User Classification Model for Quality of Web Experience

Author Nnodi J.T.[*], Asagba P.O.[**], Ugwu C [***]

[*] **Department of Computer Science, Faculty of Computing University of Port Harcourt**

DOI: 10.29322/IJSRP.12.05.2022.p12541
http://dx.doi.org/10.29322/IJSRP.12.05.2022.p12541

Paper Received Date: 20th April 2022
Paper Acceptance Date: 5th May 2022
Paper Publication Date: 14th May 2022

Abstract- In this paper, a machine learning model was developed for classifying users' quality of experience (QoE) on the web. Key Performance Indicators (KPIs) were extracted from Quality of Web Service (QWS) dataset generated using Principal Component Analysis (PCA) algorithm. The quality of web service dataset was trained using random forest algorithm of different tree sizes. The model was used to develop an application capable of classifying the users' quality of experience on the web in order to predict the user's experience based on the website of interest and the system was implemented in python programming language. The performance of the model was also evaluated using other existing models such as classification and regression trees (CART) and support vector machines. The results obtained showed that when dimensions of data are reduced through feature extraction techniques such as PCA, the most important information were kept by selecting the principal components that explained most of the relationships among the features and that PCA also reduces the dimensionality of data without losing information from any feature. From the evaluation results obtained, all the algorithms achieved a percentage accuracy of 85 and above which is a very good performance. The results also show that Support Vector Machine generated result with the least percentage accuracy on the task of quality of experience prediction while random forest and CART algorithms performed better than others. From other testing parameters used, it was also discovered that accuracy and precision had more impact on quality of experience prediction than sensitivity and other parameters used and had a very important influence on the web quality of experience measurement.

Key Words: classification, influence factors, predictive modelling. Quality of experience, user, Web.

I. Introduction

Traditional monitoring and management approaches of networks based only on quality of service (QoS) optimization are not sufficient to ensure user's needs [1]. This led to investigation into the new concept called users' quality of experience to evaluate the real quality of experience perceived by the users. There are several metrics, called Quality of Experience Influence Factors (QoE IFs), which can affect the perceived quality by the user. These factors are closely related to human perception and could serve as more valuable quality indicators for all system's actors (user, service and network provider, e.t.c). From the users' side, it ensures that he perceives the best service regardless their mobility and their context. From the provider's side, it helps them to provide, restore and ensure the best service to their users, and to decrease the error rate and increase their profit. Although many works have addressed QoE classification, but the concept is still hard to investigate especially in mobile environment and also because QoE is used in several context including video, gaming, voice over internet protocol (VoIP). In this paper, user classification model was developed for extracting key performance indicators accurately from quality of web service dataset generated using PCA algorithm.

When Quality of Experience (QoE) is mentioned in the telecommunication sector, people talk about video streaming, peer to peer file sharing, online gaming, cloud storage, cloud based computation, download speed, voice and video call quality e.t.c. The actual service "received" by users determines users QoE as opposed to Quality of Service (QoS) "rendered". Measuring QoE starts from the content providers; content distribution networks (CDNs), Internet Service Providers (ISPs), cross-CDN optimization services and finally converges on the user. A lot of research and infrastructural innovations have gone into making sure that the user receives a better QoE especially, in the design of content delivery systems. Some of these infrastructures include placing caches in front of content servers to return frequently requested content [2]. The robustness of the internet and telecommunication is tied to the communication and infrastructural improvements made over time on content delivery. The challenges of measuring QoE is enormous which includes the increasing complexity of the cable network environment, increasing load on the network, heterogeneity of technologies especially in cellular data networks [3]. Researches have also gone into measuring QoE based on metrics from the end of Content Distribution Networks, Internet Service providers, content providers and cross-CDN optimization service providers. The user who sustains these players with revenue collected directly or indirectly is left out. It is important to measure QoE using metrics from the user's end. Often, mobile phone users have complained about poor QoE received from telecommunication service providers [2]. These complaints have come most times without actual and comprehensive data backing up the claims. Researchers have explained QoE from every perspective (content providers, content distribution networks, internet service providers, cross-CDN optimization services). Yet the users out of these players directly or indirectly are less equipped with the appropriate tools and data to measure QoE from their end

II. RELATED WORKS

Some papers related to classification of web users' experiences were reviewed and are discussed as follows:

[4] analyzed and modeled mobile QoE data from a large group of Finnish users (users from Finland) in the field. Reflecting the development of mobile networks, service quality was high and better than in prior studies but the paper focused only on a single country (Finland), so the results was not generalized to countries with significantly different population or mobile network characteristics.

