EMPIRICAL RESEARCH PRESS

International Journal of Engineering and Applied Computer Science

Volume: 04, Issue: 04, May 2022 ISBN: 9780995707545

Stacked Generalization of Random Forest and Decision Tree Techniques for Library Data Visualization

Stanley Ziweritin

Department of Estate Management and Valuation, Akanu Ibiam Federal Polytechnic,
Unwana-Afikpo, Ebonyi State, Nigeria.

10.24032/IJEACS/0404/005

Empirical Research Press Ltd.

London, United Kingdom

www.ijeacs.com

Stacked Generalization of Random Forest and Decision Tree Techniques for Library Data Visualization

Abstract— **The huge amount of library data stored in our modern research and statistic centers of organizations is springing up on daily bases. These databases grow exponentially in size with respect to time, it becomes exceptionally difficult to easily understand the behavior and interpret data with the relationships that exist between attributes. This exponential growth of data poses new organizational challenges like the conventional record management system infrastructure could no longer cope to give precise and detailed information about the behavior data over time. There is confusion and novel concern in selecting tools that can support and handle big data visualization that deals with multi-dimension. Viewing all related data at once in a database is a problem that has attracted the interest of data professionals with machine learning skills. This is a lingering issue in the data industry because the existing techniques cannot be used to remove or filter noise from relevant data and pad up missing values in order to get the required information. The aim is to develop a stacked generalization model that combines the functionality of random forest and decision tree to visualization library database visualization. In this paper, the random forest and decision tree techniques were employed to effectively visualize large amounts of school library data. The proposed system was implemented with a few lines of Python code to create visualizations that can help users at a glance understand and interpret the behavior of data and its relationships. The model was trained and tested to learn and extract hidden patterns of data with a cross-validation test. It combined the functionalities of both models to form a stacked generalization model that performed better than the individual techniques. The stacked model produced 95% followed by the RF which produced a 95% accuracy rate and 0.223600 RMSE error value in comparison with the DT which recorded an 80.00% success rate and 0.15990 RMSE value.**

Keywords- Data Visualization, Decision Tree, Random Forest, Stack.

I. Introduction

Data visualization is the method employed to represent data that can help users understand and interpret the structure of data [1],[2]. It can help transform numerical and categorical data into a graphical or visual form that makes it easier for researchers to visualize outliers and hidden data trends [2],[4]. The insight about data patterns in a database may be unnoticed but when visualized using charts and graphs will be easier to view details about the behavior of data in a database as stored in tabular form [5]. The modern visualization tools help users to quickly understand, interpret data over time, and make necessary adjustments to different variables for decision making [6]. There are several data mining techniques adopted by Adediran and Ajibade[7] suitable to effectively and accurately visualize data stored in a multi-dimensional form using line, bar, pie, bubble, and donut charts[8]. The existing methods of data visualization are faced with numerous challenges when it comes to complex and dynamic data structures and patterns with high dimensional space for interpreting the behavior of noisy, training, and testing data. The existing techniques lack merits and cannot be used to remove or filter noise from data and pad up missing values in order to get the required information. The Big data age can cause enormous growth in database size with respect to time which may affect the functionality of existing techniques. Data visualization can quickly help at a glance communicate details of data in a graphical form.

This work aims to develop a stacked generalization model capable of combining the functionalities of random forest (RF) and decision tree (DT) to effectively visualize library data. The RF and DT technique is employed to effectively visualize a large volume of school library data for the purpose of decision making. The proposed system will be implemented using Python programming language that can learn from noisy data and help visualize detailed insight about the behavior of training and testing data with its relationship. The model will be developed, trained, and tested to learn and extract hidden data patterns using a cross-validation test. The output of the RF and DT model will be used as the new training dataset for the stack model at its base level. It combines the functionalities of both models to form a generalized stacking technique that can perform better than using the individual models with the help of a cross-validation test. The stacking stage will contain the RF at the top of the stack followed by the DT tree. This work will be useful to consultants, research libraries, institutions, and scholars. It provides user speed and accuracy with the ability to act on visual findings for better decision-making.

The organization of this paper is divided into different sections as followings: section 1 contained the introduction, section 2 presents a brief review of previous approaches relating to the study area and the gap in exploring the proposed model; Section 3, introduces materials and methods employed for developing the model; Section 4, focuses on the results and detailed discussion of results; Section 5 presents the conclusion to the paper.

