International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

A Comparative Machine Learning Approach for Usability Evaluation of Localized Software in Afghanistan

Abdul Jalil Dilawar[1*], Irfanullah Sabawoon[2], Usman Zaki[3]
[1]Dean and lecturer of Computer Science Faculty, Tabesh University, Jalalabad Afghanistan
[2]Head of Artificial Intelligence Department, Tabesh University, Jalalabad Afghanistan
[3]Lecturer at Noor Ali Learning Center, Jalalabad Afghanistan

ABSTRACT: Usability is one of the most important software quality attributes in determining user satisfaction, productivity and system acceptance. Afghanistan is a multilingual country, and usability problems with localized software applications may occur due to linguistic diversity, cultural differences, and scarce resources for evaluating the usability of software applications. These traditional usability assessment techniques, such as heuristic evaluation and laboratory testing, are difficult to implement in a capacity-constrained environment, as they require significant time, expertise and funds. The study suggests an automatic usability evaluation framework by machine learning techniques for localized software systems in Afghanistan. Data were gathered from ISO 9241-11 usability framework-based structured questionnaires from users of localized software applications. After collecting the responses, they were subjected to a pre-processing procedure involving cleaning, encoding and normalizing. Three supervised machine learning algorithms Decision Tree (DT), K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) were trained and tested with 80:20 ratio of training and test data. Model performance was evaluated using the accuracy, precision, recall, F1-score and confusion matrix analysis. Experimental results indicate that the highest accuracy of 91.4% is obtained by DT followed by SVM with 90.0% accuracy and KNN with 88.7%. The results show that the Decision Tree is a good model for balancing the prediction accuracy and interpretation of local software environments. The proposed framework has been designed to be both scalable and cost-effective, and can be used to complement existing usability evaluation methods, and can also be used to aid in software quality improvement efforts in multilingual and developing country settings.

KEYWORDS: decision tree, human–computer interaction, localized software, k-nearest neighbors, machine learning, support vector machine, usability evaluation, Afghanistan

INTRODUCTION

Software systems are essential in today's society and are widely used in education, health care and medicine, banking, business management, communication, and government services. As software applications become more and more important, software quality attributes have gained significance, particularly usability that directly affects the user satisfaction, productivity and acceptance of the system [1, 2]. A software application can have a lot of capabilities, but if the user has problems as they interact with the system, they may affect the application's adoption and success. The ISO 9241-11 standard defines usability as the ability of specified users to perform specified tasks with effectiveness, efficiency and satisfaction in a specified context of use [1]. Effectiveness is the degree of correctness and completeness of the actions performed by users to complete the task, efficiency is the resources used to complete the task and satisfaction is the comfort and positive attitude of users when using the software system. These dimensions together give a holistic perspective on the user's assessment of software quality. In the case of localized software systems, usability is even more crucial. The term software localization is for the process of making software products suitable for a target user group in terms of language, culture and region. Localization goes beyond translation, involving the modification of interface components, phrases, date formats, navigation, symbols and cultural references to ensure that they are compatible with local user expectations [20]. Good localization makes technology more accessible, better user experience and more legible to different groups of people. The localization of software takes on great importance in Afghanistan, where multilingualism prevails and both the local languages, Dari and Pashto are used. All kinds of organizations, such as educational institutes, government bodies, and public service providers are turning to localized software systems for better communication and service delivery. Although these efforts have been made there are still usability problems with many of the localized applications such as translation inconsistencies, culturally inappropriate interface designs, navigating the system, and limited user centered evaluation practices. These problems have a tendency to decrease software effectiveness and negatively influence user satisfaction. A number of classic usability evaluation methods have been used to evaluate software quality including heuristic evaluation, cognitive walkthrough, usability testing, interviews and questionnaire assessment [2]. While these techniques are useful, they often involve considerable time, money and specialized personnel. In addition, manual evaluation methods can be subjective and can be costly and complicated when dealing with large numbers of users or multiple software applications. The restrictions are especially noticeable in resource-poor settings where there might not be a lot of usability experts or testing facilities. The last few years have given rise to new development tools in the field of Artificial Intelligence (AI) and Machine Learning (ML), which offer the possibility of automating software quality evaluation and usability assessment. Machine learning methods can detect patterns in vast amounts of data and create predictive models that can assist in making decisions [3]–[8]. Machine learning can be used to automate usability assessment through analysing user feedback, questionnaire answers, behavioral data, and software usage patterns, whilst simultaneously minimizing human effort and enhancing the uniformity of usability evaluation. Decision Tree (DT), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) are among the supervised machine learning algorithms that have shown to be efficient in classification and prediction problems [3]–[13]. The Decision Trees offer understandable and interpretable decision rules, KNN classifies an observation based on similarity measures between the different observations, and SVM builds optimal decision boundaries that maximize the classification accuracy. These properties render the three algorithms as good candidates for usability prediction and software quality assessment. Machine learning methods for usability assessment, sentiment analysis, user experience evaluation, and software quality prediction have proven their effectiveness in previous studies [15]–[19]. Yet, very little research has been conducted on developing countries and on software environments in general. Little attention has been paid to local software systems used in multilingual and resource-limited environments like Afghanistan. In addition, there is a lack of studies that perform comprehensive comparisons of various machine learning algorithms for usability prediction in a localized software application environment.

