ISSN : 2393-8188 (print)
2393-8196 (online)
www.milliyasrcollege.org.journal.php

Implementation on Uncertain Numerical Dataset Using XBOOST Bagging and IndRNN Boosting Techniques

Santosh S Lomte[1], Sanket Gunderao Torambekar[2]

[1]*Radhai Mahavidyalay., Aurangabad, Maharashtra, India*

Abstract

In real-world applications, confusing data is common due to factors including estimate mistakes, network delay, outmoded sources, and sampling flaws. This kind of uncertainty must be handled carefully otherwise the mining results may be erroneous. For the classification and prediction technique, this research work employed Bagging and Boosting methods for Decision Trees. A decision tree may describe sequential choice issues with ambiguity and classifiers manage ambiguous data. Bagging is a well-known ensemble method that builds data bags with the same class mark and probability as the initial data collection. To deal with data ambiguity, the proposed work prefers PDF distribution. The distribution of the sample depends on the difficulty of identifying each individual. Aside from abstracting unknown data via statistical derivatives, the decision tree works effectively when it utilizes the PDF distribution. This research work delivers the dataset to the Bagging and Boosting algorithm instead of straight to the decision tree, and then the created dataset is supplied to the decision tree and then the classification is done.

Keywords: Bagging, Boosting, Decision Tree, XGBoost, IndRNN

1. Introduction

Data Mining (DM) is a really developing area every day now. It is to extract information that is previously unknown from a massive data warehouse that is secret, significant, & useful. In meteorology & atmospheric sciences, as well as many other areas, DM is also used. The withdrawal of secret predictive knowledge from huge datasets is DM. It helps us to locate the pine needle that is concealed in this data haystacks. [1, 2] The prevailing brand-new equipment has tremendous potential to support businesses in their data warehouses that concentrate on the most relevant details. DM applications forecast future patterns & habits, enabling organizations to be proactive [3].

The development of massive databases has resulted in improvement in numerical data collection & storage. From the mundane to the more exotic, this has happened in all areas of human life. No wonder, then, that the prospect of tapping these records has increased in significance, obtaining through them info that could be of use to the database owner. DM has become well known as the sector involved in this activity. Figure 1 shows the context of DM [4].

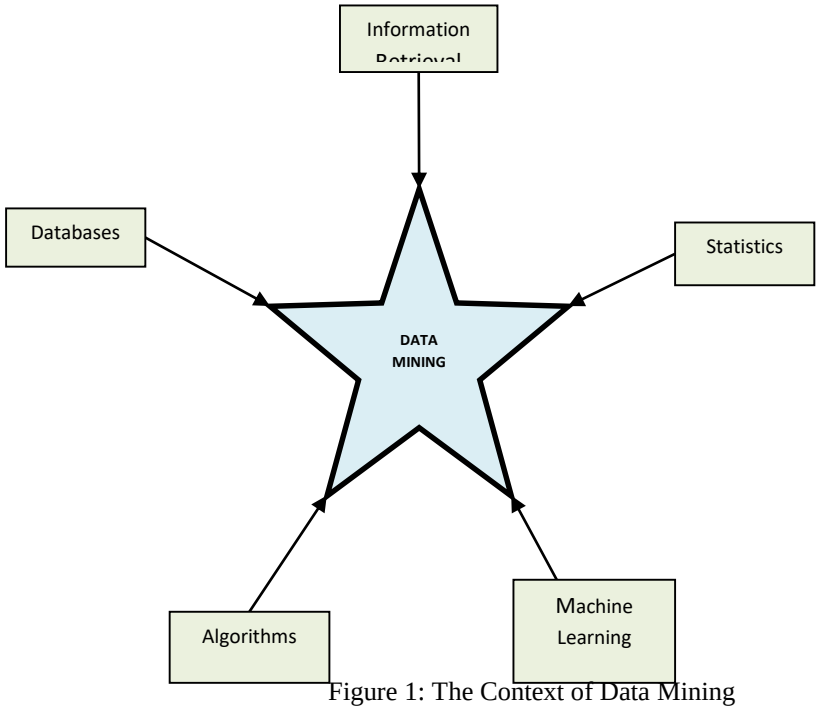


Figure 1: The Context of Data Mining

Several improved technologies have been created in recent years to store & archive vast amounts of data continuously. Data that, in some situations, may include faults or might only be partially complete. Sensor networks, for ex., usually generate large numbers of unknown sets of data. Data points that apply to objects that are only vaguely defined in their representation & are thus deemed unknown in other instances. These have made a requirement regarding unclear algo's & frameworks for data processing. In the management of uncertain data, the records of data are typically characterized through distributions of probability instead of destined values [5].