II. Related Works

Nazeer et al.,[9] compared different data visualization techniques with a sizable dataset obtained from a self-study questionnaire. From the analysis; 90% of the respondents are in favor of adopting data visualization techniques and recommended the use of modern data visualization tools like histograms, pie, and bar charts. Moore [10] reviewed the history of data management systems that leads to the problems of employing Bigdata visualization techniques that require advanced machine learning visualization tools to provide better insight into data. The created platform was effective in communicating to the user relevant information necessary for making a better decision. Narayanan and Shanker [11] recommended the use of some selected data visualization techniques to research scholars in order have a better foundation of business intelligence. The visualization techniques include network diagrams, box plots, correlation matrix, donuts, pie and bar charts, and line and bubble plots. Plank and Helfert [12] discussed the adoption of interactive Bigdata visualization techniques to provide managers with better policies in making decisions. The research findings enhanced user understanding, and knowledge and provided valuable insights about data in the organization. Zhang [13] discussed about the hierarchical view of different authoring systems using the Kyrix platform with a decision tree for data visualization. It reduced the barriers of entries with five (5) different components: namely layers, canvas view, transformers and jumps for zooming. It provided a more interactive user platform with the possibility of adding new and edited icons (button). A graphic user interface was developed to help add layers with the help of a button click which helps in keeping track of the application scripts. The Kyrix platform could not be used to visualize data behavior, training, testing data, and relationships that exist between database attributes. Ogier and Stamper [14] proposed the use of modern data visualization tools in offering services to research libraries and research scholars. The design strategy follows a top-down approach that cut edges across concepts, refining, ethics, and layouts. The concepts deal with the audience and intend, refining used to tire pieces of data that relate to the whole. The design layout focuses on data visualization techniques, framework, and communication but there was no practical implementation. Chawla et al.,[15] identified Bigdata visualization techniques into three (3) different groups namely: volume, variety, and dynamics of data. The decision tree was employed to visualize data in a hierarchical form using the divide and conquer technique with the help of a decision rule to insert sub-nodes into root and parent notes. The hierarchical model suffers from the limitations of distorted positive and zero-pixel values. Butavicius and Lee [16] carried out empirical research using four data visualization techniques to showcase the similarities of 50-unstructured short text messages stored in one-dimensional space. The greedy nearest neighbor with the ISOMAP technique was employed as a base representation to rank the one-dimensional list using two (2)-dimensional visualization and multi-dimensional data scaling (MDS). The MDS display was better than the ISOP technique and the 2-Dimension display over 1-Dimensional. But recorded significant variations on different dimensionalities of data storage. Gorodov and Gubarev [17] researched on different Bigdata visualization techniques in relation to noise, large perception, and misclassified patterns. The visualization techniques revealed the dynamics of changes that occurred in stored data over time in terms of volume, format, and dimension using the Tree-Map method. But the model suffers from varying scales and selection of suitable methods among others for Bigdata analysis and visualization. Tay et al.,[18]. Proposed the use of data visualization techniques in organizational research for Bigdata analysis considered to be in high volume and shows how responses are graphically represented using line plots and charts. But identified issues of data integration that help combine different data modes in revealing details about interest or phenomenon and interactivity to uncover and identify hidden and new data patterns.

III. Materials and Methodology

This work focuses on the use of RF and DT methods of visualization with the stacking concept to combine the functionality of both methods in forming a better technique that performs better than the individual models. A cross-validation test is employed to help learn with noisy data and generalize well using the testing dataset. This can help overcome the problem of model over-fitting. The proposed data visualization techniques like a heat map, functional graph, ROC, AUC, RF, DT, error bar plots, and bar charts are employed with machine learning (ML) visualization libraries in Python. The bar charts are used to showcase the training and testing dataset, and error bar plots with ML model for sensitivity rate against different data samples. The Heat map as a diagnostic tool is used to visualize the correlation between attribute pairs and missing values measured in two dimensions. The functional graph provides a clear insight into training, testing, and validation set over noisy and missing data values. The Area under curve (AUC) graph visualizes the variation of model performance or learning rate using training and cross-validation test in terms of accuracy over sample data size. The decision tree is used to organize data in a normal tree-like structure with nodes and sub-nodes (leaf nodes) using the decision rule. The tree size after construction is always proportional to the data values or points. The receiver operating characteristics (ROC) is used to visualize the performance and the relationship between hit rate and false-positive rate in revealing algorithm tradeoff.

A. Dataset

The experimental data was sourced from a public library made available on data.world "Source: https://data.world/datasets/public-library" containing attributes of Book_ID, Title, ISBN, ISBN1, Average ratings, Number of pages, Rating values, language used, publisher name, Publication date, and the Authors name, etc. as shown in Table I. The dataset was divided into 80% ($\frac{80\ x\ 45640}{100}$=36512) training and 20% ($\frac{20\ x\ 45640}{100}$=9128) testing sets with data attributes.

TABLE I. Library Book information system dataset

	Title	- - -	Publisher	Ratings
0	Harry Potter and the Half-Blood Prince (Harry Potter #6)	- - -	Scholastic Inc.	2095690
1	Harry Potter and the Order of the Phoenix (Harry Potter #5)	- - -	Scholastic Inc.	2153167
2	Harry Potter and the	- - -	Scholastic	6333

	Chamber of Secrets (Harry Potter #2)			
- - -	- - -	- - -	- - -	- - -
45637	The Picture of Dorian Gray	- - -	W. W. Norton & Company	663
45638	The Picture of Dorian Gray	- - -	Oxford University Press USA	1089
45639	An Arab-Syrian Gentleman and Warrior in the Period of the Crusades: Memoirs of Usamah Ibn-Munqidh	- - -	Columbia University Press	68
45640	Technical Manual and Dictionary of Classical Ballet	- - -	Dover Publications	393
45641	The Ballet Companion: A Dancer's Guide to the Technique Traditions and Joys of Ballet		Touchstone	524

B. Data Transformation

Data transformation is the advent of converting row data into a format, and structure that is suitable for model building using the Load, transform, and extraction techniques [19],[20]. The transformation stage help normalize numeric features to produce a better model and allows regression techniques like the random forest to have nonlinearity features in its forest space [21].