To address these limitations, this study proposes a comparative machine learning framework for usability evaluation of localized software systems in Afghanistan. The framework utilizes questionnaire data collected from users of localized software applications and applies Decision Tree, K-Nearest Neighbors, and Support Vector Machine algorithms to predict usability outcomes. The comparative analysis aims to identify the most suitable classifier for automated usability evaluation and contribute to the growing integration of machine learning techniques in usability engineering.

Problem Statement

There are usability related issues which occur regularly in localized software in Afghanistan and have an impact on user satisfaction, usability efficiency, and software adoption. Conventional methods for usability evaluation are often time consuming, costly and high in the number of experts needed and are hard to apply in many settings. Furthermore, standards and automated usability evaluation frameworks are not uniform, which makes it difficult for software developers to easily find usability issues. There are limited studies that have explored the application of machine learning techniques in software quality prediction and usability assessment in Afghan localized software.While machine learning techniques have demonstrated some effectiveness in the context of software quality prediction and usability assessment, there is a lack of studies that investigate their use in the context of Afghan localized software. Moreover, there are limited comparative studies of usability prediction algorithms (DT, KNN, SVM) in multilingual and developing countries. Thus, an objective, scalable and data-driven framework that can automatically assess software usability and determine the best machine learning classifier to accomplish this is needed.

Significance of the Study

This research is relevant for Human–Computer Interaction, Software Engineering and Machine Learning, showing the applicability of predictive models for usability evaluation. The proposed framework is an objective and scalable solution to the traditional usability assessment methods. The results can guide software developers in improving software quality, educational institutions in addressing user needs and preferences, government bodies and policymakers in creating more inclusive policies and regulations. In addition, this study offers empirical evidence of the comparative performance of DT, KNN and SVM algorithms in localized software environments

Organization of the Paper

The remainder of this paper is organized as follows. Chapter 2 presents the literature review and identifies existing research gaps. Chapter 3 describes the research methodology, including data collection procedures, preprocessing techniques, machine learning models, and evaluation metrics. Chapter 4 presents the experimental results and comparative performance analysis of the selected classifiers. Finally, Chapter 5 concludes the study, summarizes the key findings, and provides recommendations for future research.

LITERATURE REVIEW

Usability evaluation is an important part of the software quality assessment process since it ensures the software systems are usable to the users so they can perform the desired tasks. With software applications growing in a number of fields, organizations are looking for ways to assess usability efficiently and enhance the user experience. Traditional usability evaluation approaches have been extensively used, but there have been recent advances in Artificial Intelligence (AI) and Machine Learning (ML) that have helped develop automated methods to enhance the efficiency and scalability of usability assessment. This chapter summarizes literature on the existing researches related to the usability evaluation, application of machine learning techniques in usability prediction, Decision Tree (DT), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) algorithm, and the gap this research aims to fill.

Usability Evaluation in Software Systems

Usability is acknowledged as one of the most significant quality attributes of software systems that affect user satisfaction, productivity and software acceptance. In line with ISO 9241-11, usability is the degree to which specified users can use a product to achieve specified goals effectively, efficiently and in a satisfying way in a given context of use [1]. Most current usability evaluation frameworks are based on these three dimensions. Traditional approaches to usability evaluation are heuristic evaluation, cognitive walkthroughs, usability testing, interviews, questionnaires and expert review [2]. This is a way of identifying interface problems, navigation issues and user experience issues. But many classical approaches are time consuming, costly and involve highly trained personnel. Besides, manual evaluation process can be subjective and might not be feasible for software assessment on a large scale. Conventional usability assessment has many problems that lead researchers to study automatic evaluation and data-driven evaluation. As more and more user-generated data, questionnaire responses, and software usage logs are becoming available, there are opportunities to use machine learning techniques for usability prediction and software quality assessment.

