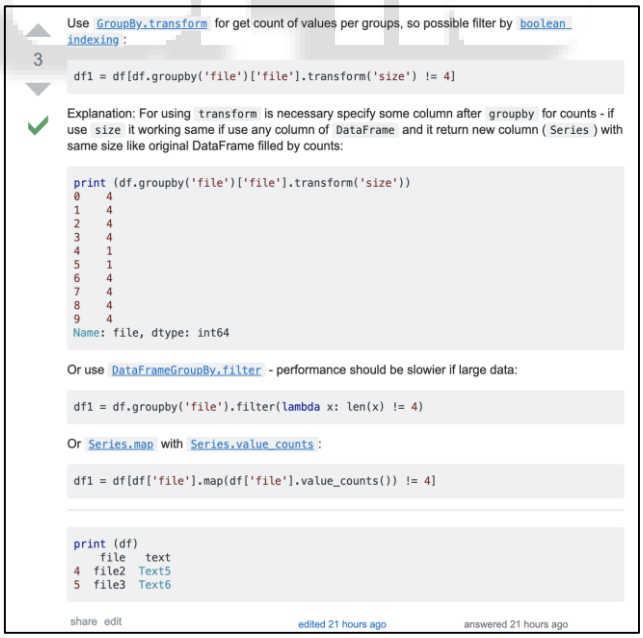
Another problem with this is the description of the fragment script programming language. With any lines of code, the language in which they are written also needs to be identified. Stack Overflow relies on a question tag to determine how any snippet is to be typed. If the query is not labeled with the programming language, the code is not labeled. Furthermore, the code for the various syntactic structures in the language is displayed in different colors, if a tag is applied. Stack overload requires only the first programmable language mark for typing all the fragments of the query if a question has samples of two or more languages. There is a significant amount of language text information available to explain and address code snippet(s)/ programming language in natural languages. Text data may be used as a blog, code documentation, bug report, code observation or a snippet summary.

Sites such as Github, Jira, Bugzilla and Stack Overflow or Quora provide text data that describes and addresses fragments of code or languages in their software. Stack Overflow articles are an example of language textual data. Many stack overflow posts include a code snippet or text or textual information. This information might be exploited to improve the prediction of programming language. It could be used without using a snippet to predict the language.

Stack Overflow is an example of textual data in both the name and code structure. Users post questions to address their issues or to check for information about a specific language in programming. The post can contain texts only without a code snippet. This work aims to examine textual information to forecast the programming language.

This thesis is used to simulate the programming languages in Stack Overflow from documents, code snippets, and the synthesis of textual and computer snippeting data in Machine Learning (ML), and Natural Language Processing (NLP) systems.

A. *Example of Stack Overflow post:*



II. RELATED WORK

A. *Mining Stack Overflow:*

The Stack Overflow Dataset is the most popular research platform to study and understand developer discussions. Barua et al, using Latent Dirichlet Allocation (LDA) and quantitative the modeling methods in stack overflow post

analysis, have discussed key issues in the developer community. Rosen has examined the app developers ' concerns. To explore which questions have been answered well and why other questions are not answered, Treude studies how developers pose and answer questions to social media by using Stack Overflow as a database to evaluate and answer messages. Morrison studied how age-related programming knowledge was used to answer their research questions with the data from Stack Overflow. Nachi analyzed what a good or bad post is based on the number of code columns, code reliability and text on a stack overflow post to consider variables that differentiate a good post from a bad post. Another method analyzed in Stack Overflow was Denzil [4] The addressed and unanswered questions. They suggested that a classifier predict how long a question post takes to receive a response. Since there is a large number of questions in Stack Overflow, some users are asking questions on Stack Overflow before looking for them. It's a duplicate question that already exists in Stack Overflow. M. The technical classification to mine duplicated questions were suggested by Ahasanuzzaman [3] Gustavo conducted an empirical study to explain how software developers speak in technology-community energy issues. In this analysis, the Stack Overflow database was used and over 300 questions and 550 answers have been analyzed.

