Identifying Community-Supported Technologies and Software Developments Concepts by K-means Clustering

Farag Almansoury, Segla Kpodjedo, Ghizlane El Boussaidi
Department of Software Engineering and Information Technology, École de Technologie Supérieure(ETS)
University of Quebec, Montreal, QC, Canada

***Abstract*—Working on technologies that have community support is one of the most important factors in software development. Software developers often face difficulties during software development, and community support from other software developers help them significantly. This paper presents an approach based on K-mean clustering technique to identify the level of community support for software technologies and development concepts using Stack Overflow discussion forums. To test the approach, a case study was performed by gathering data from SO and preparing a dataset that contains over a million of Java developers' questions. Then, K-mean clustering was applied to identify the community support levels. The goal is to find the best features that group community-supported software technologies and development concepts and identify the number of groups to determine the community support levels. Statistical error, clustering and classification evaluation metrics were applied. The results indicate that the best features to formulate community supported technologies and development concept levels are Failure Rate and Wait Time. The results show that the approach identifies two groups of community supported and development concept levels based on the best silhouette index value of 97%. According to the results the majority of Java technologies and development concepts are labeled with less community supported technologies and development concepts (Cluster 2). Random Forest classifier was applied to indirectly evaluate the approach to detect the identified community support class. The result shows that RF classifier presents a good performance and shows high accuracy value of 99.49% which indicates that the identified groups improve the performance of the classifier. The approach can be utilized to assist software developers and researchers in utilizing the SO platform in developing SO-based recommendation systems.**

***Keywords*—Stack overflow; unsupervised machine learning; k-means clustering; empirical study; machine learning; random forest; software development; Java; classification; community support**

I. Introduction

Developing a software from scratch with standard libraries is no longer a viable option for most meaningful software projects. Thus, some of the key decisions for a software project are about which technology to turn to, or which APIs or projects to depend on. Choosing the right technology is very important as it can significantly impact a project's quality and velocity. Depending on the particularities of their projects, software developers may have to sift through a wide range of rapidly evolving technologies (be it frameworks or libraries) across various platforms (PC, Mobile, Web). To inform their decision, they try to get guidance from online articles or blogs about which technologies are the best. For example, a developer may be looking for an IDE for his project and end up on a website or a forum post about the "10 Best IDE Software"[1]. These online resources, though valuable, often provide opinion-driven commentary, sometimes informed by the experience of a single writer or blogger. Moreover, they run the risk of being outdated, given the fast pace of many technologies. For up-to-date, interactive discussions, developers sometimes turn to Stack Overflow (SO), the leading Q&A website for software development. However, their questions about API or technology recommendations are systematically dismissed as seeking opinions[2], which is explicitly banned by Stack Overflow. For example, "What IDE to use for Python?"[3], "What good, C++ programming IDE is available for Linux?"[4], "What is the best IDE for PHP?"[5], and "What programming language to create and format book?"[6]

Community-supported technologies used among software developers are important and help speed up software productivity. For this reason, we have turned to SO since it contains big data from software developers' discussions. SO has more than 20,000,000 questions related to different topics dealing with developers' issues in different domains and platforms. Hence, it become a target to developers since it enables them to find solutions for their problems. SO has also been utilized by researchers to carry out their studies [1], [2], [3], [4], [5]. Software developers and researchers are trying to find a mechanism to organize the data to facilitate and speed up the search processes and find the appropriate answers to the questions raised [6]. Thus, we turned to SO to identify which technologies and software development concepts have community support.

This study aims to provide an approach that help software developers and the software engineering research community to identify the technologies and development concepts that have the most and least community support by leveraging SO and the unsupervised machine learning technique k-means clustering. The study aims to answer the two following questions:

[1] https://websitesetup.org/best-ide-software/

[2] There are good reasons to ban opinion and recommendation-seeking questions on a Q&A forum as they may devolve into never-ending discussions or be fodder for bitter arguments about which technologies are the best. Moreover, entities behind those technologies may participate and recommend their own products.

[3] https://stackoverflow.com/questions/81584

[4] https://stackoverflow.com/questions/24109

[5] https://stackoverflow.com/questions/116292

[6] https://stackoverflow.com/questions/68275077

RQ1: What are the features that identify clusters of technologies and software development concepts that have community support?

RQ2: What is the clusters quality and if the identified clusters consistent to increase the ability to differentiate between community-supported technologies and development concepts?

To gain more insight into the potential application of the approach, it has been applied to all Java-related postings on SO from 2014 to 2021 and studied the distribution of technologies and development concepts among community support levels. We chose to concentrate on Java ecosystem to demonstrate that the technique is applicable. If the approach is effective, it is simpler to generalise it to a wider range of domains and technologies than to determine that a technique designed for multiple domains at once would not work.Java ecosystem was utilized as a case study since Java is one of the programming language that is gaining the most traction among software developers on SO.

To apply the approach and analyze data, we turned to the most useful and popular library for machine learning in Python Scikit-learn [7] and the full-featured AI and ML integrated tool that supports multiple scripting languages and is easy to work with huge datasets Tableau [8]. Scikit-learn ML is a high-level API built on data frames and datasets that allows pipelines and is easier to build. Tableau features an analytics pane with drag-and-drop machine learning that allows us to forecast future outcomes based on historical data, discover future trends for your data using multiple models, or understand the relationships between data points using clustering. Cluster validity technique for the k-means clustering algorithm had already been proposed in the literature, thus, statistical error techniques the sum of squares within each cluster (WSS) and the sum of squares between clusters (BSS) in addition to and silhouette index were used.

The paper's primary contributions can be summarised as follows:

- The novel approach introduced in this paper can be used as a decision support based on K-Mean Clustering, for community-supported techniques and development concepts detection.
- We built a clustering model to identify community-supported software technologies and development concepts level of community support based on new features namely Failure rate (FR) and wait time (WT).