[5] presented the application of diverse Machine learning techniques in key areas of networking across different network technologies but did not advance the state-of-the-art to finally realize the long-time vision of autonomic networking.

[6] devised a method that predicts the QoE Mean Opinion Score based on machine learning classifier to deduce the best algorithm but the model was limited to M5P algorithm.

[7] carried out a short term study to develop a system that runs on commodity access points (APs) to assist ISPs in detecting when Wi-Fi degrades QoE but in particular, intermittent events are challenging to troubleshoot and require a long-term monitoring approach.

[8] presented an extensive review of the state-the-art research in the area of QoE modeling, measurement and prediction but QoE should be performed over several months to achieve accurate results which the study failed to address.

[9] proposed an initial method for estimating the quality of experience of web services for web service selection using a fuzzy-rough hybrid expert system. The presented how different QoS parameters impact the QoE of web services. For this, they conducted subjective tests in controlled environment with real users to correlate QoS parameters to subjective QoE. Based on this subjective test, membership functions and inference rules for the fuzzy system were derived. Membership functions were derived using a probabilistic approach and inference rules were generated using Rough Set Theory (RST). They evaluated the system in a simulated environment in MATLAB. The simulation results showed that the estimated web quality from system had a high correlation with the subjective QoE obtained from the participants in controlled tests. But feature selection methodologies that could automatically select the most impacting QoS parameters were not included.

[10] proposed an alternative approach to select base classifiers forming a parallel heterogeneous ensemble in order to trim poorly performing classifiers; so that a more effective heterogeneous ensemble can be generated. More specifically, the proposed trimming approach was designed to find an optimal subset of classifiers to form the desired heterogeneous ensemble. To address this issue, the differences in effectiveness between base classifiers forming the ensemble were utilized to spot weak classifiers. For evaluating the proposed approach, eighteen benchmark datasets were used for generating the heterogeneous ensemble classification and comparisons with the state-of-the-art methods were conducted. The experimental analysis demonstrated the effectiveness and superiority of the proposed approach when compared to other state-of-the-art approaches. But the authors failed to investigate alternative approaches to exclude poorly performing classifiers to enhance the performance of heterogeneous ensemble classification.

[11] presented a new prediction model to detect technical aspects of teaching and e-learning in virtual education systems using association rules mining. Supervised techniques are applied to detect efficient QoE factors on virtual education systems. But some meta-heuristic algorithms could be applied to improve the feature selection strategy.

[12] proposed a fast localization iris recognition algorithm which used the iris segmentation algorithm to quickly extract the iris region for recognition but network training will improve the recognition rate of iris network when combined with deep learning.

[13] in another paper investigated the impact of the most widely used preprocessing techniques, with respect to numerical features, on the performance of classification algorithms using deep learning approach but they would have also considered clustering algorithms.

[14] reviewed data preprocessing techniques that were used to analyse massive building operational data but did not consider using semi-supervised learning to fully exploit the hidden values in massive amounts of unlabeled data.

[15] proposed enhanced pre-processing algorithms with feature selection and machine learning and evaluated the algorithms using performance evaluatory measures but failed to experiment classifiers from statistical, neural, fuzzy, genetic algorithms and also, tree families that could enhance the resulting accuracy was not taken into consideration.

But in this paper, a user classification model was developed to classify the user's quality of experience on the web. Key Performance Indicators (KPIs) were extracted from Quality of Web Service (QWS) dataset generated. Principal Component Analysis (PCA) algorithm was used to extract relevant features and reduce the dimensions of the dataset generated. The quality of web service dataset was trained using random forest algorithm of different tree sizes. The model was used to develop an application capable of classifying the users' quality of experience on the web based on the website of interest and was implemented with python programming language.