Machine Learning in Usability Evaluation

Machine Learning is a subcategory of Artificial Intelligence which allows a computer system to learn the patterns in data and predict without being explicitly programmed [3]–[8]. Machine learning methods have proven to be effective in various applications such as classification, prediction, recommendation and sentiment analysis, as well as software quality evaluation. Machine learning algorithms have been used in usability engineering to detect usability patterns and to forecast software quality based on user feedback, survey responses, behavioral metrics, and interactions. These can help decrease reliance on manual review and offer scalable solutions that can efficiently manage larger datasets. One of the first usability assessment methods using machine learning was proposed by Oztekin et al. [2] and they proved that usability issues could be identified correctly by predictive models in web based information systems. Their study identified the possible use of automated evaluation methods in addition to the conventional usability assessment methods. Recently, Liu [15] performed a systematic review of Artificial Intelligence applications in usability evaluation and found that the application of machine learning methods has made a significant contribution to efficiency in the assessment process and to its predictive accuracy. The review highlighted the need for automated usability evaluation as an aiding tool for evidence-based decision making and decreased effort for software quality assessment. The widespread use of machine learning in Human Computer Interaction (HCI) research indicates that it can solve the usability problems confronted in contemporary software systems.

Decision Tree for Usability Prediction

One popular supervised machine learning classification and prediction algorithm is Decision Tree (DT). The algorithm builds a hierarchical tree structure with the decision criteria at the internal nodes and the possible outcomes at the branches and the classification result at the leaf nodes [9]. The biggest benefits of Decision Tree models are their interpretability. Decision Trees are a type of machine learning algorithms that are considered as "black box" models, unlike most of the other machine learning models, which can be understood by the researchers and the software developers easily with the explicit decision rules. This is an important characteristic from the usability evaluation point of view, as developers may be interested in the explanations of factors affecting the usability results. Rahman and Ahmed [16] have used Decision Tree classification in predicting software usability and achieved positive results. They showed that Decision Trees were able to identify relationships between usability attributes and software quality outcomes. Likewise, Basri et al. [17] concluded that Decision Tree models can be used to classify users' feedback sentiment related to usability and maintain transparency and interpretability of the model. Decision Trees can also take a mixture of numerical and categorical variables without needing to do too much in terms of preprocessing. Their properties make them ideal for usability data sets commonly obtained through questionnaires, which are likely to use Likert scale measurements. As a result, Decision Tree is still one of the most potential algorithms to use for usability prediction tasks.