The rest of the paper is organized as follows: Section II introduces the theoretical background. Section III reviews the literature related to the study. Section IV outlines the approach overview and its application on Java ecosystem. Section V evaluates the approach based on clustering evaluation techniques. Section VI provides results and discussion. Section VII provides concluding remarks. Finally, Section VIII provides the future work.

II. Theoretical Background

A. Stack Overflow and its Tagging System

"Stack Overflow (SO) is a question and answer website for amateur programmers and professionals programmers"[7]. It is a privately owned website that was established in 2008 by Atwood and Spolsky. Users are encouraged to participate as they can earn points towards their reputation and other privileges (e.g. editing), for being actively involved on the site. Users can vote for questions and answers, both upvotes (positive feedback) and down votes (negative feedback) are allowed. The number of the upvotes minus the number of downvotes represent the score.It is the largest Software development Q&A community, according to the SO Annual developer survey 2019[8] which reported that SO had 80 million visit, of which 25% are developer professionals and university students, and more than 80% rely on SO for educational purposes and also 65% of Stack Overflow's professional developers contribute to open source projects.

Tagging System: According to Stack Overflow's tagging system, a question must have between one and five tags. A tag is a single word or compound words (for example, WebGL, vertex-shader, respectively) that define the technical term at the centre of the question [9] (see Fig. 1 for an example). Tags on Stack Overflow include a broad variety of technical terminology [10], [11], from definitions to programming languages, IDE, frameworks, libraries/tools/, and individual APIs (at class or module level). Researchers frequently use these tags as a starting point for investigating the issues addressed on SO.

B. Clustering

Clustering is the breaking down of a set of data or objects into a number of clusters. Each cluster consists of a group of similar facts that behave identically. Clustering is equivalent to classification, except that the classes in clusters are not defined and determined in advance, and data grouping is performed with no supervision [12], [13]. Different techniques used for clustering include partitioning based, hierarchical, density and grid [14]. K-means [15], which is the most basic and widely used partitioning procedure among scientific clustering algorithms [16], [17] was utilized.

C. Classification

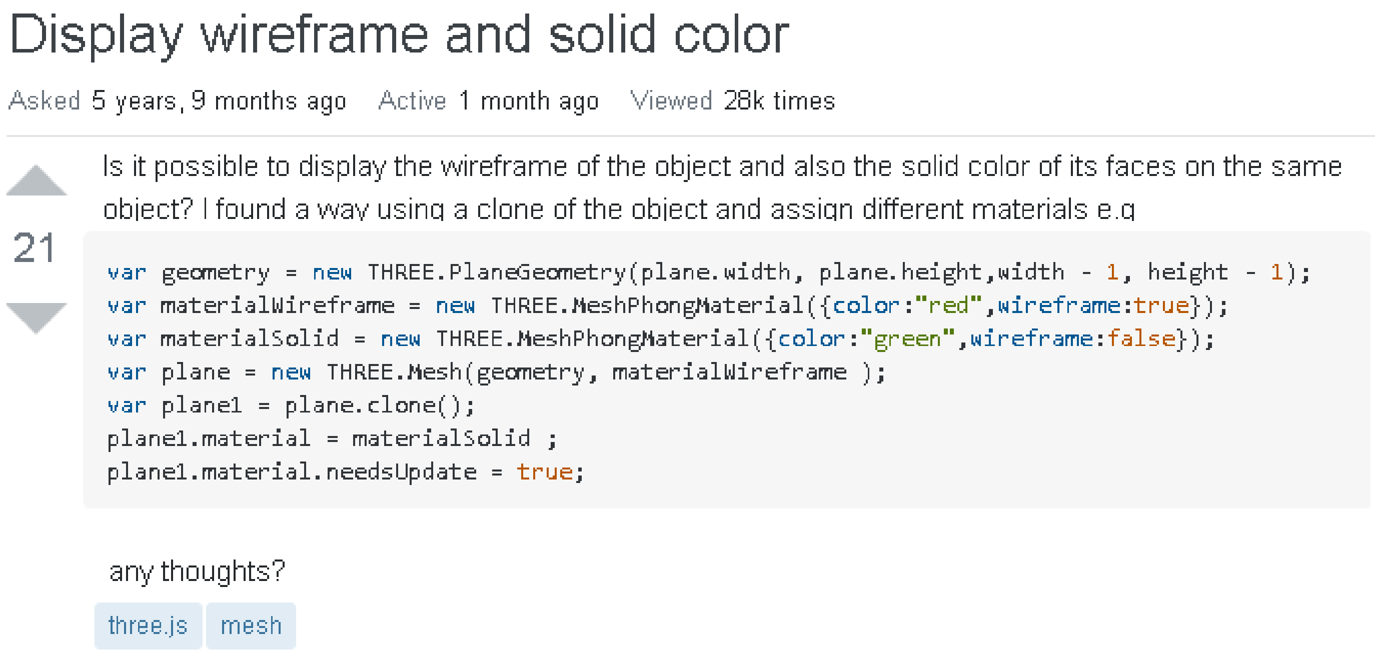
Classification is an important aspect of data mining as a technique for forecast modelling. Simply put, classification is the process of breaking down data into dependent or independent categories [18]. Based on previous decisions, classification is utilised to make some future decisions. Different techniques used for classification include Random Forest, support vector machine,decision tree learning, neural networks, nearest neighbour, and Naves Bayes method [19]. In the experiment, the Random Forest classifier was employed.

III. Related Work

In recent years, Stack Overflow questions and answers have been the topic of extensive research. One of the goals of

[7] https://stackoverflow.com

[8] https://insights.stackoverflow.com/survey/2019



Display wireframe and solid color

Asked 5 years, 9 months ago Active 1 month ago Viewed 28k times

Is it possible to display the wireframe of the object and also the solid color of its faces on the same object? I found a way using a clone of the object and assign different materials e.g

```
var geometry = new THREE.PlaneGeometry(plane.width, plane.height,width - 1, height - 1);
var materialWireframe = new THREE.MeshPhongMaterial({color:"red",wireframe:true});
var materialSolid = new THREE.MeshPhongMaterial({color:"green",wireframe:false});
var plane = new THREE.Mesh(geometry, materialWireframe );
var plane1 = plane.clone();
plane1.material = materialSolid ;
plane1.material.needsUpdate = true;
```

any thoughts?