III. MATERIALS AND METHODS

A. Research Methodology

The research methodology used to develop this model is Object Oriented System Analysis and Design Methodology (OOADM).

This programming paradigm was used because the components are constituents of objects which involve classes and their methods of interaction. The Unified Modeling Language (UML) was employed to visualize the architectural blueprint of the proposed system in diagrams since UML is a component of object oriented system analysis and design.. To develop the proposed system, Quality of Web service dataset with features that impact users' experience of these services (youtube, facebook, VoIP, web browsing, file transfer) were extracted.
The tools used in gathering data include: Quality of experience extraction tool that is capable of extracting parameters from data set obtained from a network operator. These metrics generated are stored and periodically sent to the QoE server for preprocessing and analysis. Data argumentation was also done at this level to get web based Key Performance Indicators (KPIs) which was used for training and prediction. We ended up having a fully labeled QoS/QoE dataset which was used to build and train machine learning based predictors / models.

B. *Design Architecture of the User Classification Model*

Figure 1 shows the architecture of the user classification model developed for web users.

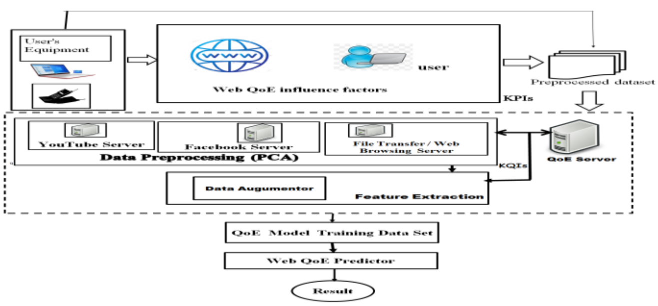

Figure 1: Architecture of User Classification Model for web users

The system architecture consists of the following layers: user equipment, data preprocessing module, Quality of Experience server, feature extractor, model training and the predictive module.
The data capture module/ Users Equipment: This layer is responsible for collecting the data from the user's device as soon as the network is launched. The required parameters are captured for the different webservers (facebook, skype, Youtube, e.t.c.). The data set in table 1 consists of 364 instances of measured key performance indicators (KPIs) obtained from Quality of Web service dataset generated.

QoE Server: This layer holds the data temporarily when packets are sent on a network. It sends and receives requests from various servers (youtube, facebook, Skype, whattsapp e.t.c) and records them depending on the web site the user wants to visit. It then passes the QoE metrics measured for preprocessing.

Data preprocessing was also carried out to constructively pattern it to be trainable with random forest and also to extract the relevant features from the dataset. The data is preprocessed using different techniques and libraries in python, which converts the data into binary numbers.

The following preprocessing steps were used to scale the quality of web service dataset obtained before training with machine learning models.

1. Load data set
2. Import libraries
3. From sklearn.preprocessingimport MiniMaxScaler
4. Set data link
5. Data parameters
6. Prepare dataframe using the given link and define
7. Separate array into input and output
8. Initialize the MinMaxScaler
9. Learn the standard parameters for each of the data and transform.
10. Summarise transformed data

Table 1 :Quality of Web Service Dataset

S/N	Response Time	Availability	Throughput	Successability	Reliability	Latency	WSRF
1	45	83	27.2	50	97.4	43	100
2	71.75	100	14.6	88	85.5	64.42	93
3	117	100	23.4	83	88	111	90
4	70	100	5.4	83	79.3	63	90
5	105.2	100	18.2	80	92.2	104.6	90
6	224	100	24.6	83	80	223	90
7	99.2	100	13.7	80	76.3	62.4	89
8	108.2	100	16.8	80	90.7	108	88
9	125.2	100	16.4	80	89.2	125	88