1.1. Decision Tree

A decision tree contains nodes that generate a rooted tree, which means that a "root" node it is a directed tree with no incoming edges. There is only one incoming edge for the additional nodes. The decision tree is a classifier of the

example space known as a resource partition [6].

Decision tree, corresponding to a particular distinguishing feature of the i/p attribute values, each inner node split up the instance space into 2 or additional subspaces. Every test believes a single attribute in the simplest & most common case, in order to partition the instance space corresponding to the attribute's value. This condition corresponds to a spectrum in the case of numeric properties. One class representing an extremely important target value is allocated to every leaf. Alternatively, the leaf is able to keep the probability vector indicating the chance of achieving some value for the target attribute. Cases are categorized by directing them by tree's root down to a leaf based on the result of tests with the way [7] as illustrated in figure 2.

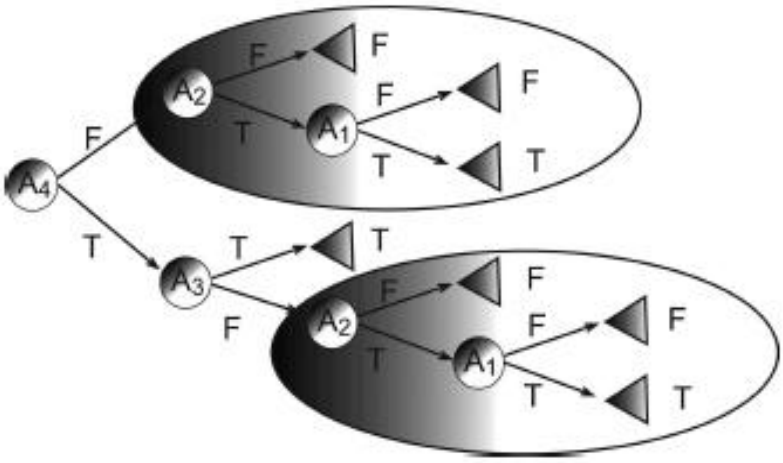


Figure 2:Illustration of Decision Tree with Replication

1.2. Boosting

It is a technique that, by increasing its precision, creates optimal use of the classifier. The classifier technique is used as a subordinate in the training set to make an extremely correct classifier. Different classifications are applied repeatedly on the training data, the single classifiers obtain & unite them in the training set into a final, highly accurate classifier When the process has been completed. The ultimate classifier achieves high precision in the test range. Boosting algorithms have many variants, most used is AdaBoost, the AdaBoost can only be used for the issues having the binary classification. Boosting is a widespread method for the betterment of the efficiency of learning also. by uniting "weak hypotheses", a technique for searching for an extremely precise classifier on the training set, all of which need only average accuracy on the training set, previously have seen an overview of numerous ways

to unite neural networking [8] as shown in figure 3.

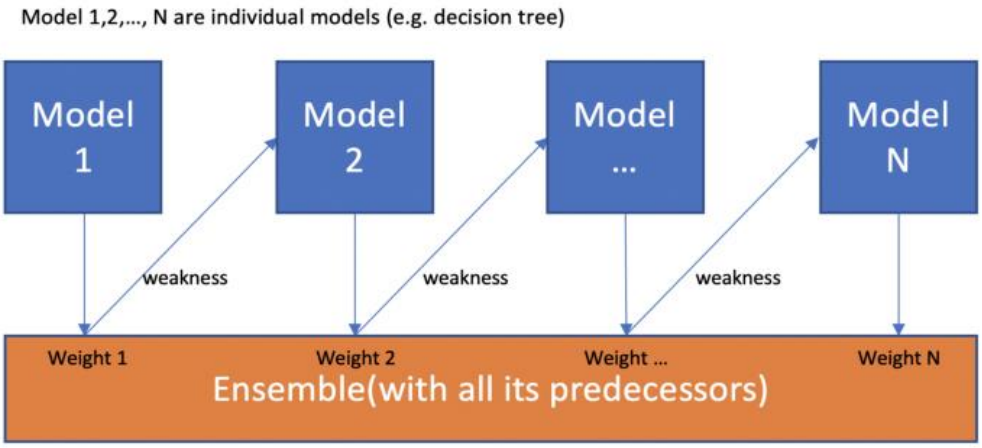


Figure 3: Boosting Algorithm

1.3. Bagging

A combination of aggregating & bootstrapping is known as Bagging. If the data distribution parameter for bagging is quite accurate then a similar method can be used to achieve the goal, after combining them, with the classifier of better quality. Considering the training dataset (Tn), B samples of bootstrap (Tb) are generated, where b=1; 2;::::::::B. these bootstrap samples are generated based on the replacement of the numbers in common considering the original dataset as n.