C. Feature Extraction

Feature extraction is mandatory for any application that involves relevant feature identification from a database. It is a process of extracting features to assist the task of classifying data patterns [22]. This phase is considered to be important because it can influence classification and regression tasks in adopting ML tools.

D. Classification

Classification is one of the keys and useful components involved in the decision-making process that categorizes data based on some observed features using particular criteria [23]. The row and class-feature sampling techniques are used for each and every decision tree in the forest to reduce bias, noise, and high variance [24]. The change in the input dataset caused low variance in the tree and accuracy with majority votes for the binary classification model [24].

E. The Decision Tree

The construction of DT follows the concept of divide and conquer by splitting the source dataset into subdivisions called sub-nodes based on the test value [26],[27]. And this process is repeated on each subset and recursion is completed and terminated when splitting no longer adds value to the predictions. In DT, data arrives in the form of records [28],[29] written as:

$$(x,Y) = (x_1, x_2, x_y \dots, x_k, Y) \qquad 1$$

Where Y is the dependent variable referred to as the target variable while x is the independent variable with a vector containing the input variables X1, X2, X3 ....., Xn etc. The DT uses a decision rule characterized by an ordered pairs [30],[31].

F. RF Regression Model

The scaled training and testing dataset was feed to the fitted RF model using the regression class and classification of the sklearn ensemble library, trained and tested with specific number of n_estimators (decision-trees) as fine-tuned parameters and random states set to be 0, maximum number of leaf node to 100 and n_job parameter = -1 to obtain an optimal solution with the adjustable parameter to clearly visualize database relationships, noisy data content and over-fitting features. The mode training process was done using Python code: model.fit (x_train, y_tes data set). The output of the RF model was then fed as input to the stacking model using a cross-validation test to reduce and filter the noisy data content from the database system. The nsemble library was employed to classify objects into similar groups in constructing multiple decision trees in the forest space. And proposed to fine-tune or adjust the n_estimators in the forest class.

G. The Stacking Model

It is also called stacked generalization is a technique used to combine the functionality of two or more techniques to form a model that performs better than using the individual learning models. The two different learners are combined to build an intermediate prediction model, one prediction for each learning model that learns from intermediate patterns with the same target variable. The final model is said to be stacked using the RF on top of DT. It improves the overall performance and often ends up performing better than individual intermediate models at the base level as shown in Figure 1.

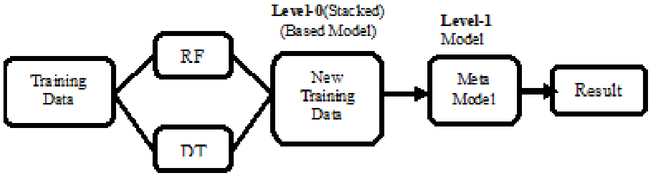
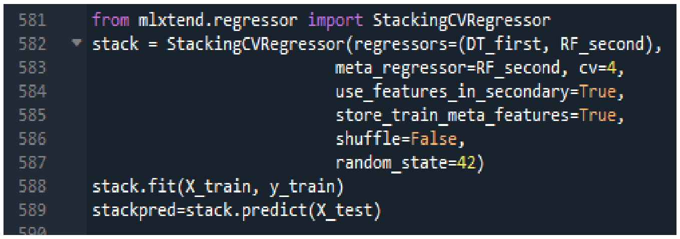


Figure 1. The Stacking Model

The stacked model architecture combined RF and DT techniques often referred to as the base model at level-0. The functionalities of both are used as at the base level to build a meta-learning model called the stack. The random forest is placed at the top of the stack followed by a decision tree as shown below in the code segment using Python programming language.



```
from mlxtend.regressor import StackingCVRegressor
stack = StackingCVRegressor(regressors=(DT_first, RF_second),
                            meta_regressor=RF_second, cv=4,
                            use_features_in_secondary=True,
                            store_train_meta_features=True,
                            shuffle=False,
                            random_state=42)
stack.fit(X_train, y_train)
stackpred=stack.predict(X_test)
```

H. *New Training Data*

The training dataset was divided into k-folds cross validation and fitted using the base model on k--1 path of the whole training set to compute its performance using the test dataset with predictions made for the kth part. This process is repeated and predictions from training set are used as features for training the stacked model and used to visualize the testing dataset.

I. *Meta Model*

Meta model at level-1 is a meta-layer that accepts output from the base models (first level) as the new training data. The stacking regressor class was invoked from mlxtend library that contains the stacking cross validation regressor in Python and RF stacked on top DT model. The meta_regressor was set to point at RF, CV=4, use_features in secondary=rue, shuffle=False, random_state=42) and stack build with the code sedment: Stack.fit(x_test, y_train).