Fig. 1. Post Example ob Satck Overflow.

these studies is to track developers' interest in various topics and how it evolves over time, as well as their relationship to current technology trends[10], [20]. The majority of research also emphasizes how challenging it is to maintain the quality of SO Q&A [11].

Stack Overflow has received a lot of attention from the research community in the recent years. The rapid increase in the number of studies are a result of two main reasons: 1) the influx of new technologies that generated discussions on Q&A forums/websites; and 2) the increased use of these technologies by software developers due to their capability of solving problems, knowledge sharing, and learning.

The popularity of SO and its sheer volume of questions and answers have made it a platform of interest for research on specific areas such as mobile development [1], web development [21], [22], [3],web 3d [23], security [24], [4], [25]. In [21], for instance, used data from Stack Overflow to obtain a better understanding of the challenges faced by web developers. Their results show there was an increase in the number of questions related to web-development, concurrently with a downtrend for cross-browser related posts. In particular, [21] used data from SO to get a better understanding of the challenges faced by web developers. It extracted questions tagged with JavaScript, HTML5, CSS and found that cross browser issues were trending down. Another study in [22] investigated web developers' concerns pertaining to Web APIs. It found that ”known issue/bug” is a dominant topic of discussion, and observed that discussions are majoritarily (three times out of four) about occasional concerns that disappear quickly, which would suggest that “*Web API providers tend to timely address most problems encountered by client developers*”. Finally, [3] focused on popularity and difficulty of issues related to the web frameworks Laravel and Django and found that half the issues are shared by both, with installation being a popular but difficult issue for both. The study by [9] reported that the key technologies that the question is about can typically be deduced from the question tags. These research studies are the basis for the approach, which uses Stack Overflow's crowd sourced expertise to answer information needs in technology community support inquiries.

IV. The Approach Overview and its Application on Java Ecosystem

The aim of this research consists of two parts, the first part is to what extent we can leverage K-means clustering to distinguish and group the community-supported technologies and development concepts based on the stack overflow platform to identify Which features contribute significantly to the the clusters formation. The second goal is to examine to which extent is the discovered community supported level consistent enough to increase the ability to distinguish between community-supported technologies and development concepts? it can be used as a detection tool based on machine learning classifier model. Fig. 2 introduces the approach that starts from select targeted technologies to identify the community supported technologies and development concepts. Then data from the questions tagged with the targeted technologies was extracted after the topics based on tags co-occurrence with the target technologies were grouped. We then identified the most important features to be used as an input to the model. Later, the K-mean Clustering algorithm was applied and the model was evaluated to check the quality of the resulting clusters based on the classification and evaluation-clustering quality techniques. In the next section, we demonstrated the approach on 1297109 Java questions asked by Java developers as a case study to examine it.

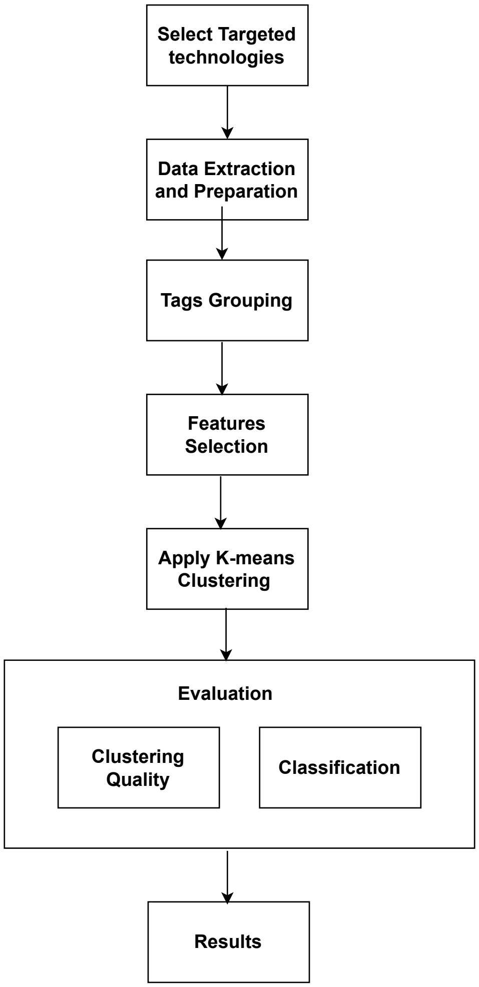


Fig. 2. An Approach to Identify Community Supported Techniques and Development Concepts.

A. Data Extraction and Preparation

In this step, data preparation refers to reprocessing the dataset for the modelling phase. We present a dataset obtained by analyzing 1297109 questions from the SO platform. The data set used in this study includes all Java discussions on SO. Over 782 topics of the million questions found, are unique to Java ecosystem. Since we are interested in identifying community-supported technologies and development concepts of the java ecosystem, hence tags with more than 30 questions were included. Table I shows data set information. Table II present the list of features to be used during clustering should be revised.

TABLE I. General Information of the Data-Set Used

Data set info	Items	Description
Java Questions	1297109	All questions related to the java ecosystem on SO from 2015 to 2021
All paired tags (Topics)	980	All co-occurrence topics with java questions related to software technologies and development concepts
paired tags >=30 Questions	794	All co-occurrence topics with java questions that have at least 30 questions as threshold related to software technologies and development concepts

TABLE II. Features and Measures.

Features and metrics	Description
Questions	The number count of the questions related to tag x.
Tag Name	keywords provided for the questions by developers that define the technical term at the center of the question.
Views	Number of views for the question, extracted by metadata ViewCount attribute of the post.
Score	Number of upvotes minus number of downvotes, extracted by metadata upvotes and downvotes attributes of the post.
Favorite	Number of Favorites For the question, extracted by metadata Favorites Count attribute of the post.
Comment	Number of comments For the question, extracted by metadata CommentCount attribute of the post.
Answers	Number of answers For the question, extracted by metadata AnswerCount attribute of the post.
Failure rate (FR)	The percentage of questions that do not have an accepted answer.
Wait Time (WT)	The median time for satisfactory answers (in these cases where the question got an answer that its asker accepted).