Bagging, also recognized as Bootstrap aggregating, is a process that repetitively tests from a data set corresponding to a uniform distribution of probability as shown in figure 4. Bagging increases the generalization. Error, i.e., bagging improves to decrease errors associated with random training data variability. It eliminates misclassified elements in simple statements [9].

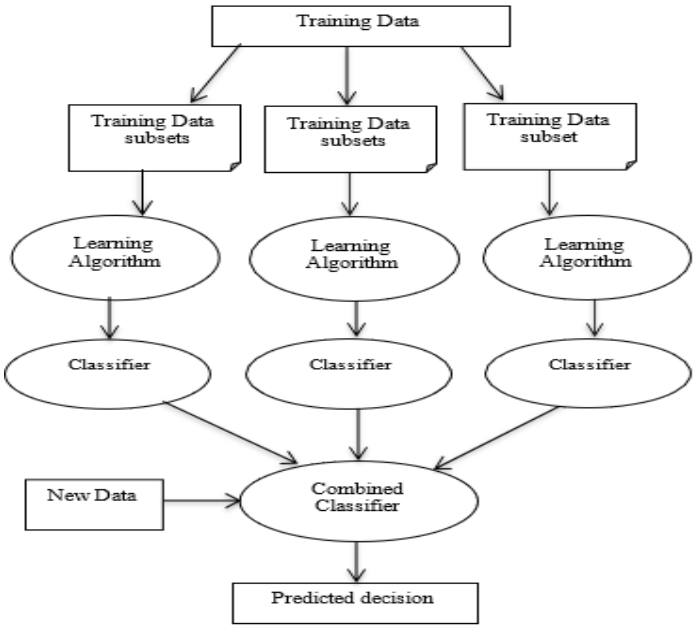


Figure 4: Bagging Procedure

2. Literature of Review

Various research has been done by the researchers. Different models have been suggested by the researchers. Various papers and models have been studied for this paper which is included in this literature review.

Xie et al. [10] suggested an unknown Kernel SVM (UKSVM) algo. which obtained values set by the univariate tasd interval attribute on the fundamental probability density function that samples interval. SVM with additives for kernels, i.e. To classify samples, HIK, χ2, & Hellinger's kernels are then utilized.

In 2017, Dou et al. [11] implemented an active learning query-by-committee system where no labeled data is needed. So, the initial decision tree is generated with no use of any already defined data labels, & then the decision tree is incrementally updated as newly labeled instances are added.

This approach is capable of exploiting uncertainty & analyzing pool's informativeness into various categories. But categorization efficiency fluctuates dependent on the seed & pool sets. Destercke & Wang [12] developed a DT learning technique in 2016 that is able to minimize epistemic complexity through querying & handling very useful ambiguous scenarios. The evidential probability for querying unknown training events is extracted from entropy intervals. Those data whose values are exactly defined are managed by classical decision trees.

In 2015, Reitmaier et al. [13] suggested a transductive active learning approach where in each active learning period a named data pool was created. This approach reveals a probabilistic generative model which, when several labels are available during the active learning process, can be improved & iteratively refined.

For the evaluation of the work, G. Michael et al. [14] considered such meta-classifiers such as AdaBoost, Attribute Selected Classifier, Regression Classification, Bagging Filtered Classifier, Logit Boost, Multiclass Classifier. The author has considered some techniques of supervised ML & has also considered meta-classifiers on the dataset being studied for the identification of the possibility of network attacks. To predict classifiers precision used, the 10-fold cross-validation technique is used, & outcomes generated are then compared to determine the accuracy of the technique. The author claimed, on the basis of the assessed results, that the Bagging meta-classifier outperformed the simple meta-classifier. In addition, metrics such as TP, FP Rates, F-measure & ROC Area are some of the variables that are higher in the case of regular class professionals & in the case of the possibility of attacks by the administrator.

Centered on the assessment of context, diversity & density in this technique, the paradigm of ε-insensitive support vector regression (SVR) has been established. Hajmohammadi et al. [15] suggested a model that combines a semi-supervised & uncertainty-based active learning approach in 2015. To prevent outlier choice in the active learning technique, the density of the chosen examples is considered in this approach. In 2015, Yang et al. [16] published a semi-supervised batch mode multi-class active learning algo. to assess the uncertainty of the active pool's visual concept identification results.