Algorithm 1: The RF Algorithm

Step	Processes involved
1	Start
2	Assume cases in the training set to be C and randomly select cases with replacement
3	If there are inputs of M features with a variable representing number m<M. Then Take the best split on the node and peg the value of "m" to be constant
4	Grow each decision-tree to have the largest possible size without pruning
5	Use majority vote for classification and mean for regression task
6	Stop

Algorithm 2: Decision Tree (DT)

Step	Processes involved
1	Start
2	Compute class frequency value (CCFT) and return a leaf_Node
3	Create a decision tree of N nodes
4	Loop through Each Attributes of A to ComputeGainValue(A
5	N(test) = Best_attribute_Gain
6	If N(test) is continuous
7	For Each CFT' in the splitting of CFT
8	(a). If CFT is Empty Then Child_N_{Node} is a leaf Node Else: Child_N_{Node}=Form_Decision_Tree(DT')
9	Compute_Error_N_{Node} return N_{Node}

This is the final step used to measure model's performance in terms of accuracy, root mean square error (RMSE), standard deviation, mean score, sensitivity values, ROC and AUC to adequately measure the performance of the proposed system for comparison.

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN} \qquad 3$$

Where TN represents true negative, FP is false positive, TP is true positive and FN is false negative cases.

Sensitivity is the ratio of the number of correctly classified positive classes of data computed using a function in Python as given in equation 3 as:

$$\text{Sensitivity} = \frac{TP}{TP+FN} \qquad 4$$

J. *Standard Deviation*

Standard deviation is the measure of dispersion for set of numerical values. It basically computes the square root for the spread of 'x' data distribution from the average point. It is a computation showing how far data from the average or mean point.

$$\text{Standard deviation (S.D)} =\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{X})^2} \qquad 5$$

Where N is the total dataset, x_i individual observations in the sample dataset and $\bar{X}$ is the sample mean [32]. The standard deviation was computed using the sqrt-function from math module of Python standard library with the stdev-function that takes data from a population and returns its standard deviation.

IV. Results and Discusions

The results of classification and regression analysis are presented and discussed in detail using suitable ML visualization tools. The design and implementation were done with some varying fine-tuned hyper-parameter values to provide better insight about library data. A dataset was generated with some recursive distribution of noisy data using piecewise function [f(x)] given as:

$$f(x) = e^{-x^2} + 2.5e^{-(x-2)^2} \qquad 2$$

Where f(x) is the piecewise function and x is the training and testing samples. A function is created to distribute noise within the n_samples across the interval -4 to 4 in visualizing training and testing data behavior. The predictions and classification accuracy of both models are visualized, presented and discussed using cross validation curve, neighborhood graph, standard deviation and mean scores reported as given bellow:

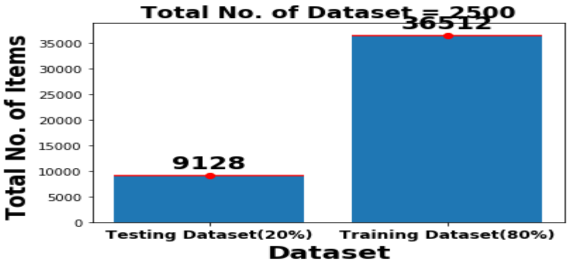


Figure 2. The Total Number of Datasets

Figure 2 depicts the total number of training and testing dataset for the proposed model. The dataset was divided into 80% training and 20% testing and the model trained using training dataset and the performance evaluated with the test dataset.

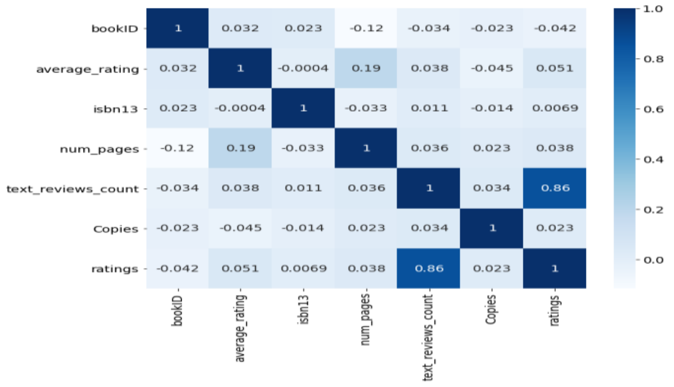


Figure 3. The Correlation Matrix

Figure 3 is the correlation matrix used to measure the relationship between database attributes(variables). The matrix depicts a linear correlation between all possible pairs of Book_ID, Average ratings, ISBN number, text reviews count and rating values. There is a positive and negative correlation between attributes in the database as shown in the main diagonal and other pairs as recorded above and below the main diagonal. The positive correlation values indicate that; the independent and dependent variables move in opposite direction while negative correlation; shows that, both variables are moving in same direction.

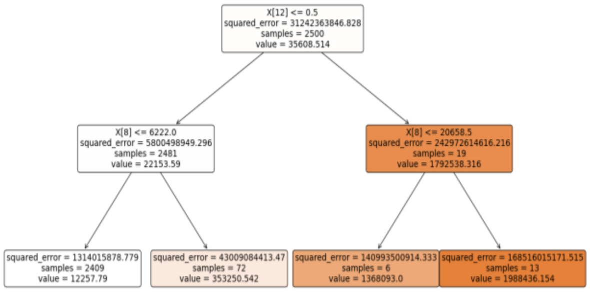


Figure 4. Decision Tree Structure for Data Visualization.