The transformed data was stored in Quality of experience server which was fed into the decision trees for training the random forest classifier. In training the dataset, random forest was used which involved using the extracted features as input to train a QoE model. The accuracy of the model was evaluated using appropriate testing parameters like accuracy, precision, specificity and sensitivity. The model is useful for real time quality of experience predictions. To classify the user's experience, the correlation between KPI and KQI enables the model to predict the unknown KQI from the known KPI. The system classifies the quality of experience of web users in two ways:

a. When a user wants to predict his experience from a live website, he simply extracts parameters from the website and then provide the web address of the site he wants to visit. The system then returns the estimated QoE class of the site in question.
b. The user can equally predict his experience by using the dataset already captured from different websites to predict the quality of experience of the user on that website. The result from this module can now be compared between objective parameters QoE score and subjective QoE to know how accurate the estimation is. The result of this module can equally be used to send report to network operators or ISPs upon request.
The last is the User's Interface which is the main interface where inputs of parameters are done. It comprises of the graphic user interface of the system which the user can see to make inferences and decisions pertaining to his opinion on web experience.

B *Model Building*

The first goal of this paper was to scale the data from the original range so that all the values are within the new range 0 and 1, while the end goal is

to develop user classification model for classifying the user's web experience into four categories. Class 1 represents very good experience, class 2 represents good experience, and class 3 represents fair experience while class 4 represents poor experience as the case may be. To achieve the first goal, scikit-learn object called MinMaxScaler was used to normalize the dataset in table 1 to obtain the scaled dataset shown in Table 2.

Table 2: Normalised Quality of Web Service Data

S/N	Response Time (s)	Availability	Latency	Jitter	QoS level
1	68	0.00	1.00	0.20	1
2	78	0.00	1.00	0.20	1
3	83	0.00	0.00	0.80	1
4	85	0.00	0.70	0.20	1
5	71	0.00	0.30	0.10	1
6	65	0.00	0.30	0.30	1
7	61	0.00	0.00	0.40	1
8	76	0.10	0.30	0.20	1
9	70	0.00	0.30	0.10	1

After Rescaling, table 2 shows the output with all the values in the range between 0 and 1. The training dataset in Table 2 contains the most relevant features that affect the web quality of experience of internet users.

http://dx.doi.org/10.29322/IJSRP.12.05.2022.p12541

ISSN 2250-3153

C PCA Based Feature Selection

The main goal of this paper was also achieved by building the QoE feature extraction model using the Principal Component Analysis (PCA) algorithm and training it with random forest and then implemented in python. This was necessary to ensure that the model built was accurate and must be capable of transforming large volume of the data set into principal components to make it less prone to errors. The result of the PCA transformation of the dataset is shown in table 3. A PCA based feature extraction model of web quality of experience was developed by extracting features from the normalised dataset shown in table 2. The PCA algorithm also helped in dimensionality reduction in order to enhance the accuracy of the model. The feature extraction module was developed to extract these relevant data which was trained with random forest algorithm. The columns Response time, Availability, Latency, and Jitter are the features which represent inputs to the model while the column "QoS level" is the target variable or the classes. The Principal Component Analysis Algorithm was used in extracting the relevant data. Similarly, the Random Forest algorithms on SPM 8.2 was used to train the model with the results compared with other existing models.

The detailed steps employed in the feature extraction algorithm are as follows;
Step 1: For any two feature vectors of the training data get their matrixes
Step 2: Calculate the summation of the matrixes for the two feature vectors of the training data.
Step 3: Calculate the covariance matrix of the feature vectors. The covariance matrix is given by

$$Cov(Matrix) = \frac{1}{(n-1)}\sum (x-\mu)(x-\mu)^T. \quad (1)$$

Where n is the number of training samples, x is the corresponding vector values, U is the mean of the training data and T is the transpose of the matrix
Step 4: Find the eigen values and the corresponding eigen vectors