Satya Ranjan Dash et al., [17] in their work considered the different methodologies for the classification of the datasets like Bayesian classification, Decision tree induction-based classification of the datasets & also considered some of the classification techniques using the concept of fuzzy logic. The author has considered different factors for the purpose of the comparison like ROC area, RSME, Kappa statistics, MAE, time consumed in the development of the model, etc. Based on the results & analysis done in the work. The author indicated in the conclusive part that the FuzzyRoughNN & ID3 have resulted as the best techniques for the purpose of prediction & also for classification, in the evaluation work the author has used a post-operative dataset of the patient. & in the case when the ID3 & Fuzzy Rough NN are compared over the same dataset then the ID3 has resulted as

the best from the discussed two techniques.

Pfahringer et al. (2012) [18], described the concept of the meta-feature generation method where the author has considered meta-learning process, in the work meta-learning, actually is the process where the performance of every of the base learners are compared in the manner of one-to-one. Using the suggested meta-learner technique, the author described enhanced meta-learner which is named as ART forest & sometimes also been termed as the Approximate ranking tree forest, the results obtained from the ART methodology can be compared with the traditional meta-learning technique.

In their work, Aman K. Sharma et al. [19] considered the assessment of the techniques used by the WEKA tool for the following four techniques: ID3, J48, Simple CART & Alternating Decision Tree, for which the author used the email spam dataset & then compared techniques considered on the basis of the accuracy of the classification. The J48 appears higher in terms of classification accuracy relative to other strategies such as ID3, CART & AD Tree on the basis of the evaluation performance.

The suggested extensions of the imbalanced data ensembles may be classified in a different way. The taxonomy suggested by Galar et al. [20] showed the difference between the different cost-sensitive approaches & the different integrated approaches for pre-processing. The cost optimization factor is considered in the first group of techniques by considering ensemble methods for boosting, such as AdaBoost, AdaqC, or Rare Boost, etc.

3. Problem Formulation

Bagging is a voting scheme in which includes n templates are constructed, typically of the same kind. Each model's predictions are reported for an unknown instance. The class that has the highest vote among the models' predictions is allocated. For bagging incorporating the decision tree as a base classifier, A decision tree, working in a divide-&-conquer manner, is a set of tree-structured decision tests. Every non-leaf node is connected to a feature test, also called a break; according to their different feature test values, data falling into nodes will be divided into different subsets. A mark that will be allocated to instances falling within this node is associated with each one leaf node.

In the same way, the ANN is being incorporated as a base classifier for boosting technique, Neurons are linked from weighted connections to the network of formal. Several possible network architectures exist, the most common of which is a multi-layer feed-forward network. The neurons are linked layer-by-layer here, & there are neither in-layer

links nor cross-layer links. There is an i/p layer that receives vectors of input features, where every neuron normally corresponds to 1 part of the vector of the function. The classifiers result in some prediction form & the predictions generated from two separate classifiers are then combined & the output is generated based on MDT (Meta- Decision Tree).

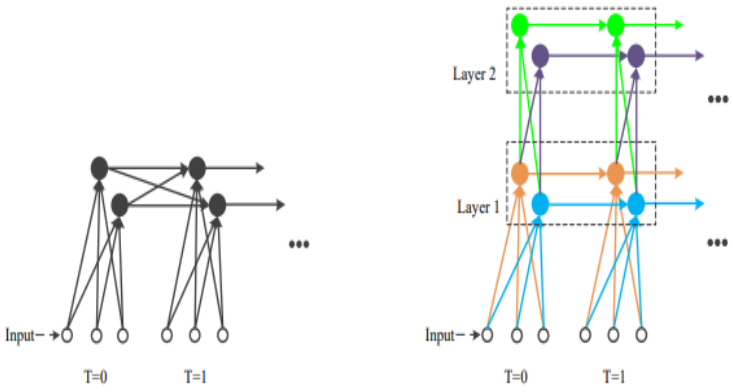
4. Research Methodology

In this research methodology techniques used are Independent Recurrent Neural Network (IndRNN), XGBoost, Bagging Boosting, and Decision tree.

4.1. Independent Recurrent Neural Network (IndRNN)

RNN can effectively preserve knowledge about the past depending on the number of time levels. By supplying words or characters as input parameters, RNN has remained widely helpful in real language management, speech identification, and system interpretation. Figure 5 shows the IndRNN Model.