The above decision tree is generated using the If...Then...else rules given as: If (Value<=Node): Attach_to_Left else:Attach_to_Right Endif.

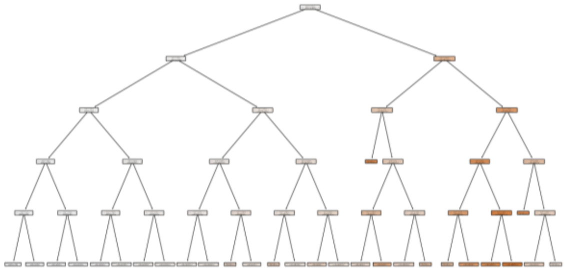


Figure 5. Tree Structure of Library Book Details

Figure 5 is a single DT of higher depth constructed and grown from Figure 4 using the decision rule. The splitting is done based on the value and clearly defined the position of inserting sub nodes from the root. Figure 5 is the tree structure generated from the system dataset. The model randomly selects patterns from original set in growing the tree and variables to represent random subset at each step. This generated a tree of height five (5).

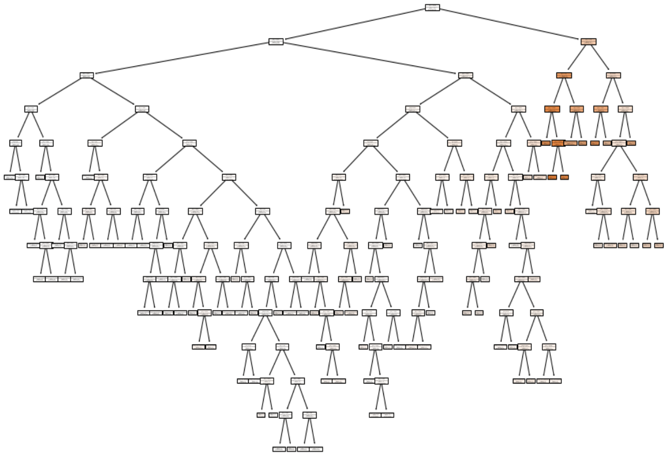


Figure 6. Forest Structure of Library Book Details

Figure 6 is the random forest generated from the proposed training dataset which combined the simplicity of different decision-trees through voting that resulted in high accuracy using library data. This was done by choosing samples randomly from the original set with replacement to grow trees and variables which represents random subset at each step. This resulted in a wide variety of forest trees.

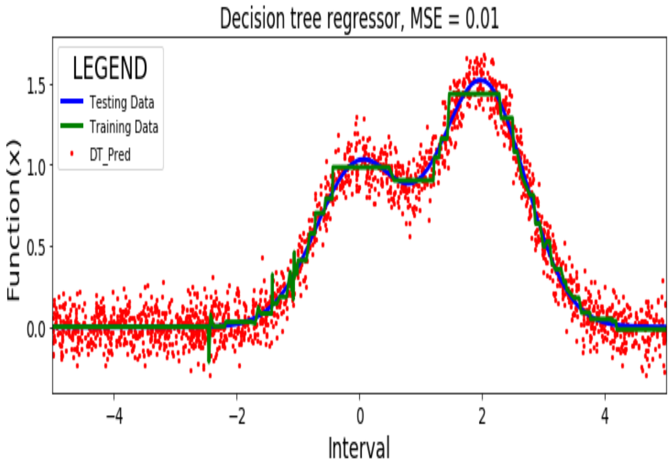


Figure 7. Training, Testing and Validation Set of Data Behavior

Figure 7 is the learning curve of DT model. The tree learns too fine details of the training data and noise = 0.2 injected through a smoothening piecewise function but overfitting occurred as shown in the predicted obtained from testing set. The results of DT as shown in Figure 7 could not generalize well on testing dataset but worked perfectly well with training set of 0.01 mean square error values within the intervals of -4 to +4.

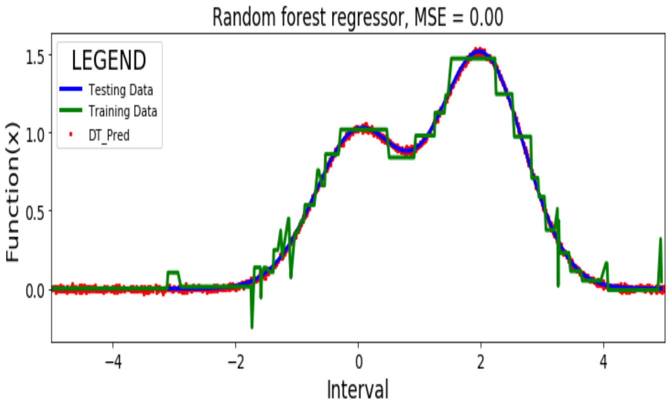


Figure 8. The Functional Graph of RF

Figure 8 is the RF functional graph showing the behavior of testing, training and validations set and no over-fitting was observed even with the presence of noisy data. The training data pattern fluctuates at certain points and matches target at other points caused.