$$AX = \lambda X \Rightarrow \lambda_1 \lambda_2 \quad (2)$$

AX = λX
Where A is the eigen value, λ is the corresponding eigen vector and X is the corresponding vector values.
Step 5: When the eigen vector corresponding to the largest eigen values is obtained, then the first is referred to as the first principal component else if they are equal to each other, both feature vectors are taken.
Step 6: Continue testing the feature vectors until all the significant eigenvectors with the highest eigen values are gotten. The subsequent ones are called principal component PC(2), PC(3)…, PC(n).
Step 7: Reducing the dimensions of the data set
The last step in performing PCA was to re-arrange the original data with the final principal components which represent the maximum and the most significant information of the data set. In order to replace the original data axis with the newly formed principal components, we simply multiplied the transpose of the original data set by the transpose of the obtained feature vector.
The output from the feature extraction using PCA is shown in table 3. After the PCA, the new data has been reduced to two features as shown in the table with the same number of rows as the original features. This shows the strength of using PCA to extract relevant features from a large dataset as well as for dimensionality reduction.

Table 3: Output features from PCA

Principal Component 1	Principal Component 2
9.19283683	1.94858307
2.3878018	-3.76817174
5.73389628	-1.0751738
1.25617928	-1.90229671

IV IMPLEMENTATION

The implementation was done by extracting parameters from 346 websites and training them in order to classify the experience of the users into the four categories – very good, fair, poor and very poor experiences.

*A **Results***

Figures 2 – 10 show the details of the results obtained from the modelling techniques and the comparison with existing algorithm.

Classification and predictions were performed by loading the dataset into the user's device. This is simply achieved by clicking the extract parameter icon on the home page of the user classification system. This enables the system to locate various web servers available on the internet to extract parameters relating to quality of web experience. The system prompts the user to type a web address of interest to view his or her experience.

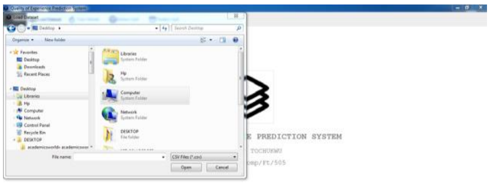

Figure 2: Dataset Loading dialogue

Figure 3 : QoE **parameter extraction interface**

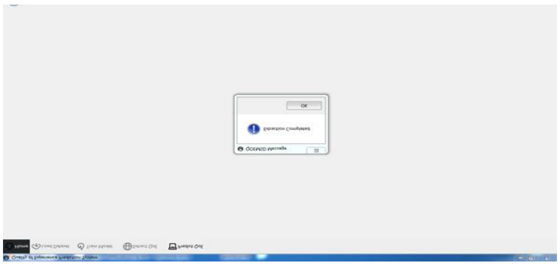

Figure 4: Feature extraction completion confirmation page

Figure 5: QoE **prediction prompt for multiple website**

Figure 6: QoE **prediction prompt**

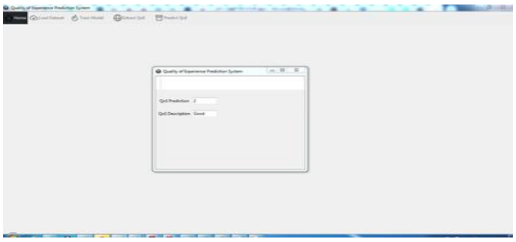

Figure 7 : QoE **prediction output for single website**

Figure 8 : QoE **prediction output fo multiple websites**

B Discussion of Results

In order to predict the user's QoE from different websites, using the parameter extraction feature of the system, network parameters were captured for 346 websites according to the dataset model designed in the system. The websites were assessed over a period of 50 days spread over the different times of the day. Similarly, the parameters were captured over different Internet provider's network namely MTN, Glo Mobile, Airtel and Etisalat using 3G, 3.5G and 4G networks respectively.

The outputs from user classification model for quality of experience of web users are shown in figures 2 - 8. Figures 2 and 3 shows the website QoE parameter extraction interface and the parameter extraction completion confirmation dialogue respectively. Figure 4 shows the parameter extraction output and figure 5 is the QoE prediction prompt for a multiple website, while figure 6 shows the QoE prediction prompt for a single website.