Because of gradient vanishing and detonating difficulties, RNNs are notoriously challenging to work out, making it difficult to learn long-term patterns and build deep networks. To solve these issues, a new type of RNN called an IndRNN was developed with the recurrent connection described as a Hadamard product, where the neurons in the similar layer are independent of each other and linked across layers. IndRNN with regulated recurrent weights efficiently addresses gradient exploding and vanishing difficulties due to better-behaved gradient backpropagation, allowing long-term dependencies to be learned. Furthermore, an IndRNN may be trained successfully using non-saturated activation functions for example rectified linear unit (ReLU) [21].



(a) Connecting all neurons using RNN **(b) Connecting using IndRNN**

Figure 5: Illustration of simple (a) RNN and (b) IndRNN unfolded in time

The IndRNN neural network is defined in equation 1:

$$h_t = \sigma(Wx_t + u \odot h_{t-1} + b) \quad (1)$$

In this equation σ is the element-wise activation function of the neurons in RNN layer. In opposition to classic RNN, which uses a matrix as the recurrent weight and analyzes the recurrent input using matrix product, IndRNN uses a vector as the recurrent weight and analyzes the recurring input utilizing component vector product. Every neuron in a single layer is autonomous from the others, which is referred to as "independently recurrent". For the nth neuron, the hidden state $h_{m,t}$ could be acquired as h as shown in equation 2.

$$h_{m,t} = \sigma(W_m X_t + u_m h_{m,t-1} + b_m) \quad (2)$$

where W_m, u_m and b_m are the mth row of the recurrent weight, input weight and bias, correspondingly. The IndRNN may be expanded to a convolutional IndRNN by doing the convolutional operations of every time step instead of processing the input using the fully connected weight Wx_t. (W $*$ x_t, where $*$ signifies the convolution operator).

The IndRNN model as it evolved over time is seen in Figure 5 (b). Each solid dot symbolizes a neuron in a layer, whereas every line indicates a connection. The processing of each neuron in one layer can be seen to be independent of one another, as indicated by the different colors. At each time step, each neuron only obtains data by its own hidden state and the input. Every neuron in the IndRNN is responsible for one kind of spatial-temporal pattern. On the other hand, the recurrent weight connection, connects neurons in the simple RNN over time, as shown in Figure 5 (a). Figure 5 (b) further, by feeding the output of all neurons in the previous layer to neurons in the following layer, one extra IndRNN layer may be used to explore the correlations (cross channel data) between neurons in a single layer [21].

4.2. XGBoost

Chen et al. introduced XGBoost, a scalable tree boosting method [22]. It was heavily used in Kaggle's Higgs sub-signal identification competition. It has lately gained widespread notice owing to its exceptional efficiency and great forecast accuracy. Indeed, XGBoost is an upgraded GBDT method [23], which is generally utilized in the area of classification and regression and consists of several decision trees. However, XGBoost is not identical to GBDT in certain ways. To begin, GBDT uses just the first-order Taylor expansion, while

XGBoost augments the loss function with a second-order Taylor enlargement. Second, the goal function is normalized to prevent overfitting and to reduce the model's complexity. The construction of the XGBoost is shown in Figure 6.

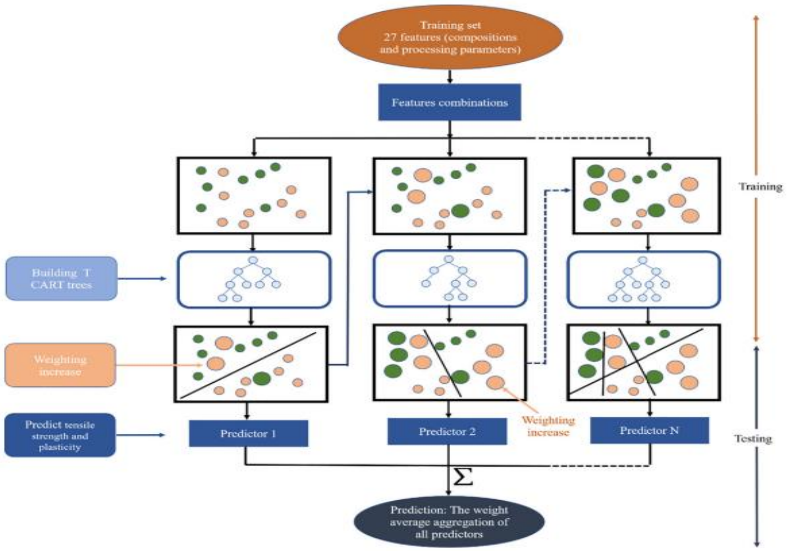


Figure 6: Structure of the XGBoost

4.3. Proposed Methodology

This methodology is totally based on Bagging, boosting, and decision tree. As two separate classifiers are used in this approach, where the decision tree is the working classifier and the XGBoost and IndRNN are used for the classifier's training at the same time, the fundamental principle of the work is to choose the best classifier from the available category. Decision Tree techniques are utilized successfully to improve the consistency of classification & prediction systems & can also support medical professionals in their decision-making processes as shown in figure 7.

