International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Analysis of Class Imbalanced Brain Tumor Using Machine Learning Techniques

Prabhat Kumar Sahu[1], Mitrabinda Khuntia[2], Satish Choudhury[3], Binod Kumar Pattanayak[4]

Submitted: 29/12/2023 **Revised**: 05/02/2024 **Accepted**: 13//02/2024

Abstract: It is evident that healthcare has become a critical global priority, and the intelligent utilization of clinical datasets is essential for establishing an effective and efficient healthcare system capable of monitoring and managing people's health. However, the issue of class imbalance in real-world datasets, including clinical datasets, poses significant challenges to the training of classifiers and can result in reduced accuracy, precision, recall, and increased misclassifications. In our comprehensive literature review, we've examined the performance of five well-known classifiers—Logistic Regression, Decision Tree, Gaussian Naive Bayes, Random Forest, and Extra Tree classifiers—over imbalanced brain tumor datasets. We have also evaluated the effectiveness of four different class balancing techniques—SMOTE, ADASYN, ENN, and SMOTE-ENN—in addressing the challenges posed by imbalanced class distributions. The results of our study indicate that the SMOTE-ENN balancing approach has demonstrated superior performance compared to the other three data balancing strategies when used with all five classifiers. Additionally, although the other three balancing strategies, namely SMOTE, ADASYN, and ENN, performed relatively well, they slightly underperformed in comparison to the SMOTE-ENN approach. The identification of the SMOTE-ENN approach as the most effective strategy for handling imbalanced datasets is significant, as it highlights the importance of combining over-sampling and under-sampling techniques to achieve a more balanced and representative dataset for training classifiers. By effectively addressing the issue of class imbalance, the SMOTE-ENN approach allows for the development of more robust and accurate predictive models, thus improving the overall performance of the classifiers on imbalanced brain tumor datasets. Our study contributes valuable insights into the selection of appropriate data balancing strategies and classifier choices when dealing with imbalanced datasets in the healthcare domain. By providing a comprehensive overview of the empirical performance of different classifiers and balancing techniques, we have laid the foundation for implementing more effective and reliable supervised machine learning algorithms in the context of clinical data analysis. The recommendations we offer for dealing with class imbalanced datasets further enhance the practical applicability of our research findings.

Keywords: *Smote, Enn, Smote-Enn, Adasyn, Decision Tree, Logistoc Regression, Random Forest, Gaussian Naïve Bayes, Extra Tree Classifiers*

1. Introduction

Cancer is one of the most serious health diseases as well as difficulties confronting humans today. Because of rapid improvements in medical technology, the age of critical medical data is rapidly coming. Investigation, interruption, along with diagnosis of infectious tumors are all dependent on data handling, as well as suitable evaluation in infectious tumor diagnosis as well as therapy [1],[2],[3]. Brain tumors are the deadliest and most lethal type of cancer [4], [5]. The brain is the nerve control centre system that controls the total human body's organs. As a result, having an atypical brain has a negative impact on patient's health. According to the World Health Organisation (WHO), around 10 million deaths from brain cancer will be documented in 2020, making it the second highest cause of death worldwide. Cancer is regarded as the most lethal and destructive illness due to its many features, low survival rate along with the aggressive nature. Misdiagnosed brain tumors result in inefficient medical therapy, lowering the patient's odds of life [5], [6]. Pituitary, glioma, lymphomas, Medulloblastoma, malignant, as well as auditory neuroma are all various types of tumors depending on

1Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, INDIA
2,4Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, INDIA
3Department of Electrical Engineering, Siksha 'O' Anusandhan (Deemed to be) University, Bhubaneswar, INDIA

their texture, location, and form [7],[8].