B. Predicting Programming languages:

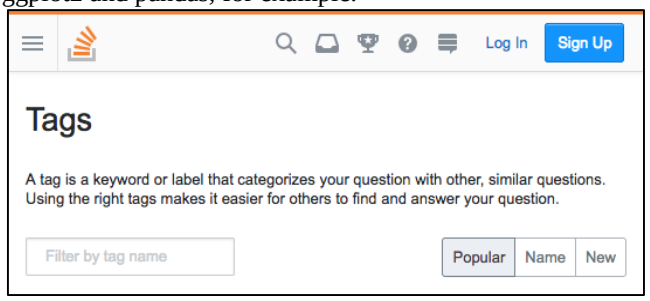
Kennedy looked at the issue of using natural language recognition to classify the programming language of all GitHub source code files (rather than Stack Overflow questions). Our identification is based on five NLP numerical language models, defining 19 programming languages, and achieving 97.5 percent accuracy. Khasnabish has suggested a method to use source code files to identify 10 programming languages. The system with Bayesian learning strategies was learned and evaluated using four algorithms, i.e. NB, Multinomial Naive Bayes (MNB) and Bayesian Network (BN). The highest accuracy of MNB was shown to be 93.48%.

Baquero suggested a category to predict the programming language of the Stack Overflow problem. They gathered 18,000 questions for each of the 18 languages of programming in Stack Overflow, including fragments of text and code, 1000 questions. They trained two classifiers on two different datasets, text bodies, and code snippet features, using a Support Vector Machine model. Many editors including Sublime and Atom add to the application highlights depending on the language of the software. However, explicit extensions are required e.g .. .html,.css,.py. Portfolio[16] is a search engine that helps programmers find functions that allow high-level query requirements.

III. WORKFLOW

A. Data on the Tags:

Stack Overflow, a programming question, and answers with over 16 million programming questions is an outstanding source of data. Through calculating how many questions that methodology has, we can get a realistic idea of how many people use it. To assess the relative popularity of languages like Como R, Python, a language and a language script we're going to use open data from Stack Exchange Data Explorer over time. Any query about stack overflow has a label that defines the concept of engineering. There is a tag for languages such as R and Python, or applications such as ggplot2 and pandas, for example.



B. Data Preprocessing:

Data Cleaning routines work to clean the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies, it is one of the most pivotal tasks to achieve the accuracy and thereby we will clean the data. Furthermore, data integration will combine the data into a coherent data store. Using transformation techniques the data will be transformed and consolidated into appropriate forms and once that is completed, the data will be reduced for the representation of data into smaller volumes, yet will produce more or less the same analytical results.

C. Visualize Change Over Time:

Data visualization is the information and data graphical representation. Data visualization applications provide an accessible way of seeing and observing trends, backgrounds and patterns in the information by using visual elements including diagrams, charts, and maps. Data visualization tools and technologies are essential in the big data world to analyze large quantities of data and to make choices based on data. The visualization of information is another type of visual art that captures our attention and looks at the text. We see trends and outliers fast when we see a chart. We will easily internalize something if we see it. It's purpose storytelling. You know how much more powerful visualization can be if you have ever looked at a big data sheet and could not see a pattern. It is considered as a branch of describing statistics by some, but also as a tool of theoretical development of others. Data visualization is both an art and a science. Increasing information volumes provided by the Internet and an increasing number of environmental sensors are referred to as "big data" or the Web of stuff. Data visualization has ethical and analytical challenges in the processing, analysis and communication of this data and data scientists have been named for the field in data science. The analysis of the terminology and space available for two programs is also an important contribution to this research. System Word2Vec was used to view code snippets features, text data sets (title and body) with Gensim, a vector space simulation Python framework. Concepts cannot be seen in such a big space so that T-SNE [5] has been used to decrease the number of steps to 2. The most frequent 3% of words were selected from the vectors for various programming languages and analyzed using word similarity and cosine distance.

D. Classification of Data:

The Random Forest Classifier and XGBoost ML algorithms (a booster algorithm for gradient) were used. Some algorithms were much more reliable than others, like ExtraTree and MultiNomialNB. These were more accurate. In this thesis, precision, recall, precision, F1 score, and confusion matrix are the performance metrics used for classifiers.

1) Random Forest Classifier (RFC)

RFC is a collective algorithm combining more than one classification. This classifier generates a sequence of decision-making structures from randomly selected learning datasets. That sub-set contains a decision tree to decide on the final decision. Another advantage of this classifier is that, when another or several trees make a noise or error, the reliability of the test will not be compromised. The total number of forest trees is of the utmost importance because a large number of forest trees provide very precise information.