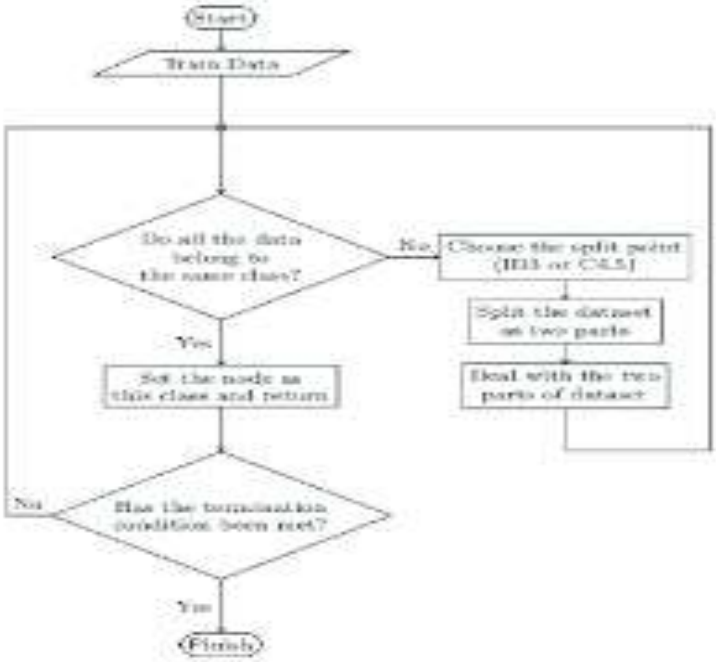

Figure 1. Decision tree flowchart

K-Nearest Neighbors for Usability Prediction

The KNN (K-Nearest Neighbors) algorithm is a non-parametric supervised learning method, which classifies an unknown object from new datasets based on the similarity between an observation in the new dataset with one of the observations in the training set [10]. The algorithm sets the class label based on which class most of the neighboring points in the feature space are. KNN is successfully used in a variety of fields such as healthcare, recommendation systems, educational data mining, and prediction of user behaviour. In usability evaluation, KNN provides similarity based method and in that if two users have similar perceptions and experiences, they are likely to be in the same usability category. KNN is a simple classifier, widely used as a benchmark in machine learning research due to its simplicity and effectiveness. KNN is one of the non-model based approach that does not need much training and is able to accommodate to different types of classification problems. The performance is however highly sensitive to the choice of K parameter and the distance measure. In addition, when KNN is used on large data sets, distance calculations may be time consuming for the number of classification tasks. KNN still shows good performance in software quality prediction and usability assessment studies, despite these drawbacks. It has the advantage of pattern classification by similarity and it could be presented as an alternative approach to tree-based and hyperplane-based pattern classification.

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

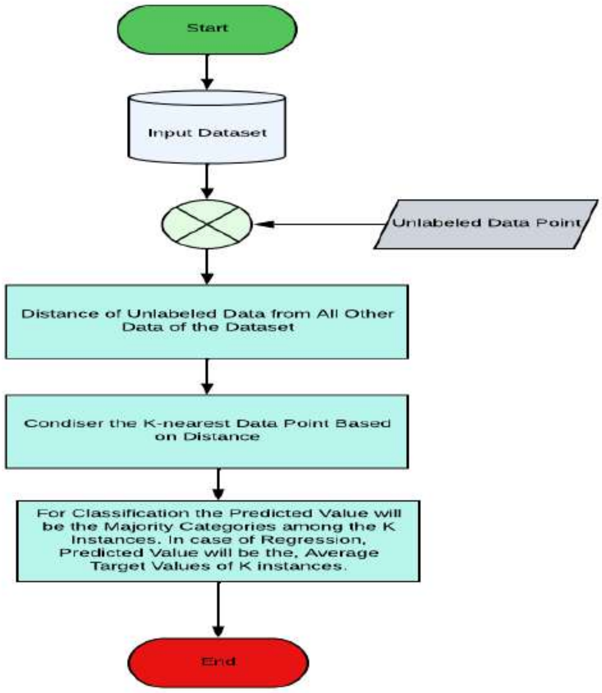

Figure 2. KNN flowchart

Support Vector Machine for Usability Prediction

Supervised machine learning algorithm, called Support Vector Machine (SVM), is used for classification problems and has proven to be a powerful algorithm [11] [12]. The algorithm is able to build an optimal hyperplane that maximizes the distance between the classes in the feature space. SVM can deal with linear and non-linear classification problems efficiently by utilizing kernel functions. Many studies have shown that SVM has good predictive ability for both software quality assessment and software usability prediction. Shah et al. [18] showed that SVM could be used to get the reliable classification accuracy in automated usability assessment systems. Likewise, Asghar et al. [19] suggested an intelligent usability prediction framework based on SVM and achieved a classification accuracy of over 90%. The successful SVM is due to its powerful generalization ability and the immunity to overfitting. The algorithm works well especially in high dimensional data sets where many usability properties contribute to the classification results. While SVM is often highly accurate, the decision-making process is less interpretable than those of Decision Trees and therefore may not be suitable for use in certain usability evaluation scenarios. However, SVM is still among the most powerful machine learning algorithms for classification problems, and is an important benchmark in comparative usability prediction studies.

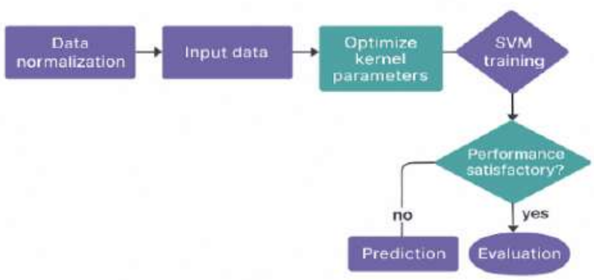

Figure 3. Support Vector Machine flowchart

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

Comparative Studies of DT, KNN, and SVM

Comparative evaluation of machine learning algorithms is crucial as classifier performance can be influenced by the nature of the dataset and the application. In order to determine the most appropriate model for usability prediction and software quality assessment, several researchers have made comparisons of multiple classification techniques. Basri et al. [17] evaluated several machine learning algorithms in classifying sentiment from usability data and found that the effectiveness of the classifiers to classify the sentiment of a usability depends on the data characteristics and evaluation purpose. Their results showed that SVM performance in terms of predictive accuracy is high and DT is better in terms of interpretability. Likewise, Shah et al. [18] compared several supervised learning algorithms for automated usability prediction and found variations in the performance of classifiers. They highlighted the need to perform comparative analysis prior to choosing a machine learning model for actual usability evaluation tasks. In previous comparisons, it has been argued that Decision Trees are transparent and easy to interpret, KNN is a good choice of method in similarity based classification problems and SVM is a good choice of method in complex data sets. Few studies have investigated their performance in localized software and multilingual user settings, however.