B. K-mean Clustering to Identify Supporting Level

Clustering analysis is a well-known concept in the field of data mining [26].Clustering is a popular method for grouping data based on shared patterns or similarities. Numerous applications [27], [28], including science, technology, biology, social science economics, medicine, smart farming, geospatial, stock market, and many more, have made extensive use of cluster analysis.

In this paper k-means clustering was performed. The K-Means algorithm is an unsupervised learning approach for classifying/grouping objects based on their features. The technique splits the data into k clusters for a specified number of clusters k. Each cluster has a centre (centroid), which is defined as the mean value of all its points. K-means locates cluster centres iteratively by minimising the distance between individual cluster points and the cluster centre. K-means requires the specification of cluster centers from the outset. The method begins with a single cluster and selects a variable whose mean is used as a threshold for splitting the data in half. The centroids of these two components are then utilized to initialize k-means in order to optimize the two clusters' membership. Following that, one of the two clusters is chosen for splitting and a variable within it is picked whose mean is utilized as a threshold for splitting the cluster in half. K-means is then used to partition the data into three clusters, each of which is initialized with the centroids of the two split clusters and the remaining cluster's centroid. This procedure is repeated until a predetermined number of clusters has been attained [29], [30].

To compute the k-means clustering for each k. Assume a given a sample dataset $T = \{T_w|w = 1, 2, 3, \ldots, n\}$. Each sample data in T contains f features of continuous data, denoted by $f1, f2, f3\ldots, fn$. The algorithmic approach used in K-Means is as follows: To begin, k initial clustering centres are chosen at random from T, denoted by $Ci(1 < i < k)$. The Euclidean distance betweenCi and the sample data is then calculated and divided by Ciin T, and find the sample data closest to Ci. The sample data is then assigned to the

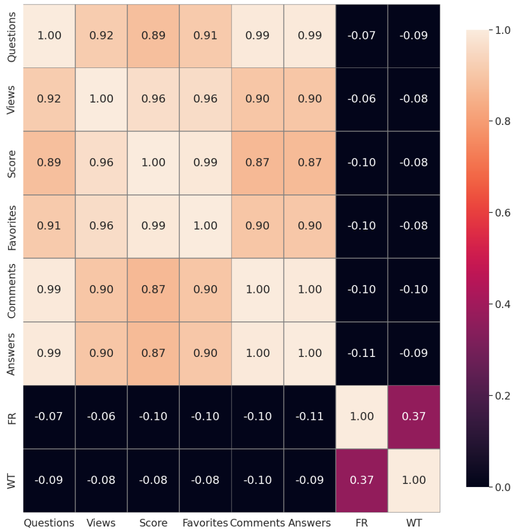


Fig. 3. Covariance Heatmap.

cluster corresponding to Ci, and the average of the sample data in each cluster is recalculated as the new clustering centre. Repeat these steps until the Cluster centre no longer changes or the maximum number of iterations is reached. The Euclidean distance computation formula is as follows:

$$d\left(t, C_i\right) = \sqrt{\sum_{i=1}^{n} \left(t_j - C_{ij}\right)^2} \tag{1}$$

Note: T is the sample data, Ci is the ith cluster center, n is the number of features, tj and Cji, are the jth attribute values of$T$ and Ci respectively. The result of clustering can be judged by the sum of square error of the data set. The formula for calculating the sum of squares of errors is as follows:

$$SSE = \sum_{i=1}^{n} \sum_{n \in Ci} \left|(t, Ci_i)\right|^2 \tag{2}$$

Before applying k-means clustering, the scale and variance of the features, as well as the multi-linearity between the features must be examined as significant correlations between features may lead to erroneous conclusions by overemphasizing one or more underlying components.Pearson correlations between features were calculated, and it was determined that the current data set has a multilinearity effect. A 2D correlation matrix was shown in Fig. 3 to show the relationships between the features.

K-means cluster analysis was carried and the number of clusters was chosen as two. the number of software technologies and development concepts are 353 for the most community supported ones (Cluster1) and 441 of software technologies and development concepts. The distance between the centers of the clusters was determined to FR be 49.45% and 62.79%for of the clusters1 and cluster2, respectively. where the WT the centers are 89.44 minutes and 365.77 minutes for Clusters 1and 2, respectively. After deciding on the number of clusters, some tests must be carried out to check stability, the relative size of the clusters, and external validity.

V. Cluster Evaluation

A good clustering algorithm should achieve high similarity between the data points within the same cluster. To assess cluster quality: The criteria for determining the appropriate number of clusters were as follows: two strategies were employed to assess the quality of the clustering based on criteria for determining the appropriate number of clusters, including Cohesion and Separation metrics, classification and Silhouette index.

A. Cluster Cohesion and Separation

Separation and Cohesion are internal metrics. Cluster Separation quantifies how distinct or well-separated one cluster is from others. Whereas Cluster Cohesion measures the degree to which objects inside a cluster are connected. Separation is calculated by the sum of squares between clusters (BSS). Cohesion is measured by the sum of squares within each cluster (WSS). We can therefore take WSS to be the measure of density and BSS to be the measure of separation. For clustering to be effective, a lower WSS and a larger BSS [31] are required.