The variety in tumor location, kind, size, and form creates a significant barrier in identifying brain tumors. A brain tumor is diagnosed based on its kind as well as location so that physicians can predict the patient's prospects of survival and make treatment decisions ranging from surgery to radiation or chemotherapy [8]. As a result, recognizing and diagnosing brain tumors at an early-stage aids in treatment planning and patient monitoring. It is crucial in enhancing therapy and raising survival chances. To obtain information on tumors, a variety of medical imaging as well as diagnostic approaches are performed [9]. Because of the diversity of tumors, MRI pictures may include no discernible characteristics that would allow for effective decision-making. As a result, humans cannot rely on natural diagnosis. A proper diagnosis allows the patient to receive appropriate therapy and live a long life. Furthermore, the brain tumor is dangerous since it reduces the efficiency of therapy along with the odds of survival [42], [43], [44]. As a result, the use of Artificial Intelligence (AI) approaches in computer-aided diagnostic (CAD) systems' capacity to diagnose medical images such as MRI scans has become required. These strategies aid clinicians and radiologists in making accurate diagnoses while also lowering workload [10], [11]. Machine learning has emerged as one of the most significant as well as prominent disciplines of artificial intelligence technologies, with applications in a wide range of fields. The flaws and disadvantages of each technique are evaluated using numerous criteria, including performance and scalability. Machine learning methods for classification are often constructed with the premise that each class has an equal number of examples [12]. Furthermore, because to the wide range of baseline models available, ensemble learning can reduce the risk of overfitting. Ensemble learning has been shown to outperform solo models in a range of industries and circumstances [13-16]. Ensemble learning techniques combine many models to create a more comprehensive and robust model [30]. There are several ensemble ways for training and integrating distinct baseline models. The most common ensemble techniques consist of averaging, bagging, random forest, stacking, and boosting. The literature has various assessments of ensemble learning methods and tactics [3],[17],[18]. Classic ensemble learning is based on the integration of classic machine learning models and their use in a variety of industries [19-23]. These attempts, however, were restricted to rudimentary single models. Several attempts have been undertaken in recent years to adapt ensemble learning to deep learning [24-29]. The majority of these efforts, however, are represented in the average voting strategy of basic DL models. In contrast, using average voting procedures to combine baseline learners biases the ensemble process towards weak baseline learners. Although there are several methods for combining baseline learners that may be employed in ensemble deep learning, these strategies have limitations of generalisation [28].

The Class Imbalance (CI) problem exists in a variety of fields, and several techniques to overcoming the problem have been offered during the last decade for a review. As illustrated in Fig.1, CI problem methodologies may be separated into three basic kind: (i) data level strategy, (ii) algorithm level strategy, (iii) hybrid strategy. The resampling technique is employed at the data level strategy (known as pre-processing approaches) to manage CI concerns in unbalanced datasets. Data level strategy is further differentiated into under sampling (balancing data by eliminating observations from the majority class), oversampling (balancing data by adding observations to the minority class), along with the hybrid, which is a mixture of previous two methods of sampling. Random oversampling, which augments the minority class by making identical duplicates of minority class observations, and random under sampling, which eliminates minority class observations at random, are the simplest data level techniques. An algorithm level method may be used to build or improve existing algorithms, as well as to investigate the implications of minor classes in dealing with imbalanced data. To solve the problem of class imbalance, the hybrid approach employs both data and algorithm level solutions. Data-level technique for balancing class data is more successful than other two strategies. It is utilised during the data pre-processing step. As a result, the purpose of this study is to create a performance assessment setup as well as evaluate the performance implications of key data balancing strategies with numerous classification algorithms on brain tumor data.

For binary class data, imbalance ratio (IR) is ratio of no. of samples from majority class (S_{maj}) to no. of samples from minority class (S_{min}).

$$IR = S_{maj} / S_{min} \quad (1)$$

Working with unbalanced data has the problem of forcing most ML algorithms to overlook and hence perform poorly on minority class, despite the fact that

performance on minority class is frequently coveted. Because ML algorithms try to improve accuracy by minimising error, they do not take into account class distribution. One strategy for coping with imbalanced datasets is to oversample the minority class. The most basic technique is to reproduce cases from minority class, even if these examples do not offer any new information to the model. Instead, new instances may be created by combining old ones. The Synthetic Minority Oversampling approach (SMOTE) is a minorities-specific data augmentation approach.

Classification analysis can also benefit from up-sampling, down-sampling, and the SMOTE, depending on no of patients in each class. SMOTE generates a more balanced dataset by artificially mixing the data to generate sufficient synthetic samples for minority group. SMOTE generates a more balanced dataset by artificially mixing the data to generate sufficient synthetic samples for minority group.