Research Gap

Despite the evidence from previous studies that shows how effective machine learning methods could be used in usability evaluation, there is still an evident gap in literature in the following areas. Firstly, most of the studies focus on the evaluation of general software programs and not localized software application systems tailored for multicultural societies. Secondly, the usability prediction of software in developing countries has not been extensively studied because it is greatly impacted by the cultural, linguistic, and other limitations faced by such nations. Thirdly, while many researchers have evaluated their usability predictions based on the use of certain machine learning methods, only few have performed the comparison of several classifiers in their experiments. Consequently, it makes it difficult for developers of software products to choose which classifier would be more applicable for certain conditions. Lastly, the studies on usability evaluation of localized software systems have not been done for Afghanistan despite its increasing need for localized software applications in education and public sectors. In order to fill this gap in literature, the current study will use the comparative analysis approach by evaluating localized software application systems using Decision Tree, K-Nearest Neighbors, and Support Vector Machine classifiers in Afghanistan.

METHODOLOGY

The study follows a quantitative and experimental research design. A questionnaire-based dataset was collected from users of localized software systems operating in Afghanistan. The collected data were transformed into machine-readable format and used to train supervised machine learning models.

The research process consisted of the following stages:

1. Questionnaire Design
2. Data Collection
3. Data Preprocessing
4. Feature Selection and Encoding
5. Model Development
6. Performance Evaluation
7. Comparative Analysis

The overall framework enables automated usability prediction by analyzing user responses related to software effectiveness, efficiency, and satisfaction.

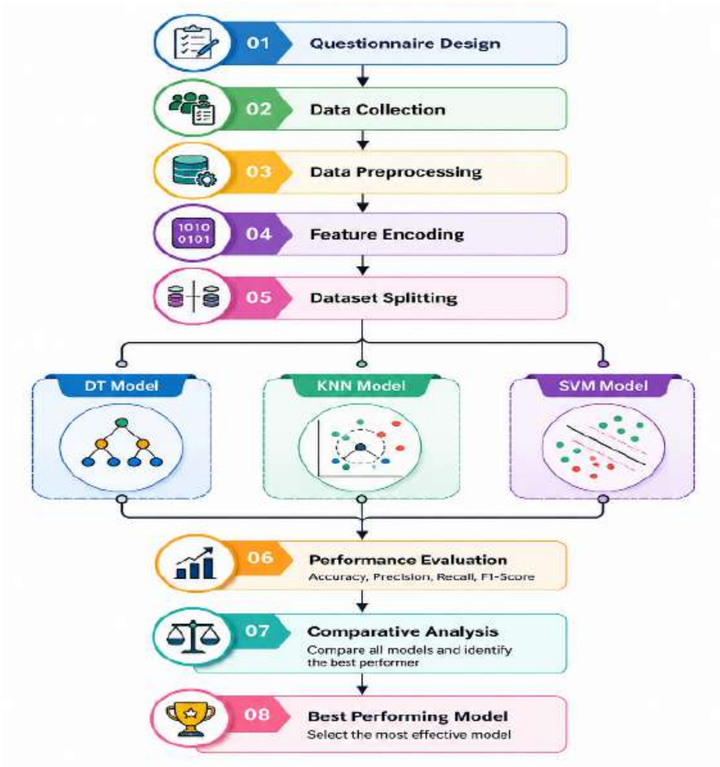

Figure 4. Proposed Research Framework

The sample group was users who frequently used the localized software applications in Afghanistan. The sample group involved university students, professors, IT specialists, administration officials, and software specialists. In total, 300 valid responses were collected through a structured questionnaire. Responses that were incomplete and inconsistent were eliminated during pre-processing.