B. Silhouette Index

The silhouette index [32] is used to study validity of the separation distance between the generated clusters. It is one of the most well-known techniques for clustering validation [33], [34], [35].It shows the closeness of points in one cluster is to points in nearby clusters and thus provides a visual way to examine factors such as cluster number. The range of this metric is [-1, 1]. Silhouette coefficients of near +1 (as these values are known) suggest that the sample is far distant from the surrounding clusters. A value of 0 denotes that the sample is on or very near the decision boundary between two neighbouring clusters. However, negative values suggest that the samples might be assigned to the incorrect cluster [36], [37]. This is how the silhouette index is computed:

d(i; j) represents the distance between cluster Ci data points and j. We read a(i) as an indication of how well I is allocated to its own cluster (the smaller the value, the better the assignment).

$$a\left(i\right) = \frac{1}{\left|c_i\right| - 1} \sum_{j \in C_i i} d_{i \neq i}\left(i, j\right) \tag{3}$$

Then, we present the mean dissimilarity b(i) of point i to a cluster Ck as the average distance between i and all Ck points (where $Ck \neq Ci$). For each data point, $i \in Ci$.

$$b\left(i\right) = mink \neq i \frac{1}{\left|c_k\right|} \sum_{j \in C} d_k\left(i, j\right) \tag{4}$$

The value of silhouette of one data point i is defined as follows:

$$S\left(i\right)=\frac{b\left(i\right)-a\left(i\right)}{\max\left\{a\left(i\right),b\left(i\right)\right\}},jf\left|c_i\right|>1 \tag{5}$$

Consequently, the s(i) present in the dataset is a measure of the clustering accuracy of the data.

1) Evaluation by Classification : Clustering is performed on unlabeled data to label each cluster. After data has been clustered into groups, a classification technique can be employed. When each cluster's classification model is built separately, there's a good probability of getting better results in terms of accuracy.

Classification based on the clustering process can be indirectly used to evaluate the quality of the clustering process. The evaluation uses the set of Java technologies and development concepts to train model that automates the classification of java technologies and development concepts topics using the supervised machine learning algorithms Random Forest (RF) [38]. RF was chosen since it has been effectively used in many research-related tasks. As a result, the RF approach will be utilised in this study to classify and discriminate between supported and less supported java technologies and development concepts.

VI. Results and Discussion

In this section reports results of the clustering formulation based on test analysis of variance (ANOVA) and the quality of clustering using the statistical error and silhouette score techniques. Additionally, the clustering assessed indirectly by a classification technique, namely, RF.

A. Features to Formulate K-mean Clustering

Fig. 4 shows the result of k-mean clustering. It is important to note that before the application of cluster analysis, the scale and variance of the variables and multilinearity among the variables should be checked. According to the results, the calculated Pearson correlations between the features proved that multilinearity effect exists in the current data set. The correlation matrix is presented in Table III.

The results show that there are features that significantly contribute to the formation of the clusters.The failure rate and Wait time features were found to be the best features that formulate the community support technologies and development concepts clustering. The ANOVA results depicted in Tables X and V and the model Summary diagnostics as shown in Table III and VIII demonstrate the features that significantly contribute to the formation of the clusters. In Table III when clustering model was fed with all features (Views,Score,Favorite,Comment,Answers,FR and WT) there was a high correlation with these features and the clustering model. We further found that Within-group Sum of Squares (WSS) is higher than Between-group Sum of Squares(BSS). As mentioned before that WSS means the sum of distances between the points and the corresponding centroids for each cluster and BSS means the sum of distances between the centroids and the total sample mean multiplied by the number of points within each cluster. After performing the experiments by inserting the features into the model, the best results were obtained only when using the two features FR and WT after achieving a value of WSS lower than BSS as shown in Tables III and VIII. It is also noted that the Table VIII has a value of WSS greater than BSS, and this indicates that the features that achieve the best results for the k-mean clustering formation to determine the levels of community supported technologies and development concepts are FR and WT. Cluster 1 represents the best community supported technologies and development concepts for java developers. Whereas Cluster 2 represents the less community supported technologies and development concepts for java developers. Table IV shows that 44% of the technologies and development concepts in cluster 1 are more supported than the items in cluster 2 that comprises 55.5% of the Java technology and development concepts. The majority of the technologies and development concepts in cluster 2 have less wait time and failure rate. the result help developers gain insights about the community supported technologies and development concepts.

TABLE III. Inputs for Clustering and Diagnostics based on FR and WT

Inputs for Clustering	
Features	Sum of FR
	Sum of WT
Summary Diagnostics	
Number of Clusters:	2
Number of Points	794
Between-group Sum of Squares	10.039
Within-group Sum of Squares	9.3994
Total Sum of Squares	19.438

TABLE IV. THE Average Value within each Cluster base on FR and WT

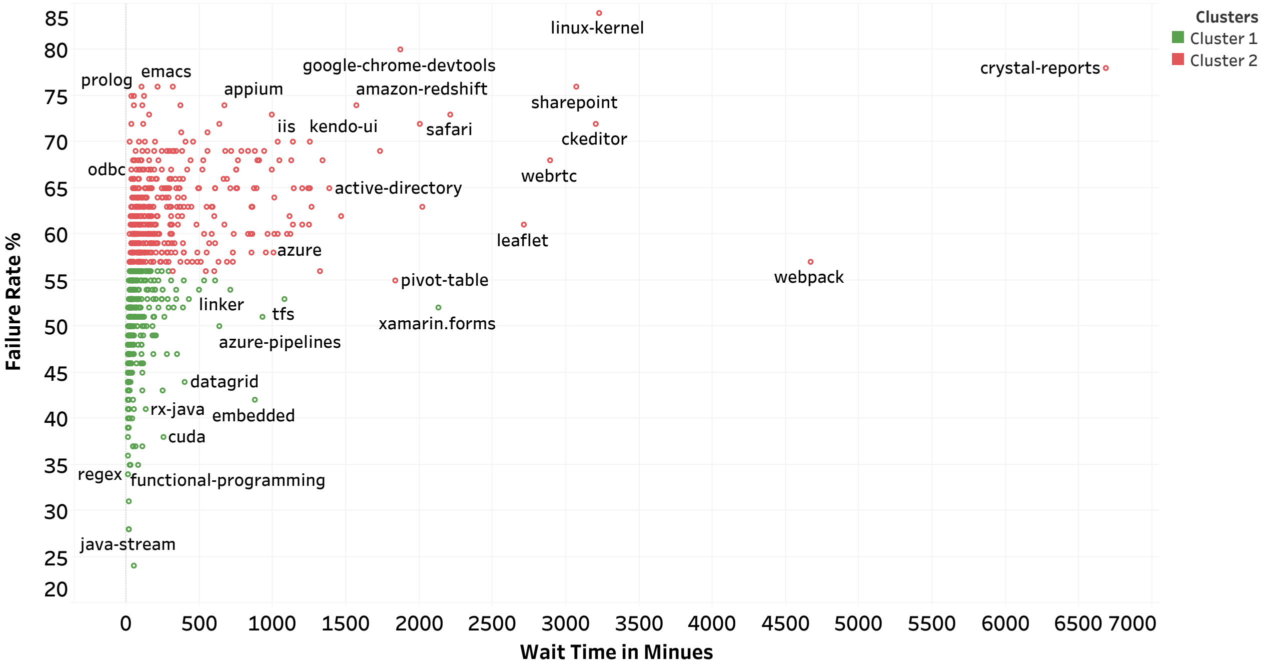
Centers	Cluster1	Cluster 2
Number of Items	353	441
Sum of FR	49.45	62.798
Sum of WT	89.446	365.77