Purpose of this research is to determine acceptable assessment measures for imbalanced data models that have been pre-processed with SMOTE, SMOTE-ENN, and ADASYN. To handle challenges such as regression and classification, ensemble learning strategically blends classifiers or expert models [31],[32]. The three primary kinds of ensemble learning are bagging, boosting, and stacking [33-34]. Ensemble approaches, in essence, build numerous alternative predictive models from different copies of training data (usually re-weighted/re-sampled), and then aggregate predictions of these models in some way, generally by simple averaging/voting (potentially weighted). Ensemble learning algorithm's fundamental idea is to build numerous classifiers with poor generalisation performance and then apply a specific technique to merge them into a classifier with good generalisation performance. As a result, the ensemble's performance outperforms that of a single classifier.

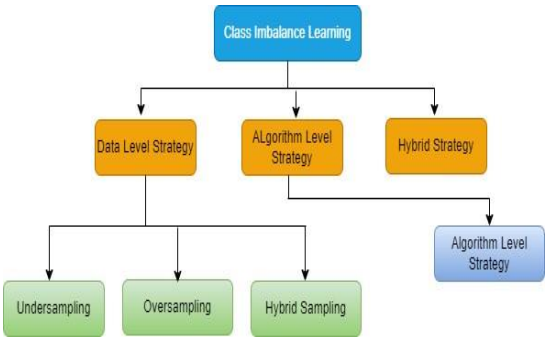

Fig. 1. Classification of Class Imbalance Learning

2. Literature Survey

Data is required in machine learning to train the model. In reality, we are constantly confronted with the problem of uneven data. Because most clinical datasets are fundamentally imbalanced, this section highlights the work done to increase the efficiency of certain machine learning algorithms while dealing with distinct clinical datasets. To counteract the impacts of imbalance, many algorithms are offered. The most common methods are investigated and assessed for dataset balance, and then various ML approaches are used to evaluate their performance.

Undersampling along with random sampling for majority as well as minority occurrences can aid in distribution shift of the original data. SMOTE is used to address the limitations of traditional sampling methods, such as the danger of overfitting with oversampling and the risk of information loss with undersampling [35]. For data balance, M. Mostafizur Rahman along with D. N. Davis suggested a modified cluster-based undersampling technique, and a high-quality training set was acquired for developing classification models [37]. SMOTE employs a unique oversampling strategy. Undersampling combined with SMOTE surpasses pure undersampling. SMOTE was applied to a wide range of datasets with varied degrees of imbalance and training datasets with varying volumes, resulting in a varying test field [35].

ADASYN can generate synthetic data samples for minority classes to counteract favouritism caused by unequal data distribution. Furthermore, ADASYN's capacity to alter boundaries and focus more on difficult-to-learn scenarios improves learning performance [36].

ENN is utilised to delete instances of both types [38]. SMOTE comes in about a hundred various flavours [39]. The structure of the flowchart is built using a

decision tree (DT), in which each node represents a test on an attribute value, each branch represents the outcome of the test operation, and the tree's leaf nodes represent classes. In a DT, classification is accomplished with minimum processing and easily generated rules [40]. Gaussian Nave Bayes (GNB) is used when the bulk of the features in a dataset are continuous. This method is based on the assumption that predictor values are samples from a Gaussian distribution [41].

3. Data Balancing Algorithms

Our primary objective is to investigate different balancing techniques on clinical datasets with varying degrees of imbalance. When the dataset for a machine learning model is large enough, the model performs better and more accurately. Data augmentation is used to improve the performance of such models by creating new data instances for training. To solve class imbalance, it is a set of ways for expanding the amount of data synthetically by manufacturing new data points from current data. Simple alterations to the present dataset, such as picture flipping, transformations, or rotation, would be required for a simple approach for artificially producing data. Data augmentation techniques have become essential in Deep Learning and graph data. It has also been used in a range of applications including Natural Language Processing, text classification, time series analysis, and tabular datasets. In our experiment, we employed three distinct balancing approaches to balance the datasets: SMOTE, ADASYN, and SMOTEEN. Following the balancing of the unbalanced datasets, five machine learning approaches, Linear Regression, Decision Tree, Gaussian Nave Bayes, Random Forest, and additional tree classifier, are used.