Table 1. Sample Distribution

Respondent Category	Description
Students	Users of educational software
Faculty Members	Academic software users
IT Officers	Technical professionals
Administrative Staff	Organizational software users
Software Practitioners	Software developers and testers

Questionnaire Development

Data were collected through a structured questionnaire developed according to the ISO 9241-11 usability framework [1]. The questionnaire was designed to measure three major usability dimensions:

- Effectiveness
- Efficiency
- User Satisfaction

The survey consisted of ten usability-related questions and one overall usability classification question. Responses were recorded using a five-point Likert scale:

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

Table 2. Responses values

Response	Value
Strongly Disagree	1
Disagree	2
Neutral	3
Agree	4
Strongly Agree	5

The questionnaire evaluated several usability attributes including:

- Interface clarity
- Ease of learning
- Navigation efficiency
- Error recovery
- Task completion effectiveness
- User confidence
- Language appropriateness
- Cultural suitability
- System responsiveness
- Overall satisfaction

These attributes served as independent variables, while the overall usability assessment served as the target variable for classification.

Data Preprocessing

In order to enhance the quality of the dataset and prepare it for machine learning operations, data preprocessing was conducted.

Data Cleaning

Collected data was checked for any missing value, duplicate entries, or incomplete entries. All invalid entries were then deleted.

Data Encoding

As algorithms require numeric input values, the collected data was encoded using the Likert scale into numeric data.
Strongly Disagree = 1
Disagree = 2
Neutral = 3
Agree = 4
Strongly Agree = 5
This transformation converted all questionnaire responses into machine-readable format.

Data Normalization

Feature normalization was done to make sure all the usability parameters contribute equally when learning the model. This is because normalization helps lessen the effect of features having larger numeric values. Normalization also improves the performance of distance-based classifiers like KNN.

RESULT AND DISCUSSION

Responses were obtained from 300 participants in Afghanistan based on a survey of 10 questions together with the usability assessment score. Interestingly enough, nobody left unanswered any questions, which is rather rare for user surveys. From the correlation heatmap, one may observe that all 10 questions show some kind of positive relationship to usability overall, though only in a weak manner - the weakest is Q3 with the correlation coefficient equal to 0.16 and the strongest being Q1 with the coefficient of 0.29. So there is no standout question, however, almost all questions have their own contribution to making. Question Q1 shows