Step 1: Input dataset is preprocessed for training and testing process.

Training process

Step 2: In the training phase, a dataset is processed for classification using the Decision Tree approach, and the classified data is retrieved.

Testing Process

Step 3: In the testing phase, the dataset is passed through PDF which defines the probability function and represent the density of the dataset.

Step 4: Bagging with XGBoost and boosting with IndRNN classifiers are used to preprocess the dataset.

Step 5: The retrieved dataset is processed for classification using the Decision Tree approach, and the classified data is retrieved.

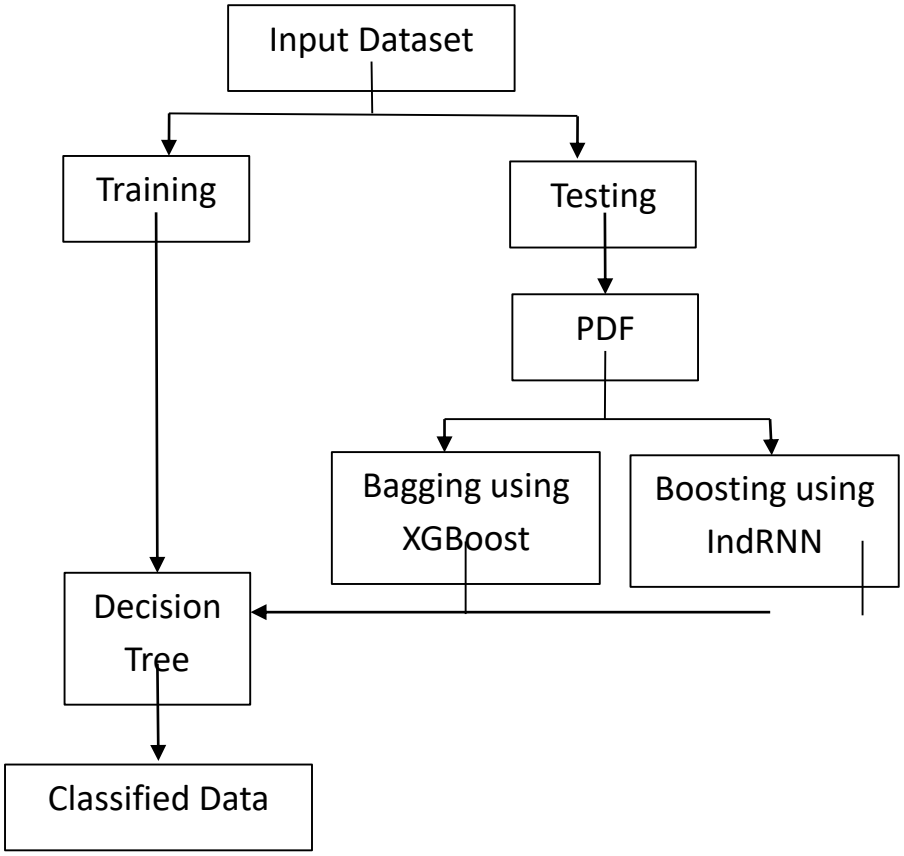


Figure 7: Proposed Methodology

5. Results

The overall result of the work considers some characteristics that determine the dataset's categorization and the process's usefulness. The condition is considered as a data point in the assessment portion, and restrictions are only evaluated for one instance in data gathering. Considered are mean absolute error, mean squared error and root mean squared error of the root. And confusion matrix describes the dataset categorization and is being created for job evaluation.

Despite the ambiguity of the words, the ambiguity matrix itself is obvious. The dataset contains 400 unique samples with 5 fields: User ID, Gender, Age, Approximate Salary, and

Purchased. The work is verified as bagging and boosting in two phases, with bagging for XGBoost classification and boosting for IndRNN classification using unknown datasets.