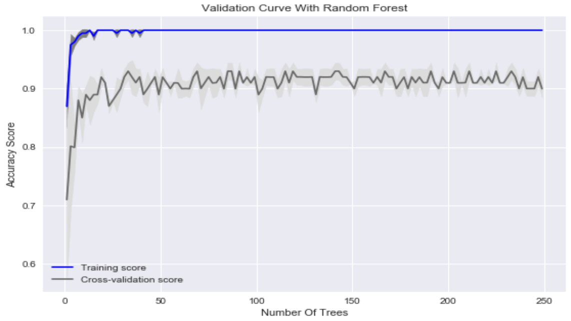


Figure 9. The Learning Curve of RF

Figure 9 is the visualized performance of RF model over range of fine-tuned hyper-parameter values (number of trees). The cross-validation curve increased and moved steadily to the right from 0.9 at the x-axis which is lower than the training score at every point.

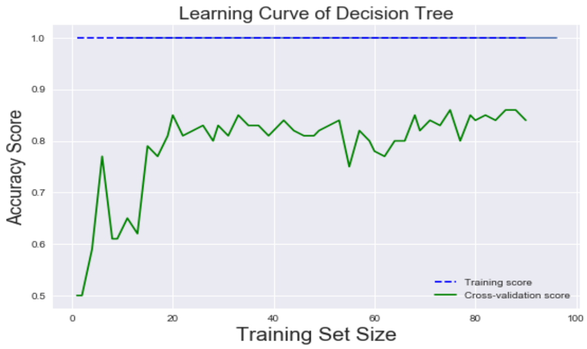


Figure 10. The Learning Curve of DT

Figure 10 depicts the learning curve of DT validation and training set. The cross-validation score increased at the initial state, degreased and increased at the final state in a fluctuating pattern and training score measured to be constant as point 1.0 at the x-axis. The training score is higher than the validation score that lies constant at 1.0 point at the x-axis along different training set with increase in size.

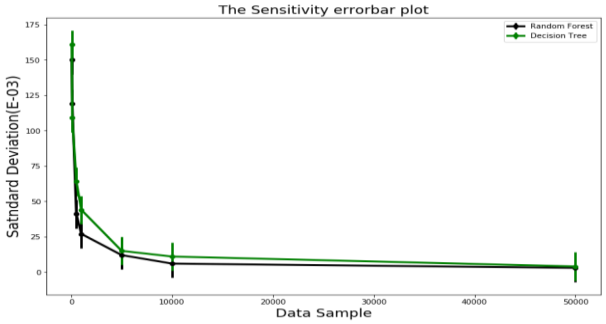


Figure 11. The Sensitivity Graph of Data over Sample Size

Figure 11 depicts the sensitivity analysis of DT and RF techniques used as a metrics to visualize the variation and relationship that exist between model performances and data sample.

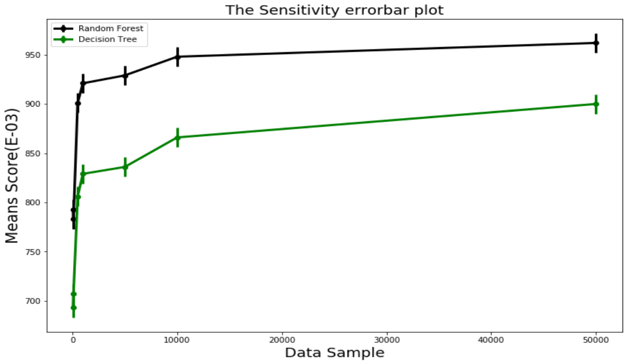


Figure 12. Mean Sensitivity Graph of Data over Sample Size

Figure 12 is the sensitivity plot used as a graphic representation of DT and RF model's performances in plotting the mean score values against different data sample.

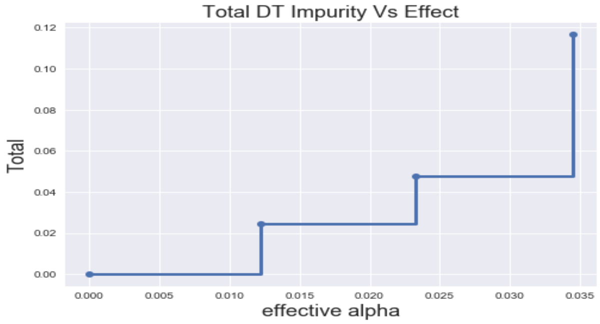


Figure 13. Total DT Impurity Vs Effect

Figure 13 is the DT graph of total impurity against effective alpha for the training dataset to reduce over-fitting using the minimal recursive cost complexity pruning. The effective values of alpha variable recorded low variance and links with the least effective alpha are pruned first. The values increased along with respect to the training data in a step wise manner.