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

its strength, but there still is no dramatic gap between it and others. Another interesting point is that questions do not correlate very well with each other. Therefore, assessing usability should be more complicated and include several questions.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300 entries, 0 to 299
Data columns (total 11 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Q1         300 non-null    int32
 1   Q2         300 non-null    int32
 2   Q3         300 non-null    int32
 3   Q4         300 non-null    int32
 4   Q5         300 non-null    int32
 5   Q6         300 non-null    int32
 6   Q7         300 non-null    int32
 7   Q8         300 non-null    int32
 8   Q9         300 non-null    int32
 9   Q10        300 non-null    int32
 10  Usability  300 non-null    int32
dtypes: int32(11)
memory usage: 13.0 KB
```

Figure 5. Dataset information

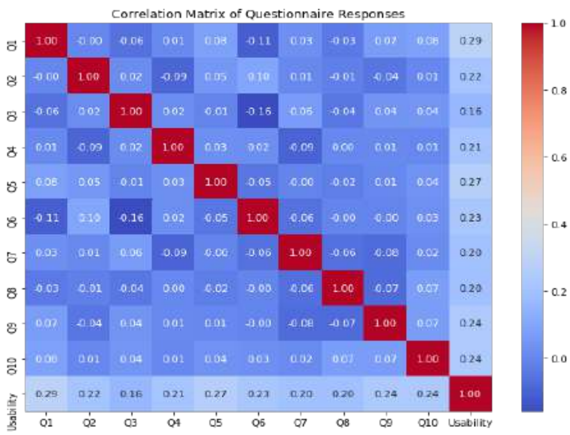

Figure 6. Correlation Matrix of Questionnaire

Confusion matrix in figure 7 indicates that of 300 individuals surveyed in Afghanistan that used the localized software, the classifier perfectly identified 34 individuals who considered the software usable and 17 others that considered it not usable. Nonetheless, the classifier made mistakes by classifying 4 individuals that considered the software usable as "Not Usable" and 5 others that had problems with using the software as "Usable". This makes it possible to easily identify the accuracy of the classifier as well as its faults. It helps in determining the appropriateness of the classifier among the users in Afghanistan and what changes should be done subsequently.

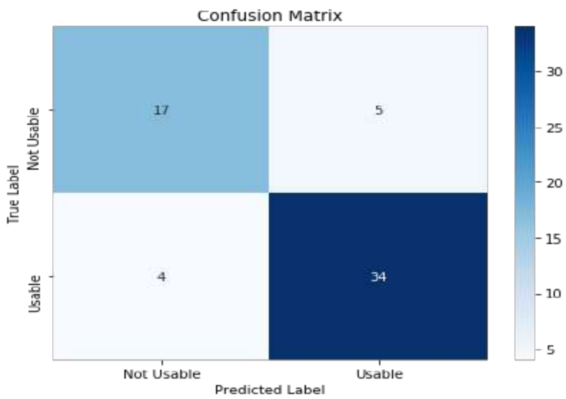

Figure 7. Confusion Matrix

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 09 Issue 08 August 2026
DOI: 10.47191/ijcsrr/V9-i8-27, Impact Factor: 8.048
IJCSRR @ 2026
www.ijcsrr.org

The figure below shows the distribution of Usability. As can be seen from the pie chart, out of the 300 Afghani users who were surveyed using the localized version of the software, two-thirds (about 200 users) indicated that the software was not user-friendly at all. On the other hand, only a few hundred users rated the software as usable. It is quite obvious that there is a lopsidedness here. The conclusion. Most users struggled to use the software and did not have great things to say about it. This skewed distribution is important because it has serious implications for what follows. Since we are going to develop a prediction model based on the data collected, it is important to remember that "not usability" is dominant. It would defeat the purpose if we failed to differentiate between the two groups.

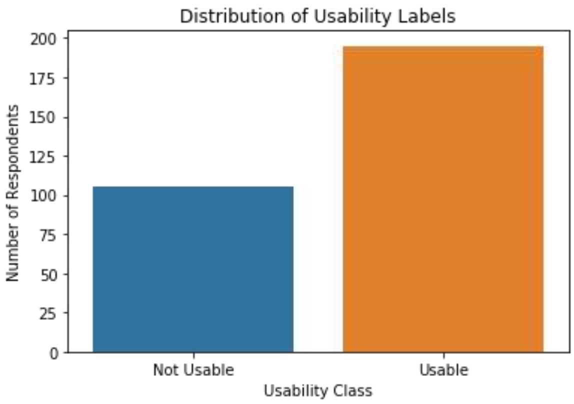

Figure 8. Distribution of Usability

The good performance of Decision Tree can be explained by the ability of the method to consider hierarchical dependencies of usability attributes while preserving the interpretation. As opposed to SVM, which uses the optimization approach, Decision Tree offers clear decision rules, giving developers insights into the usability influencing factors. Support Vector Machine showed good prediction results and performed competitively well in this experiment. The ability of the method to achieve maximum possible class separation accounted for the high level of accuracy. At the same time, SVM is inferior to Decision Tree in terms of interpretability and hence, potential practical applicability in usability engineering. KNN had the worst performance rate of all classifiers used in the experiment; however, the classifier was able to yield an accuracy level exceeding 88%.

Table 3. Classification Report

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Decision Tree	91.4	91.0	92.0	91.0
Support Vector Machine	90.0	89.0	89.0	89.0
K-Nearest Neighbors	88.7	88.0	88.0	88.0

CONCLUSION AND RECOMMENDATIONS

In this study, a comparative machine learning approach was applied to evaluate the usability of localized software systems in Afghanistan. The current investigation was triggered by the necessity for developing effective, objective, and scalable methodologies to assess the usability of computer applications in cases where conventional usability assessment approaches could be insufficient. A dataset based on users' responses to questionnaires was gathered to conduct an experiment and apply three popular classification algorithms: Decision Tree (DT), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). Data preprocessing involved cleaning, encoding, normalization, and exploratory analysis to prepare input data for building predictive models. The performance of classifiers was assessed based on various machine learning metrics such as accuracy, precision, recall, F1-score, and analysis of confusion matrices. The experimental results confirmed the applicability of all three machine learning algorithms to predict usability of software systems. Nonetheless, considerable differences in the ability of algorithms to predict usability outcomes were identified. Specifically, Decision Tree algorithm revealed higher prediction accuracy than other classifiers and reached the value of 91.4%. At the same time, Support Vector Machine obtained a slightly lower