3.1. Synthetic Minority Oversampling Techniques

SMOTE was invented by [35]. It is one of the most used oversampling methods for dealing with imbalance. It attempts to equalise class distribution by randomly introducing minority class examples through replication. This strategy successfully solves the overfitting problem caused by random oversampling procedures. SMOTE creates new minority instances by combining existing minority instances. Due to the imbalance in the case-control groups, the classification analyses provided here employed an SMOTE sampling technique. The SMOTE is used to choose close-together instances in a feature space, create a line between the examples, then draw a new example at a place along the line. The minority class is oversampled in the SMOTE technique, which considers minority class data as well as generates synthetic samples in feature region based on the selected k in the KNN. SMOTE accepts the complete dataset as input, but only the few cases are increased in percentage. SMOTE offers the benefit of creating synthetic data points that differ slightly from the original data points rather than duplicating data points.

Algorithm SMOTE(T_{min}, N_s, k)
Output : (N_s/100) * T_{min}
1. If N < 100
a) then Randomize T_{min} minority class samples
b) T_{min} = (N_s/100) * T_{min}
c) N_s = 100
2. end if
3. N_s = (int) (N_s/100)
4. numattrs1 = No of attributes
5. newindex1 = 0
7. Synthetic[][] = 0
8. for i ← 1 to T
a) Compute k nearest neighbours for i, and save the indices in the nnarray.
b) Populate(N_s, i, nnarray)
9. end for

Populate(N_s, i, nnarray)
10. while N_s ≠ 0
a) A random number, nn is chosen between 1 and k.
b) for attr ← 1 to numattrs1
i) diff = Sample[nnarray[nn]][attr] – Sample[i][attr]
ii) gap = random no. between 0 & 1
ii) Synthetic[newindex1][attr] = Sample[i][attr] + gap * diff

c) end for
d) newindex1++
e) $N_s = N_s - 1$
11. end while
12. return
(End of Pseudocode)

The method is fed the no. of minority class samples (T_{min}), the amount of SMOTE % (N_s), the no. of nearest neighbours (k), and the output is the synthetic minority class samples ($N_s/100 * T_{min}$). If N_s is less than 100%, minority class samples should be randomised since only a random fraction of them will be SMOTEd. Sample[][] is an array for original minority class samples SMOTE amounts are considered to be integral multiples of 100. newindex1 keeps a count of no. of synthetic samples generated. Only for each minority class sample does Synthetic[][] array is an array for synthetic samples that calculate k closest neighbours. To produce synthetic samples, the Populate() function is utilised. The step 10(a) chooses one of the k nearest neighbours of i.

3.2. Adaptive Synthetic Sampling (ADASYN)

It was proposed by [36] and operates similarly to SMOTE in that it creates synthetic samples for minority classes based on actual dataset's feature space. It computes the density distribution of each minority class sample as well as creates synthetic samples based on that distribution. It is a strategy used in machine learning to solve unbalanced datasets, increasing classification performance for under-represented classes. ML algorithms are biased towards majority class because both datasets contain majority class with numerous samples along with a minority class with few examples,. ADASYN is a method of oversampling that creates synthetic samples for minority classes in order to balance the dataset and improve classification accuracy.

It is a more efficient and speedier approach to sample a population than the usual, fixed strategy. ADASYN can assist medical researchers balance datasets, enhancing the efficacy of machine learning models for identifying illnesses and forecasting patient outcomes.

Algorithm ADASYN($X_{original}$, bal1, k)
1. S_min, S_maj
2. G_all ← |S_maj| x bal1 - |S_min|
3. for each $x_i \epsilon$ S_min do
a) r[i] ← $\frac{|NNi \cap S_{maj}|}{k}$
4. end for
5. for each $x_i \epsilon$ S_min do
a) r̂[i] ← $\frac{r[i]}{\Sigma r[i]}$
b) G[i] ← int (r̂[i] x G_all)
6. end for
7. Syn1 ← Ø
8. for each $x_i \epsilon$ S_min do
a) K_i ← k nearest neighbour of x_i in S_min
b) for j = 1 to G[i] do
i) n ← a sample randomly chosen from K_i
ii) diff ← n – x_i
iii) gap ← random value between [0, 1]
iv) syn1 ← x_i + gap * diff
v) Syn1 ← Syn1 U {syn1}
c) end for
9. end for
10. return $X_{result} = X_{result}$ U Syn1

The original training dataset, $X_{original}$, a balancing parameter bal1, and the no. of nearest neighbours, k, are fed into the algorithm, which produces a new training dataset, X_{result}. S_min is used in $X_{original}$ as a collection of minority samples, whereas S_maj is used as set of majority samples. G_all is used to represent the entire no. of samples to be synthesised. In $X_{original}$, NN_i represents x_i's k nearest neighbours. Ratio of majority samples in k closest neighbours of a minority sample x_i is given by r[i]. The no. of

samples to be synthesised from x_i is given by G[i]. The freshly synthesised sample is called syn1.