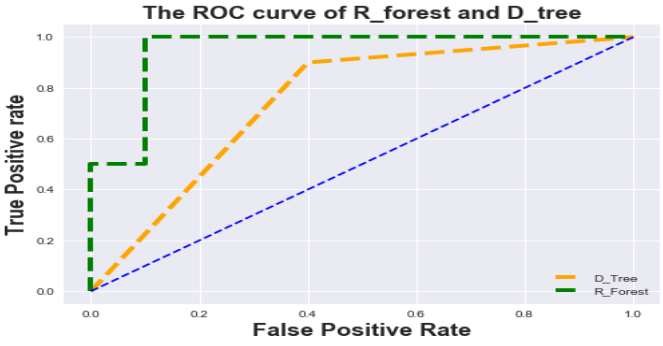


Figure 14. The ROC Curve of RF and DT

Figure 14 is the receiver operating characteristic (ROC) graph of RF and DT showing the trade-off between sensitivity or true positive rate and specificity(1-FPR). The RF ROC curve is wider and closer to top-left corner of the graph which performed better than the DT. The proposed system revealed the behavior of points lying along the diagonal (True Positive Rate=False Positive Rate) as expected with high accuracy rate.

TABLE II. COMPARING DT, RF AND STACKED TECHNIQUES

MODEL	ACCURACY (%)	RMSE	Sensitivity
DT	70.0	0.5477	0.8333
RF	85.0	0.3873	1.0000

Table II shows the performance of RF, DT and Stacked model in terms of accuracy and RMSE. That stack model recorded 95.0% accuracy level as the highest with 0.0987 RMSE, RF technique produced 85% and 0.3873 error rate compared to 70.0% and 0.5477 error rate produced by DT. This shows that the stacked model performed better in terms of accurate rate, followed by the RF and DT producing the least performance rate.

V. CONCLUSION

The proposed ML visualization techniques will help user understand database information much easier when presented in a visual form compared to when using the existing system tools. The selected ML models used in visualization makes it so appealing and artistic to interpret data and its composition for the purpose of better decision-making task. There are different and several visualization techniques adopted but some of these techniques may lead to wrong and poor data visualization. This is important in choosing the appropriate visualization technique to better understand and interpret the data for future use. The stacked model performed best followed by RF and the visualization results of RF technique outperformed the DT model in term of accuracy, error rate, behavior of training data containing noise and the model's learning rate with testing dataset.

REFERENCES

[1] K. Li, A. Tiwari, J. Alcock, & P. Bermell-Garcia, " Categorisation of Visualization Methods to Support the Design of Human-Computer Interaction Systems," Applied Ergonomics, Vol.55, 2016, pp.85-107.

[2] C. C. Gramazio, K. B. Schloss, & D. H. Laidlaw, "The Relation Between Visualization Size, Grouping, and User Performance," IEEE Transactions on Visualization and Computer Graphics, vol.20, no.12, 2014, pp.1953-1962.

[3] C. Plaisant, "The Challenge of Information Visualization Evaluation ║, Proceedings of Conference on Advanced Visual Interfaces," ACM, New York, 2004, pp.109-116.

[4] H. W. De-Regt, "Visualization as a Tool for Understanding," Perspectives on Science, vol.22, no.3, 2014, pp.377-396.

[5] M. Khan, & S. S. Khan, "Data and Information Visualization Methods, and Interactive Mechanisms: A Survey," International Journal of Computer Application ║ (IJCA), vol.34, no.1, 2011, pp.45.

[6] M. Tory, & T. Moller, "Human Factors In Visualization Research, IEEE Transactions On Visualization And Computer Graphics ║," vol.10, no.1, 2004, pp.1-14.

[7] A. Adediran, & S. S. Ajibade, "An Overview of Big Data Visualization Techniques in Data Mining," International Journal of Computer Science and Information Technology, vol.4, no.3, 2016, pp.105-113.

[8] M. Teras, & S. Raghunathan, "Big Data Visualisation in Immersive Virtual Reality Environments: Embodied Phenomenological Perspectives to Interaction," ICTACT Journal on Soft Computing, vol.5, no.4, 2015, pp.1009-1015.

[9] F. Nazeer, N. Nazeer, & I. Akbar, "Data Visualization Techniques-A Survey, International Journal for Research in Emerging Science and Technology(IJREST)," vol.4, no.3, 2017, pp.4-8.

[10] J. Moore, "Data Visualization in Support of Executive Decision Making," Interdisciplinary Journal of Information, Knowledge, and Management, vol.12, 2017, pp.1-14.

[11] M. Narayanan, & S. K. Shanker, "Data Visualization Methods as the Facilitator for Business Intelligence," International Journal of Engineering and Advanced Technology(IJEAT), vol.8, no.6, 2019, pp.3925-3928.

[12] T. Plank, & M. Helfert, "Interactive Visualization and Big Data-A Management Perspective," Proceedings of the 12th International Conference on Web Information Systems and Technologies (WEBIST), vol.2, 2016, pp.42-47.

[13] Z. Zhang, "A New Authoring System for Diverse Data Visualization at Scale," Massachusetts Institute of Technology, 2021, pp.1-71.

[14] A. L. Ogier, & M. L. Stamper, "Embedded Visualization of Services in the Library Research Lifecycle," Journal of eScience Librarianship, vol.7, no.1, 2018, pp.1-10.