B. Assets the Quality of K-means Clustering

Extra evaluation technique was used to examine the cohesiveness of the quality matrices that is computed using the findings of the average global silhouette. Fig. 5 illustrates a silhouette curve for estimating the ideal number of clusters, gauging each cluster's quality cohesiveness. Fig. 5 reveals that the average silhouette score is (97%). This is a reasonable value because the clustering is predicated on the silhouette index, which is already high. The result shows that the two group clustering is an optimal number, thus the resulted clustering can be used as a new feature as an input of the classification models.

TABLE V. ANOVA Analysis of K-means Cluster Analysis based on FR and WT

			Model	Error
Variable	F-statistic	p-value	Sum of Squares	Sum of Squares
FR	498.6	0.0	9.703	15.41
WT	66.05	1.67E-15	0.3357	4.026



z

Fig. 4. Sum of WT vs. Sum of FR. Color Shows Details about Clusters. The Marks are Labeled by Software Technologies and Development Concepts of Java Ecosystem.

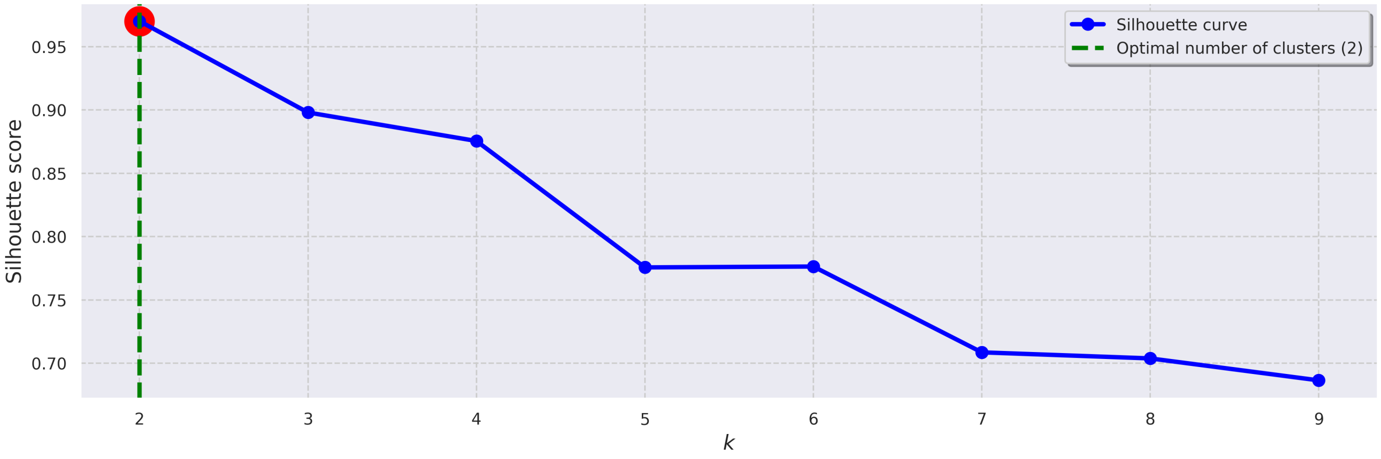


Fig. 5. Silhouette Curve for Predict Optimal Number of Clusters

C. Experimental Setup Using the RF classification

The RF model was run based on the most commonly-used method to explore hyper-parameter configuration space called Grid search (GS) [39] to tune the model to fit the best result that is provided by the respective implementation of python. GS operates by calculating the Cartesian product of a finite set of user-specified values.The code developed by [40] was used, which implements hyper-parameter optimization for machine learning algorithms, the values for the parameters of the RF ML algorithm considered in this paper are summarized in Table VI.

TABLE VI. RF CONFIGURATION HYPER-PARAMETER SPACE.

Hyper-parameter	Type	Search Space	best values
n_estimators	Discrete	[10,100]	20
max_depth	Discrete	[5,50]	15
min_samples_leaf	Discrete	[1,11]	1
criterion	Categorical	gini, entropy	gini

This study will employ a different metric of evaluation to assess this RF algorithm. These metrics are calculated based on four primary areas. In a supervised classification issue, a true output and a predicted or model-generated output exist. Therefore, each data point's result will be categorised as one of the following:

- True Positive (TP): both the label and the prediction are positive.
- True Negative (TN): both the label and the prediction are negative.
- False Positive (FP): describes a situation in which the label is negative but the prediction is positive.
- False negative (FN): although the label is positive, the prediction is negative.

These four categories are the foundation of the majority of classification evaluation metrics. Performance parameters were used to evaluate the model: Accuracy, precision, recall and F-measure.