3.3 Edit Nearest Neighbors

The Edited Nearest Neighbors (ENN) technique is a data cleaning method that aims to improve the quality of imbalanced datasets by selectively removing instances from the majority class that might be misclassified. ENN works by iteratively examining each data point and comparing its class label with those of its nearest neighbors. The algorithm follows specific steps to identify and remove potential noisy or misclassified instances from the dataset. Below is a detailed explanation of the ENN algorithm:

- Step 1: Determining the value of k: The algorithm starts by determining the number of nearest neighbors, denoted as 'k', for each observation in the dataset. If the value of k cannot be precisely calculated, it is often set to a default value, typically 3, to begin the process.
- Step 2: Finding the k-nearest neighbors: For each observation, the algorithm identifies its k-nearest neighbors from the rest of the dataset. It then examines the class labels of these nearest neighbors and calculates the majority class among them.
- Step 3: Comparing the class labels: The algorithm checks if the class label of the current observation matches the majority class obtained from its k-nearest neighbors. If the two differ, it indicates a potential misclassification. In such cases, the algorithm marks the current observation and its k-nearest neighbor for removal from the dataset.
- Step 4: Repeating the process Steps 2 and 3 are repeated iteratively until the desired proportion of each class is achieved or until the algorithm converges. This iterative process helps in gradually improving the overall balance of the dataset by selectively removing instances that might be causing noise or inaccuracies in the classification process.

By implementing the ENN algorithm, researchers and practitioners can effectively enhance the quality of imbalanced datasets, leading to improved classification performance and more accurate predictive models, especially in scenarios where the imbalance between classes can significantly affect the overall performance of machine learning algorithms.

3.4 SMOTE-ENN

[38] developed the SMOTE-ENN Algorithm which is an oversampling technique. This approach combines It is one method of transforming an unbalanced dataset into a balanced dataset. It simultaneously upsamples as well as downsamples. The primary concept is to interpolate between a large number of neighbouring minority class instances to generate new minority class instances. It decreases the likelihood of overfitting caused by synthetic instances.

Algorithm SMOTE-ENN

Input: Dataset: X, minority sample x_{i_min} , i = 1, 2 …, N, majority sample x_{j_maj} , j = 1, 2,…, M

Output: Sampled dataset X'

1. The over-sampling rate IR is set according to the sample imbalance rate.

2. for I = 1, 2, …, N do

3. For each minority sample x_{i_min} , calculate the distance of x_{i_min} to all samples in the minority according to the Euclidean distance and get k_1 nearest neighbours samples x_{ik1_min}

4. for l = 1, 2, …, IR do

5. For each minority sample x_{i_min} , a no. of samples are randomly selected from its k_1 nearest neighbours, assuming that the selected nearest neighbours are x_{ik1_min}

6. For each randomly selected nearest neighbour sample x_{ik1_min} , synthesize new minority sample x_{new} with the minority sample x_{i_min} according to

$$H_{bagging}(x) = \text{argmax}_{y \in Y} \sum_{t=1}^{T} I(h_t(x) = y), \quad y = 1, 2, \ldots, L,$$

where I() is an indicative function i.e. I(true) = 1, I(false) = 0.

7. Add synthesized new minority sample x_{new} to the original minority.

8. end for

9. end for

10. for x_{j_maj} , j = 1, 2, …, M do

11. For each majority sample x_{j_maj} , calculate the distance between x_{j_maj} and majority samples in the majority according to the Euclidean distance and get k_2 nearest neighbours samples x_{ik2_maj} .

12. For each majority sample x_{j_maj} , select 3 nearest neighbour samples from its k_2 nearest neighbours, assuming that selected nearest neighbour samples are x_{ik2_maj} .

13. For each majority sample x_{j_maj} determine whether it is "noisy sample" according to

$$H_{Adaboost}(x) = argmax_{y \epsilon Y} \Sigma^{T}_{t=1} In(1/\beta)\, I(h_t(x) = y),\ y = 1, 2, \ldots, L,$$

where β is the weight that emphasises sample weight adjustment and weighting coefficient of weak classifier.