accuracy rate of 90.0%, while K-Nearest Neighbors achieved only 88.7% accuracy. The decision rules obtained from generation offer significant insight into usability factors that affect software quality and user satisfaction. Moreover, the transparency of Decision Tree models makes them very efficient tools for identifying factors behind usability problems, which can be of great importance for software developers and decision makers. Support Vector Machine, in turn, exhibited high predictive power and provided competitive results. The algorithm was capable of classifying usability classes accurately and performed reliably throughout the testing dataset. Even though its accuracy was somewhat lower compared to Decision Tree, SVM still showed itself to be a solid classifier model and can serve as a good alternative for future usability classification tools. K-Nearest Neighbors was able to perform adequately; however, it took third place among other models tested in this project. In conclusion, it is clear that the results obtained from this study are consistent with the idea that automated usability evaluations through machine learning approaches are feasible and effective, making them viable alternatives to conventional usability assessments. The proposed approach decreases reliance on usability experts, cuts evaluation costs, and allows organizations to evaluate their software through objective measures. Moreover, the results indicate that localized software usability can effectively be predicted using questionnaires and supervised learning approaches. This study has made valuable contributions to the areas of Human Computer Interaction, Software Engineering, and Machine Learning by proposing a framework for comparative prediction of localized software usability. Practical suggestions have been provided to software development firms, universities, and government bodies involved in software localization and improvement efforts. It is concluded that Decision Tree can be considered an ideal classifier for usability evaluation of localized software applications based in Afghanistan due to its high prediction accuracy and practical implementation considerations.

REFERENCES

1. International Organization for Standardization (ISO), “ISO 9241-11: Ergonomics of Human-System Interaction Part 11: Usability: Definitions and Concepts,” Geneva, Switzerland, 2018.
2. A. Oztekin, D. Nikov, and D. Zaim, “UWIS: An Assessment Methodology for Usability of Web-Based Information Systems,” Journal of Systems and Software, vol. 86, no. 8, pp. 2039–2052, 2013.
3. T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed. New York, NY, USA: Springer, 2009.
4. C. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.
5. K. P. Murphy, Machine Learning: A Probabilistic Perspective. Cambridge, MA, USA: MIT Press, 2012.
6. A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd ed. Sebastopol, CA, USA: O’Reilly Media, 2022.
7. S. Raschka and V. Mirjalili, Python Machine Learning, 3rd ed. Birmingham, U.K.: Packt Publishing, 2019.
8. J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 4th ed. Burlington, MA, USA: Morgan Kaufmann, 2022.
9. L. Breiman, J. Friedman, R. Olshen, and C. Stone, Classification and Regression Trees. Belmont, CA, USA: Wadsworth International Group, 1984.
10. T. Cover and P. Hart, “Nearest Neighbor Pattern Classification,” IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, 1967.
11. C. Cortes and V. Vapnik, “Support-Vector Networks,” Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
12. N. Cristianini and J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge, U.K.: Cambridge University Press, 2000.
13. I. H. Witten, E. Frank, M. Hall, and C. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4th ed. Burlington, MA, USA: Morgan Kaufmann, 2017.
14. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Hoboken, NJ, USA: Pearson, 2021.
15. M. Liu, “Artificial Intelligence Applications in Usability Evaluation: A Systematic Review,” International Journal of Human–Computer Interaction, vol. 41, no. 2, pp. 145–162, 2025.

16. M. Rahman and S. Ahmed, "Machine Learning-Based Usability Prediction for Web Applications Using Decision Tree Classification," Journal of Software Engineering Research and Development, vol. 11, no. 3, pp. 88–101, 2023.
17. A. Basri, N. Abdullah, and M. Hassan, "Comparative Analysis of Machine Learning Algorithms for Usability Sentiment Classification," International Journal of Advanced Computer Science and Applications, vol. 15, no. 4, pp. 321–330, 2024.
18. H. Shah, F. Khan, and M. Tariq, "Automated Software Usability Assessment Using Supervised Machine Learning Techniques," IEEE Access, vol. 10, pp. 102345–102357, 2022.
19. M. Asghar, S. Ullah, and R. Ahmad, "Intelligent Mobile Learning Usability Prediction Using Support Vector Machine and Genetic Algorithm," Computers & Education: Artificial Intelligence, vol. 3, pp. 100071, 2022.
20. I. Sommerville, Software Engineering, 10th ed. Boston, MA, USA: Pearson, 2016.

Cite this Article: Dilawar, A.J., Sabawoon, I., Zaki, U. (2026). A Comparative Machine Learning Approach for Usability Evaluation of Localized Software in Afghanistan. International Journal of Current Science Research and Review, 9(8), pp. 4487-4498. DOI: https://doi.org/10.47191/ijcsrr/V9-i8-27