[15] G. Chawla, S. Bamal, & R. Khatana, "Big Data Analytics for Data Visualization: Review of Techniques," International Journal of Computer Applications, vol.1&2, no.21, 2018, pp.37-40.

[16] M. M. Butavicius, & M. D. Lee, "An Empirical Evaluation of Four Data Visualization Techniques for Displaying Short News Text Similarities," International Journal of Human-Computer Studies, vol.65, no.11, 2007, pp.931-944.

[17] E. Y. Gorodov, & V. V. Gubarev, "Analytical Review of Data Visualization Methods in Application to Big Data," International Journal of Electrical and Computer Engineering(IJECE), vol. 4, 2013, pp.1-7.

[18] L. Tay, N. Vincent, A. Malik, J. Zhang, J. Chae, D. S. Ebert, Y. Ding, J. Zhao, & M. Kern, "Big Data Visualizations in Organizational Science," SAGE Journal of Organizational Research Methods, vol.4, no.5, 2017, pp.1-20.

[19] J. Merino, I. Caballero, B. Rivas, M. Serrano, & M. Piattini, M. "A Data Quality in Use Model for Big data," Future Generation Computer Systems, vol.63, 2016, pp.123-130.

[20] F. Wang, "Computer Graphics Algorithm Based on Visualization Teaching Theory," Journal of Digital Information Management, vol.13, no.3, 2015, pp.137.

[21] J. M. Teets, D. P. Tegarden, & R. S. Russell, "Using Cognitive Fit Theory to Evaluate the Effectiveness of Information Visualizations: An Example using Quality Assurance Data," IEEE Transactions on Visualization and Computer Graphics, vol.16, no.5, 2010, pp.841-853.

[22] T. Nasser, & R. S. Tariq, "Big Data Challenges," Journal of Computer Engineering & Information Technology, vol.4, no.3, 2015, pp.2-10.

[23] F. Provost, & T. Fawcett, "Data Science and Its Relationship to Big Data and Data-Driven Decision Making," Big Data, vol.1, no.1, 2013, pp.51-59.

[24] H. Tickle, M. Speekenbrink, K. Tsetsos, E. Michael, & C. Summerfield, "Near-Optimal Integration of Magnitude in the Human Parietal Cortex," Journal of Cognitive Neuroscience, vol.28, no.4, 2016, pp.589-603.

[25] L. Kuncheva, & C. Whitaker, "Measures of Diversity in Classifier Ensembles and their Relationship with the Ensemble Accuracy," Machine learning, vol.51, no.2, 2003, pp.181–207.

[26] A. O. Balogun, M. A. Mabayoje, S. Salihu, & S. A. Arinze, "Enhanced Classification Via Clustering techniques using Decision Tree for Feature Selection," International Journal of Applied Information System(IJAIS), vol.9, no.6, 2015, pp.11-16.

[27] N. Bhargava, G. Sharma, R. Bhargava, & M. Mathuria, "Decision Tree Analysis on j48 Algorithm for Data Mining," Proceedings of International Journal of Advanced Research in Computer Science and Software Engineering(IJARCSSE), vol.2, no.6, 2013, pp.45-98.

[28] M. Soranamageswari, & C. Meena, " Histogram based Image Spam Detection using Backpropagation Neural Networks," Global Journal of Computer Science and Technology(GJCST), vol.9, no.5,2020, pp. 62-67.

[29] H. P. Patel, & P. Prajapati, "Study and Analysis Tree-based Classification Algorithms," International Journal of Computer Sciences and Engineering(IJCSE), vol.6, no.10, 2018, pp.74-78.

[30] T. M. Lakshmi, A. Martin, R. M. Begum, & V. P. Venkatesan, "An analysis on the performance of decision tree algorithms using students qualitative data," International Journal of Modern Education and Computer Science(IJMECS), vol.5, no.3, 2013, pp.18-27, DOI: 10.5815/ijmecs.2013.05.03.

[31] Y. Li, H. Xing, Q. Hua, & X. Wang, "Classification of BGP anomalies using decision trees and fuzzy rough sets," In Systems, Man and Cybernetics. IEEE International Conference on (SMC), 2014, pp.1312–1317, DOI: 10.1109/SMC.2014.6974096.

[32] N. Ammu, & M. Irfanuddin, "Big Data challenges," International Journal of Advanced Trends in Computer Science and Engineering, vol.2, no.1, 2013, pp.613-615.

AUTHOR PROFILE

Mr. Stanley Ziweritin is working as a lecturer in the Department of Estate Management and Valuation at the School of Environmental Design and Technology, Akanu Ibiam Federal Polytechnic, Unwana-Afikpo, Ebonyi State. He holds HND (Computer Science) from Rivers (Now Kenule Benson Saro-Wiwa) State Polytechnic, Bori. PGD (Computer Science) and M.Sc (Computer Science) from the University of Port Harcourt (UPH) respectively. His research interest revolves around: Artificial Intelligence (AI), Deep Machine Learning (DML), Natural Language Processing (NLP), Data Mining, Data Visualization, Algorithms, Database System Design, and Programming. He has published in several international journals. He is a registered member of the Computer Professionals (Registration Council) of Nigeria (CPN).