If it is "noisy sample", delete x_{j_maj} , otherwise keep x_{j_maj}

14. Remove the "noisy sample " from the majority

15. End for

16. return X'

In step 6, combination order of weak classifier T_1, T_2, …., T_t randomly generates the weak classifier, $h_t(x)$.

4. Classification Algorithms

A brief description of each classification methodology used in this study is provided below in order to provide basic knowledge about these classification approach.

4.1 Logistic Regression (LR) Classifier

Linear regression is a statistical modelling approach that uses a continuous response variable to represent it as a linear function of one or more predictors. It predicts the value of one variable depending on the value of another. The independent variable is the one used to predict the value of the other variable. It is a type of ML approach, specifically a supervised ML algorithm, that learns from labelled datasets and maps data points to the best optimised linear functions, which may subsequently be used to forecast new datasets. It creates a linear relationship between one or more independent variables and one or more dependent variables. The aim is to develop optimum linear equation for predicting the value of the dependent variable based on the independent variables. Equation (2) can be used to express the estimated regression model.

$$P = (e^{\beta 0 + \beta 1 x1}) / (1 + e^{\beta 0 + \beta 1 x1}) \qquad (2)$$

4.2 Decision Tree (DT) Classifier

A Decision Tree Classifier is most commonly utilized for classification purposes that provides a clear and intuitive representation of the decision-making process by structuring the data in a tree-like format. The core components of a Decision Tree include the following elements:

- Decision Node: This node is responsible for making decisions based on specific attributes or features of the dataset. It serves as the starting point for the decision-making process and has multiple branches representing different possible outcomes or choices based on the values of the selected attributes.
- Leaf Node: The leaf node signifies the final outcome or prediction based on the decisions made at the decision nodes. It represents the end of the decision-making process for a particular path in the tree and does not contain any further branches.
- Decision Trees are advantageous in that they can handle both categorical and numerical data, making them suitable for a wide range of applications across different domains. The algorithm recursively partitions the dataset based on the values of the input features, aiming to create the most effective and informative decision rules to accurately classify or predict the target variable.

The simplicity and interpretability of Decision Trees make them highly popular for tasks that require transparent and easily understandable models. Moreover, they can handle complex decision boundaries and interactions between features, which can be beneficial in capturing intricate relationships within the data. To address this issue, various strategies such as pruning, setting constraints on tree depth, and implementing ensemble methods like Random Forests or Gradient Boosting Trees are commonly employed.

4.3 Random Forest (RF) Classifier

It is a supervised ML approach that is commonly used in classifying as well as predicting issues. We call it a Random Forest because we utilise random selections of data and attributes to create a forest of decision trees (many trees). Integration blends the concept of bagging integration with feature selection. Random Feature classifier is made up of a group of tree classifiers, each of which is produced using a random vector that is independent of the input vector samples, and each tree votes for the classes with the highest votes to classify the input vector. The restriction of DT are circumvented by the RF

classifier. It decreases dataset overfitting while increasing accuracy. Numerous research done throughout the world have proven that the Random Forest algorithm performs exceptionally well in classification and prediction in a variety of domains.

4.4 Gaussian Naïve Bayes (GNB) Classifier

Gaussian Nave Bayes is a probabilistic ML classification strategy based on the Gaussian Distribution. According to Gaussian Nave Bayes, each parameter (also known as a feature or predictor) may independently predict the result. The total of all parameter predictions yields a probability for the dependent variable to be categorised in each group as the final prediction. The group with the highest chance receives the final classification. When we assume that all of the continuous variables associated with each feature have a Gaussian distribution, we utilise Gaussian Nave Bayes. The Gaussian distribution is sometimes known as the normal distribution. The formula in equation (3) gives the conditional probability.

$$P(xi \mid y) = \frac{1}{\sqrt{2\prod\sigma^{\wedge}2}} \exp\left((xi - \mu y)2 / (2\sigma y2)\right) \qquad (3)$$

where μy and σy are mean and variance of predictor distribution.

4.5 Extra Tree (XTree) Classifier

Extra tress (also known as Extremely Randomised trees) classifier is an ensemble supervised ML approach that predicts using multiple decision trees. Extra trees train decision trees using the complete dataset, which allows Extra trees to rely on randomization to decrease variation and computational costs. Predictive models are generated for classification and regression challenges. It is a DT ensemble and is related to bootstrap aggregation (bagging) along with RF. It is more efficient than the DT and less difficult to build than other approaches.