Similarly, figures 7 and 8 show the QoE prediction output for a single website and QoE classification output for multiple websites respectively.

C MODEL EVALUATION The accuracy metric plot of the model against that of CART Algorithm using Gini index and Information gain as splitting functions for test proportions of 0.2, 0.3 and 0.4 as shown in figure 9 shows that the QoE model performed at accuracy level of 93 % and above while support vector machine had the least performance accuracy of 82 at 0.30 test proportion.

Figures 10 also shows the plot of precision of the model compared to that of existing models for test proportions of similar values. It can be observed in the figure that the QoE model consistently performed better than models built on both CART algorithm and support vector machines in terms of precision.

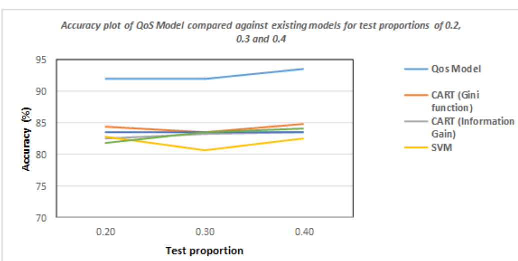

Figure 9 : Accuracy of QoE Model against existing models

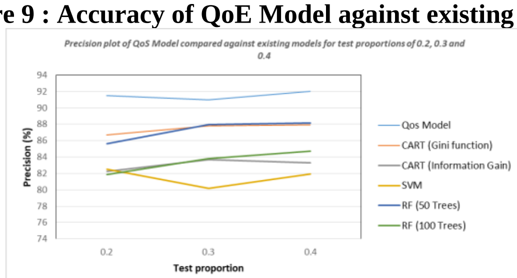

Figure 10 : **Precision of QoE Model against existing models**

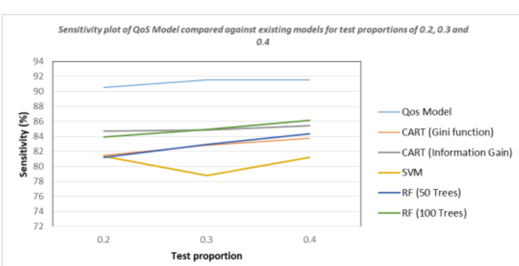

Figure 11: Sensitivity of QoE Model against existing models

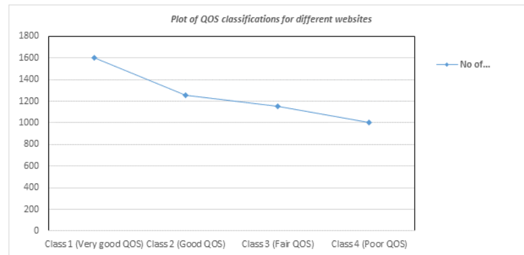

Figure 12: **Plot of QoE classification for multiple websites**

Figure 11 shows that the in terms of sensitivity; the QoE model still performs better than other models using the same test proportions of 0.2, 0.3 and 0.4 respectively.

From the graph shown in figure 11, it can be observed that out of the 1800 websites analysed, about 1200 -1600 users would experience very good service, 115- 1200 would be good while 1000 – 1200 users would have fair experience and 1000 would have poor experience on the web at the time they were analysed.

5. *Conclusion*

User classification model for web quality of experience was designed, developed, and trained for web QoE prediction with highly accurate results. From the PCA usage as feature extraction technique in QoE modeling, most of the tests that form feature vectors are mostly high dimensional data which were reduced by feature extraction module before using the random forest classifier to train the data. Also, dimensionality reduction of data was also necessary in order to increase the amount of data and reduce over fitting during training the model

ACKNOWLEDGMENT

First and for most, we thank the Almighty God the author of all wisdom for his grace and provision in starting and finishing this research. We also extend our appreciation to friends, colleagues and family members who supported us to ensure the progress of this research.