2) XGBoost

Extreme Gradient Booster (XGBoost) is an RFC-like tree-based model. The goal was to change a poor pupil to be a better pupil. Note, Random Forest is a simple ensemble algorithm that produces various sub-trees and each tree independently predicts the output. Nevertheless, XGBoost is smarter because every subtree sequentially generates the prediction. Every subtree thus learns from the mistakes of the previous subtree. XGBoost's idea was a gradient boost, but XGXoost uses a regularized model to control the overlay and improve performance. Random SearchCV, a tool in the Scikit-learn library for parameter search, has been used for the machine learning models. The XGBoost algorithm contains several parameters including minimum child weight, max length, regularization of L1 and L2 and measurement steps such as Receiver Operating Characteristic(ROC), precision and F1. RFC has the factor of the number of calculators used to match a prototype and it is a bagging type. The parameters of the models should be modified by different ones. Factor tuning using a strategy like a grid scan is however computationally expensive. Therefore, Random Search (RS) tuning is a method of in-depth learning. After RS tuning all design parameters have been defined on the cross-validation sets.

3) Performance Matrix

The classifier is evaluated on an undefined dataset to determine its output once it learns from the derived functions. In the confusion matrix, each post is included in the test data set to describe how the messages are forecast. Precision measures the number of posts predicted to a specific programming language. Remember how many related posts are projected for a certain programming language. The example of one language demonstrates the metrics of efficiency.

True Positive: posts related to a language and predicted to be a language.

False Positive: posts not related to a language and predicted to be a language.

False Negative: posts related to a language but not predicted to be a language.

True Negative: posts not related to a language and not predicted to be a language.

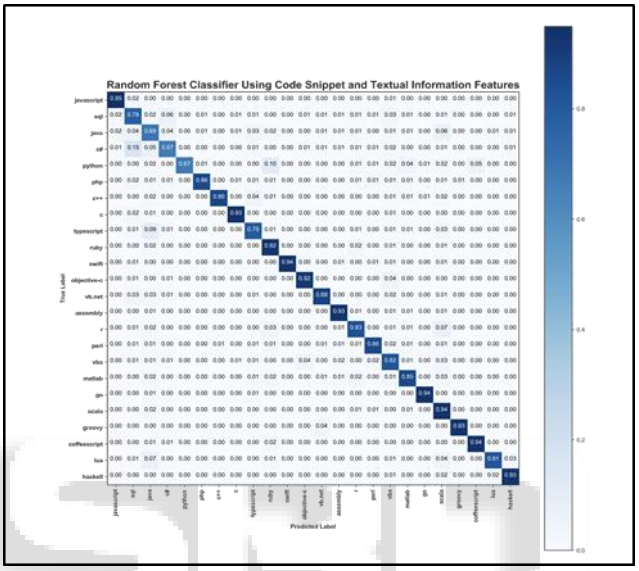
$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$Recall = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$F1Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

E. Result:

In contrast to XGBoost Classifier, Random Forest reaches much lower accuracy. Accuracy decreased to 74.5% from 86.3%.



The confusion matrix focused on snippet and term data features for the Random Forest Classifier. The diagonal is the proportion of the correctly estimated programming language.

Programming	Precision	Recall	F1-score
Java	0.61	0.69	0.65
C++	0.98	0.86	0.92
JavaScript	0.92	0.95	0.94
Swift	0.96	0.94	0.95
C#	0.78	0.67	0.72
SQL	0.69	0.78	0.73
Python	0.99	0.67	0.80
Assembly	0.91	0.93	0.92
Ruby	0.81	0.92	0.86
MATLAB	0.87	0.85	0.86
R	0.86	0.83	0.85
Go	0.96	0.94	0.95
C	0.94	0.93	0.94
Objective-C	0.90	0.92	0.91
Vb.Net	0.89	0.88	0.89
Perl	0.89	0.86	0.87
Lua	0.94	0.81	0.87
TypeScript	0.82	0.79	0.81
PHP	0.94	0.91	0.93

RFC quality trained in textual and code snippet data.

F. Features of Some Programming Languages:

The machine learning algorithm extracts characteristics from each programming language during the training stage to help the model learn to predict a new post's programming language. For programming languages, most features are very common. The ' class ' is one of the key features for 14 different programming languages, such as C, JavaScript, Java, C #, Python, PHP, C++, TypeScript, Ruby.