Result 1: In figure 8 Bagging using XGBoost classification is processed to retrieve the data. Which defines the mean absolute error: 0.05, mean squared error: 0.05, root mean squared error: 0.223606 and confusion metrics $\begin{matrix} 65 & 3 \\ 2 & 30 \end{matrix}$

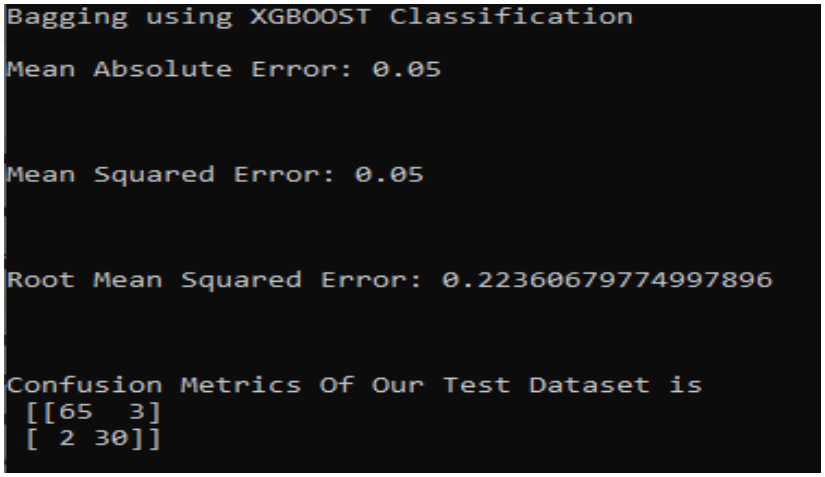


Figure 8: XGBoost classification of training dataset using Bagging.

Result 2: In figure 9 Boosting using IndRNN classification is processed to retrieve the data. Which defines the mean absolute error: 0.057, mean squared error: 0.029, root mean squared error: 0.170446 and confusion metrics $\begin{matrix} 67 & 1 \\ 2 & 30 \end{matrix}$

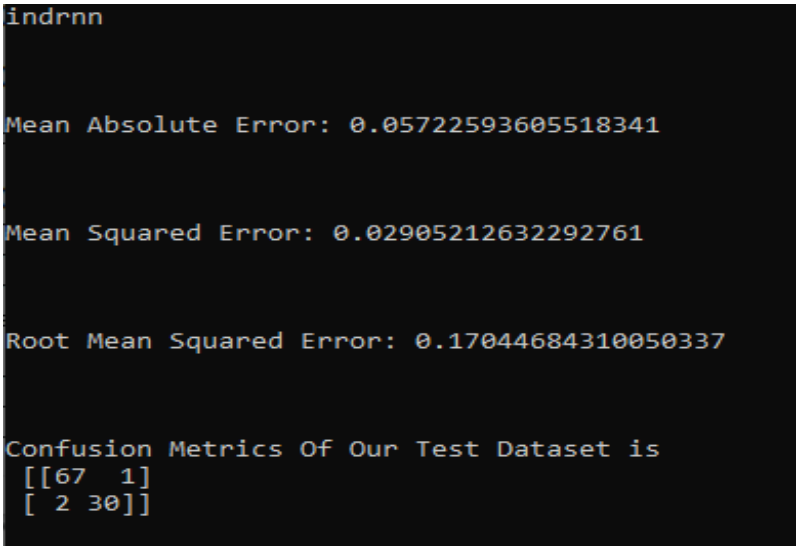


Figure 9: IndRNN classification of training dataset using Boosting

Table 1: Efficiency comparison for classification by bagging and boosting

Parameters	Existing Technique [25]		Proposed Technique	
	Bagging using Random Forest	Boosting Using ANN	Bagging using XGBoost	Boosting Using IndRNN
MAE	0.09	0.1205	0.05	0.0572
MSE	0.09	0.500	0.05	0.02905
RMSE	0.3	0.2425	0.2236	0.17044

Table 1 illustrates the comparison of dataset accuracy on the basis of different classifiers used in this proposed technique by the existing technique.

6. Conclusion

Decision trees are notable for their ability to visualize performance data and for solving numerous pattern recognition problems. The flexibility to apply alternative feature subsets and decision rules at different categorization points is a crucial aspect of DTC. The present study handles an ambiguous numerical dataset using weighted learning. Boosting and bagging are used to assemble classifiers over ambiguous numerical data. The study also covers the basics of ambiguous datasets, boosting, and bagging. In this study, two classification criteria are chosen: bagging and boosting. Following that is boosting, which uses the IndRNN algorithm and has higher accuracy than bagging classification for the ambiguous numerical dataset.