REFERENCES

[1] Amour, L., Sami, S., Hoceini, S., and Mellouk, A. (2015). Building a Large Dataset for Model-based

QoE Prediction in the Mobile Environment. Proceedings of the 18th ACM International Conferenceon Modeling, Analysis and Simulation of Wireless and Mobile Systems (pp. 313-317). Cancun, Mexico : ACM
[2] Balachandran, A. (2014). *Large scale data analytics of user behavior for improving content delivery*.carnegie-mellon univ pittsburgh pa school of computer science.

[3] Serral-Gracià, R., Cerqueira, E., Curado, M., Yannuzzi, M., Monteiro, E., & Masip- Bruin, X. (2010). An overview of quality of experience measurement challenges for video applications in IP networks. In *International Conference on Wired/Wireless Internet Communications* 252- 263. Springer, Berlin, Heidelberg.
[4] Boz, E., Finley, B., Oulasvirta, A., Kilkki, K., & Manner, J. (2019). Mobile QoE prediction in the field. *Pervasive and Mobile Computing*, *59*, 101039.

[5] Boutaba, R., M. Salahuddin, and N. Limam (2018): A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. *J Internet Serv Appl* **9, (**16) . https://doi.org/10.1186/s13174-018- 0087-2
[6] Abar, T., Ben – Letaifa A., and S. El Asmi (2020): 'Quality of experience prediction model for video streaming in SDN networks', Int. J. Wireless and Mobile Computing, 18(1), 59–7.

[7] Horad D. (2018). Narrowing the gap between QoS metrics and web QoE using Above- The-Fold metrics. *International Conference on Passive and Active Network Measurement* . 1-13. Berlin: Germany.

[8] Nadhom, M., & Loskot, P. (2018). Survey of public data sources on the Internet usage and other Internet statistics. *Data in brief*, *18*, 1914-1929.
[9] Pokhrel, J., Lalanne, F., Cavalli, A., & Mallouli, W. (2014). QoE estimation for web service selection using a fuzzy-rough hybrid expert system. In *2014 IEEE 28th International Conference on Advanced Information Networking and Applications* 629-634. IEEE.
[10] Alshdaifat, E. A., Al-hassan, M., & Aloqaily, A. (2021). Effective heterogeneous ensemble classification: An alternative approach for selecting base classifiers. *ICT Express*, *7*(3), 342-349.

[11] Tan, C., and Lin, J. (2021). A new QoE-based prediction model for evaluating virtual education systems with COVID-19 side effects using data mining. *Soft Computing*, 1-15.
[12] Alshdaifat, N. F. F., Talib, A. Z., & Osman, M. A. (2020). Improved deep learning framework for fish segmentation in underwater videos. *Ecological Informatics*, *59*, 101121.

[13] Alshdaifat, E. A., Ala'a Al-shdaifat, A. Z., & Aloqaily, A. (2020). The Impact of Data Normalization on Predicting Student Performance: A Case Study from Hashemite University. *International Journal*, *9*(4).
[14] Balachandran, A., V. Sekar, A. Akella, S. Seshan, I. Stoica & H. Zhang, (2013). Developing a predictive model of quality of experience for internet video. In *ACM SIGCOMM computer Communication Review* . 43,339–350. Hong Kong, China.
[15] Banumathi, S., & Aloysius, A.(2019). An Enhanced Preprocessing Algorithms and Accuracy Prediction Of Machine Learning Algortihms. *International Journal of Advanced Research in Computer Science.*8 (8)

AUTHORS

First Authour - Nnodi Joy Tochukwu, B.Sc(UNIZIK), M.Sc, Ph.D(UPH). nnodi.joy@gmail.com

Second Authour – Prof. Asagba Prince.O, B.Sc (UNN), M.Sc, Ph.D(UPH). University of Port Harcourt, Choba Nigeria
asagba.prince@uniport.edu.ng

Third Authour – Prof. Ugwu Chidiebere, B.Sc,M.Sc, Ph.D(UPH). University of Port Harcourt Choba Nigeria.
chidiebere.ugwu@uniport.edu.ng

Correspondence Authour – Nnodi JoyTochukwu,
nnodi.joy@gmail.com, perpma2003@yahoo.com, +234 - 8064369582