Programming Languages	Features
JavaScript	alert, body, class, click, content, data, div, document, else, false, for, form, function, getelementbyid, height, href, html, http, id, if, img, input, is, javascript, jquery, js, length, li, name, new, onclick, option, return, script, span, src, style, td, text, the, this, title, to, tr, true, type, value, var, width, window
SQL	as, begin, by, case, count, create, date, datetime, dbo, declare, default, desc, end, for, from, group, id, if, in, inner, insert, int, into, is, join, key, left, name, not, null, on, or, order, primary, select, set, sql, sum, t1, table, the, then, to, type, update, value, values, varchar, when, where
Java	add, apache, at, catch, class, com, exception, false, file, final, for, http, id, if, import, in, int, is, java, javax, lang, list, main, method, name, new, null, object, of, org, out, println, private, property, public, return, source, static, string, sun, system, the, this, to, true, try, type, value, void, xml
C#	add, asp, at, bool, byte, class, console, data, else, false, foo, for, foreach, from, get, id, if, in, int, is, item, list, name, new, null, object, of, private, public, return, select, sender, server, set, static, string, system, text, the, this, to, tostring, true, type, using, value, var, void, where, writeline
Python	__init__, and, class, data, db, def, django, error, file, foo, for, from, http, id, if, import, in, is, lib, line, models, module, name, none, not, object, of, open, os, packages, path, print, py, python, python2, request, return, self, site, sys, test, text, the, this, to, true, type, user, usr, value
C++	and, bool, boost, char, class, const, cout, cpp, data, double, else, end, endl, error, file, foo, for, function, if, in, include, int, is, it, main, name, new, null, of, operator, private, public, return, size, std, string, struct, template, test, the, this, to, type, typename, unsigned, using, value, vector, virtual, void
PHP	_post, and, array, as, br, class, com, content, data, div, echo, else, file, for, from, function, html, http, id, if, in, input, is, name, new, null, of, option, php, public, query, result, return, row, select, string, td, test, text, the, this, title, to, true, type, url, user, value, where, www
C	and, argc, argv, array, break, buffer, case, char, const, count, data, define, double, else, error, exit, file, for, from, function, if, in, include, int, is, list, long, main, malloc, name, next, node, null, of, printf, return, size, sizeof, stdio, string, struct, temp, the, this, to, typedef, unsigned, value, void, while
Ruby	activerecord, and, base, bin, class, com, def, div, do, each, end, error, file, foo, for, from, gem, gems, html, http, id, if, in, include, is, lib, library, local, name, new, nil, params, post, puts, rails, rake, rb, require, ruby, rubygems, self, string, test, the, to, true, type, user, users, usr

IV. CONCLUSION

This work addressed the major problem of predicting code snippets and text information in programming languages. The focus was on predicting the Stack Overflow programming language. The results show that the classifier learning and evaluation by mixing text data with a code snippet is as accurate as 91,1%. Specific tests using text data and texts, achieved accuracies of 81.1% and 77.7% respectively. It means that knowledge from textual features can be learned better than information from software snippet features for a machine learning system. We assume that this category can be used in many contexts, such as code and fragment management tools.

REFERENCES

[1] https://stackoverflow.com/
[2] Tiobe programming community index. https://www.tiobe.com. Accessed: 2018-04-30.
[3] M. Ahasanuzzaman, M. Asaduzzaman, C. Roy, and K. Schneider. "Mining duplicate questions of stack overflow". In IEEE Working Conference on Mining Software Repositories, pages 402–412, 2016.
[4] D. Correa and A. Sureka. "Chaff from the wheat: Characterization and modeling of deleted questions on stack overflow". In ACM International Conference on World wide web, pages 631–642, 2014.
[5] L. Maaten and G. Hinton. "Visualizing data using t-sne". Journal of Machine Learning Research, 9:2579–2605, 2008.
[6] https://pdfs.semanticscholar.org/7dae/e9ebce9753ba914166d61830306918ced58b.pdf
[7] https://www.researchgate.net/publication/274572185_Comparative_Studies_of_Six_Programming_Languages
[8] https://www.irjet.net/archives/V4/i12/IRJET-V4I1266.pdf
[9] https://web.cs.ucdavis.edu/~filkov/papers/lang_github.pdf