- Accuracy: It represents the proportion of correctly classified supported technologies and development concepts. It is technically defined as:

$$Accuracy = \frac{TP + TN}{(TP + FN) + (FN + TN)} \tag{6}$$

- Precision: It is the proportion of correctly identified supported technologies and development concepts relative to the total number of supported technologies and development concepts in a X class. The range of values is from 0 (poor precision) to 1 (high precision). The weighted average precision is determined as the mean of Precision of the true class and false class in relation to the number of tags predicted for each class. It is described as:

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

- Recall: It is the proportion of successfully classified tags relative to the number of observed true instances. The values vary from 0 (poor recall) and 1 (high recall). The weighted average recall is derived as the mean of recall of the true class and recall of the false class, weighted by the number of tags tagged with each class.

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

- F1-Measure: It denotes a performance indicator that considers both the precision and recall of the classification obtained. The formula is as follows:

$$F - Measure = \frac{2 * recall * precision}{recall + \ precision} \tag{9}$$

- Area under ROC-Curve (AUC): It's a measure of the classifier's predictive strength, essentially telling us how well the model can distinguish between classes. AUC of 1 shows the best performance, while 0.5 indicates that the performance is comparable to that of a random classifier.

The dataset was divided into a training set and a testing set. In the training phase, cross validation is used 80% of the time, and 20% of the time in the testing phase. Cross-validation of 20% is used to test the model. RF algorithm was found to be achieving an average accuracy score of 99.49%. Table VII summarizes the results of the RF classifier based on these results, the configurations are shown in Table VI

TABLE VII. DETAILED ACCURACY BY CLASS

	Precision	Recall	F-Measure	ROC Area	Class
	0,994	0,994	0,994	0,999	Cluster 1
	0,995	0,995	0,995	0,999	Cluster 2
Weighted Avg.	0,995	0,995	0,995	0,999	

VII. CONCLUSION

In this paper, we introduced an approach based on K-mean clustering techniques to identify the level of community supported software technologies and development concepts leveraging software developer's discussions on SO. The approach is based on tags explicitly assigned to questions. we identified how community supported and development concepts of java technologies based on java developers questions on SO. First, we created data set and used it to automate the clustering of software technologies and development concepts. In the

TABLE VIII. INPUTS FOR CLUSTERING AND DIAGNOSTICS BASED ON ALL FEATURES.

Inputs for Clustering	
Features	Sum of FR
	Sum of WT
	Sum of Answers
	Sum of Comments
	Sum of Views
	Sum of Favorites
	Sum of Score
Summary Diagnostics	
Number of Clusters:	2
Number of Points	794
Between-group Sum of Squares	10.188
Within-group Sum of Squares	21.245
Total Sum of Squares	31.433

TABLE IX. THE AVERAGE VALUE WITHIN EACH CLUSTER BASED ON ALL FEATURES

Centers	**Cluster1**	**Cluster 2**
Number of Items	355	439
Sum of FR	49.535	62.79
Sum of WT	88.6	367.71
Sum of Answers	5892.7	1693.9
Sum of Comments	10657	3275.3
Sum of Views	7.06E+06	2.55E+06
Sum of Favorites	1398.2	444.91
Sum of Score	5132.5	1588.6

TABLE X. ANOVA ANALYSIS OF K-MEANS CLUSTER ANALYSIS BASED ON ALL FEATURES

			Model	**Error**
Features	F-statistic	p-value	Sum of Squares	Sum of Squares
Sum of FR	492.3	0.0	9.58	15.41
Sum of WT	67.47	8.88E-16	0.3429	4.026
Sum of Answers	21.21	4.80E-06	0.04448	1.661
Sum of Comments	20.09	8.48E-06	0.0437	1.723
Sum of Score	17.47	3.24E-05	0.06513	2.952
Sum of Favorites	17.27	3.60E-05	0.0549	2.518
Sum of Views	14.44	1.56E-04	0.05726	3.141

first approach, we identified which feature can formulate clustering the community supported and development concepts. We implemented correlation analysis, ANOAVA and diagnosis the K-mean model to get the best features to formulate the levels of groups of community supported and development concepts. we found that features that formulate the two clusters to determine the community supported software technologies and development concepts levels are failure rate, that is the percentage of its questions that do not have an accepted answer, and its median wait time, that is the median time to get accepted answers;are the best features. We found that the majority of Java technologies and development concepts are labeled with cluster 1 most community supported technologies and development concepts and cluster 2 less community supported technologies and development concepts.The approach was evaluated in two steps. The quality of clustering shows that the best value is 97%, the higher the silhouette index value, the more effective the construction of clusters. To assess the approach, the identified technologies and development concept groups were added as new features to the dataset and then RF was applied. The evaluation with the java data set showed that the approach outperforms the RF with an average precision and recall of 0.995 and 0.995, respectively.

VIII. Future Work

In the future, we are planning to apply and compare the classification of results based on different types of clustering algorithms to choose the right supported technologies. Also building detailed user interface development to maximize the benefits of a decision support system (DSS) in the software development sector.

References

[1] C. Rosen and E. Shihab, "What are mobile developers asking about? a large scale study using stack overflow," *ESEM*, vol. 21, no. 3, pp. 1192–1223, 2016.

[2] M. Linares-Vásquez, B. Dit, and D. Poshyvanyk, "An exploratory analysis of mobile development issues using stack overflow," in *Proceedings of the 10th Working Conference on Mining Software Repositories*. IEEE Press, 2013, pp. 93–96.

[3] Z. e. a. Mehrab, "Mining developer questions about major web frameworks." in *WEBIST*, 2018, pp. 191–198.

[4] F. Fischer, K. Böttinger, H. Xiao, C. Stransky, Y. Acar, M. Backes, and S. Fahl, "Stack overflow considered harmful? the impact of copy&paste on android application security," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 121–136.

[5] C. Chen, Z. Xing, and L. Han, "Techland: Assisting technology landscape inquiries with insights from stack overflow," in *2016 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2016, pp. 356–366.