7. References

[1] Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. Introduction to data mining. Pearson Education India, 2016.

[2] Allahyari, Mehdi, Seyedamin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. "A brief survey of text mining: Classification, clustering and extraction techniques." arXiv preprint arXiv:1707.02919 (2017).

[3] Nisbet, Robert, Gary Miner, and Ken Yale. Handbook of Statistical Analysis and Data Mining Applications. Elsevier, 2017.

[4] Yoo, Illhoi, Patricia Alafaireet, Miroslav Marinov, Keila Pena-Hernandez, Rajitha Gopidi, Jia-Fu Chang, and Lei Hua. "Data mining in healthcare and biomedicine: a survey of the literature." Journal of medical systems 36, no. 4 (2012): 2431-2448.

[5] Aggarwal, Charu C., ed. Managing and mining sensor data. Springer Science & Business Media, 2013.

[6] Matuszny, Marcin. "Induction of decision trees for building knowledge bases of production processes." Polish Technical Review (2021).

[7] Zheng, Yifeng, Guohe Li, and Wenjie Zhang. "A New Algorithm for

Classification Based on Multi-classifiers Learning." In International Conference on Geo-Spatial Knowledge and Intelligence, pp. 254-262. Springer, Singapore, 2017.

[8] Schapire, Robert E. "Explaining adaboost." Empirical inference. Springer, Berlin, Heidelberg, 2013. 37-52.

[9] Mordelet, Fantine, and J-P. Vert. "A bagging SVM to learn from positive and unlabeled examples." Pattern Recognition Letters 37 (2014): 201-209.

[10] Xie, Zongxia, Yong Xu, and Qinghua Hu. "Uncertain data classification with additive kernel support vector machine." Data & Knowledge Engineering 117 (2018): 87-97

[11] Dou, Chenxiao, et al. "Active learning with density-initialized decision tree for record matching." Proceedings of the 29th International Conference on Scientific and Statistical Database Management. 2017.

[12] Ma, Liyao, Sébastien Destercke, and Yong Wang. "Online active learning of decision trees with evidential data." Pattern Recognition 52 (2016): 33-45.

[13] Reitmaier, Tobias, Adrian Calma, and Bernhard Sick. "Transductive active learning–a new semi-supervised learning approach based on iteratively refined generative models to capture structure in data." Information Sciences 293 (2015): 275-298.

[14] Michael, G., A. Kumaravel, and A. Chandrasekar. "Detection of malicious attacks by meta classification algorithms." International Journal of Advanced Networking and Applications 6.5 (2015): 2455.

[15] Hajmohammadi, Mohammad Sadegh, et al. "Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples." Information sciences 317 (2015): 67-77.

[16] Yang, Yi, et al. "multi-class active learning by uncertainty sampling with diversity maximization." International Journal of Computer Vision 113.2 (2015): 113-127.

[17] Dash, Satya Ranjan, and Satchidananda Dehuri. "Comparative study of different classification techniques for post operative patient dataset." International Journal of Innovative Research in Computer and Communication Engineering 1.5 (2013): 1101-1108.

[18] Serasiya, Shilpa Dhanjibhai, and Neeraj Chaudhary. "Simulation of various classifications results using WEKA." International Journal of Recent Technology and Engineering (IJRTE) 1.8 (2012).

[19] Sharma, Aman Kumar, and Suruchi Sahni. "A comparative study of classification algorithms for spam

email data analysis." International Journal on Computer Science and Engineering 3.5 (2011): 1890-1895

[20] Galar, Mikel, et al. "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42.4 (2011): 463-484.

[21] Li, Shuai, Wanqing Li, Chris Cook, and Yanbo Gao. "Deep independently recurrent neural network (indrnn)." arXiv preprint arXiv:1910.06251 (2019).

[22] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2016, 785–794

[23] Y. Xu, X. Zhao, Y. Chen, et al., Research on a mixed gas classification algorithm based on extreme random tree, Appl. Sci. 9 (9) (2019) 1728

[24] I. Sutskever, G.E. Hinton, A. Krizhevsky, Imagenet classification with deep convolutional neural networks, Adv. Neural Information Processing Systems (2012) 1097–1105.

[25] Lomte, Santosh S., and Sanket G. Torambekar. "Decision tree for uncertain numerical data using Bagging and Boosting". 5th World Conference on "Smart Trends In Systems, Security And Sustainability London, UK 29th-30th July 2021. Springer proceedings (LNNS).

[26]