[6] Y. Wu, S. Wang, C.-P. Bezemer, and K. Inoue, "How do developers utilize source code from stack overflow?" *Empirical Software Engineering*, vol. 24, no. 2, pp. 637–673, 2019.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[8] L. Beard and N. Aghassibake, "Tableau (version 2020.3)," *Journal of the Medical Library Association: JMLA*, vol. 109, no. 1, p. 159, 2021.

[9] S. M. Nasehi, J. Sillito, F. Maurer, and C. Burns, "What makes a good code example?: A study of programming q&a in stackoverflow," in *2012 28th IEEE International Conference on Software Maintenance (ICSM)*. IEEE, 2012, pp. 25–34.

[10] A. Barua, S. W. Thomas, and A. E. Hassan, "What are developers talking about? an analysis of topics and trends in stack overflow," *Empirical Software Engineering*, vol. 19, no. 3, pp. 619–654, 2014.

[11] C. Treude, O. Barzilay, and M.-A. Storey, "How do programmers ask and answer questions on the web?: Nier track," in *Software Engineering (ICSE), 2011 33rd International Conference on*. IEEE, 2011, pp. 804–807.

[12] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*. Elsevier, 2011.

[13] R. Kiani, S. Mahdavi, and A. Keshavarzi, "Analysis and prediction of crimes by clustering and classification," *International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 8, pp. 11–17, 2015.

[14] T. Sajana, C. S. Rani, and K. Narayana, "A survey on clustering techniques for big data mining," *Indian journal of Science and Technology*, vol. 9, no. 3, pp. 1–12, 2016.

[15] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[16] S. Naeem and A. Wumaier, "Study and implementing k-mean clustering algorithm on english text and techniques to find the optimal value of k," *Int. J. Comput. Appl*, vol. 182, no. 31, pp. 7–14, 2018.

[17] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping multidimensional data*. Springer, 2006, pp. 25–71.

[18] S. Joshi and B. Nigam, "Categorizing the document using multi class classification in data mining," in *2011 International Conference on Computational Intelligence and Communication Networks*. IEEE, 2011, pp. 251–255.

[19] G. Kesavaraj and S. Sukumaran, "A study on classification techniques in data mining," in *2013 fourth international conference on computing, communications and networking technologies (ICCCNT)*. IEEE, 2013, pp. 1–7.

[20] A. Ahmad, C. Feng, S. Ge, and A. Yousif, "A survey on mining stack overflow: question and answering (q&a) community," *Data Technologies and Applications*, 2018.

[21] K. Bajaj, K. Pattabiraman, and A. Mesbah, "Mining questions asked by web developers," in *MSR 2014*, 2014, pp. 112–121.

[22] V. et al., "What do client developers concern when using web apis? an empirical study on developer forums and stack overflow," in *ICWS '16*. IEEE, 2016, pp. 131–138.

[23] F. Almansoury, S. Kpodjedo, and G. E. Boussaidi, "Investigating web3d topics on stackoverflow: a preliminary study of webgl and three. js," in *The 25th International Conference on 3D Web Technology*, 2020, pp. 1–2.

[24] X.-L. Yang, D. Lo, X. Xia, Z.-Y. Wan, and J.-L. Sun, “What security questions do developers ask? a large-scale study of stack overflow posts,” *JCST*, vol. 31, no. 5, pp. 910–924, 2016.

[25] N. Meng, S. Nagy, D. Yao, W. Zhuang, and G. A. Argoty, “Secure coding practices in java: Challenges and vulnerabilities,” in *ICSE‘18*, 2018, pp. 372–383.

[26] K. A. Nazeer and M. Sebastian, “Improving the accuracy and efficiency of the k-means clustering algorithm,” in *Proceedings of the world congress on engineering*, vol. 1. Citeseer, 2009, pp. 1–3.

[27] A. Bansal, M. Sharma, and S. Goel, “Improved k-mean clustering algorithm for prediction analysis using classification technique in data mining,” *International Journal of Computer Applications*, vol. 157, no. 6, pp. 0975–8887, 2017.

[28] B. Aubaidan, M. Mohd, and M. Albared, “Comparative study of k-means and k-means++ clustering algorithms on crime domain,” 2014.

[29] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, “An efficient k-means clustering algorithm: Analysis and implementation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[30] M. E. Celebi, H. A. Kingravi, and P. A. Vela, “A comparative study of efficient initialization methods for the k-means clustering algorithm,” *Expert systems with applications*, vol. 40, no. 1, pp. 200–210, 2013.

[31] K. M. Lee, K. M. Lee, and C. H. Lee, “Statistical cluster validity indexes to consider cohesion and separation,” in *2012 international conference on fuzzy theory and its applications (ifuzzy2012)*. IEEE, 2012, pp. 228–232.

[32] P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[33] S. Chaimontree, K. Atkinson, and F. Coenen, “Best clustering configuration metrics: towards multiagent based clustering,” in *International Conference on Advanced Data Mining and Applications*. Springer, 2010, pp. 48–59.

[34] S. A. Burney and H. Tariq, “K-means cluster analysis for image segmentation,” *International Journal of Computer Applications*, vol. 96, no. 4, 2014.

[35] A. R. Mamat, F. S. Mohamed, M. A. Mohamed, N. M. Rawi, and M. I. Awang, “Silhouette index for determining optimal k-means clustering on images in different color models,” *Int. J. Eng. Technol*, vol. 7, no. 2, pp. 105–109, 2018.

[36] A. Starczewski and A. Krzyżak, “Performance evaluation of the silhouette index,” in *International conference on artificial intelligence and soft computing*. Springer, 2015, pp. 49–58.

[37] X. Wang and Y. Xu, “An improved index for clustering validation based on silhouette index and calinski-harabasz index,” in *IOP Conference Series: Materials Science and Engineering*, vol. 569, no. 5. IOP Publishing, 2019, p. 052024.

[38] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[39] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, “Algorithms for hyper-parameter optimization,” *Advances in neural information processing systems*, vol. 24, 2011.

[40] L. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: Theory and practice,” *Neurocomputing*, vol. 415, pp. 295–316, 2020.