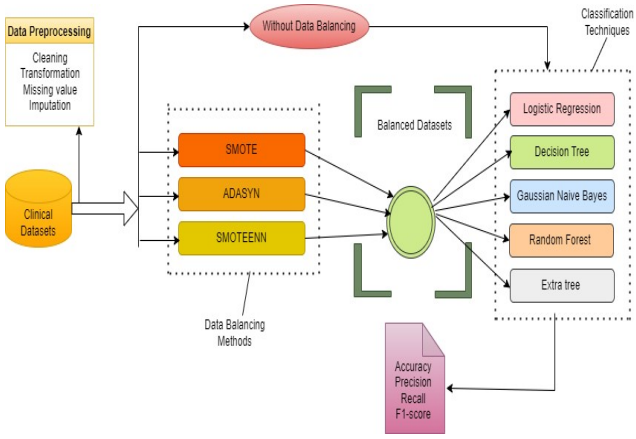

Fig. 2. Experimental setup for evaluation of classifiers over clinical datasets

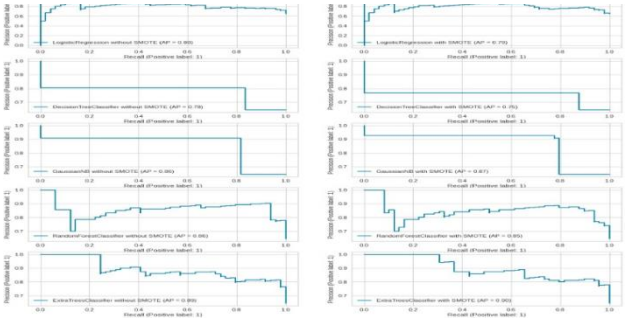

Fig 3. Precision – Recall Value for SMOTE Technique

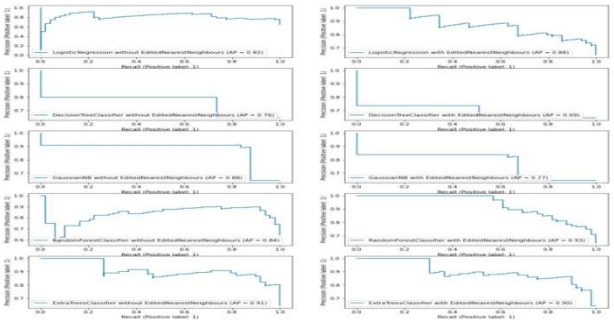

Fig 4. Precision – Recall Value for ENN Technique

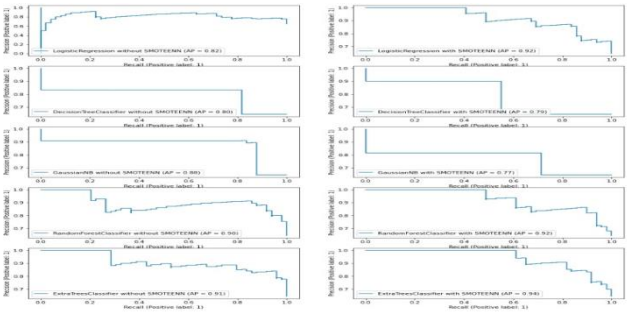

Fig 5. Precision – Recall Value for SMOTE-ENN Technique

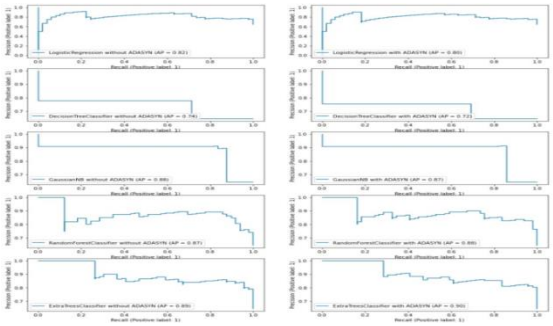

Fig 6. Precision-Recall Curve with ADASYN technique

5. Experimental Result of Imbalanced Dataset

5.1. Dataset

The use of medical records for specific disease diagnosis is crucial for advancing healthcare solutions and improving the accuracy of medical diagnosis software systems. By leveraging datasets obtained from the Kaggle database repository, we have access to a diverse range of data that can offer valuable insights into different aspects of the targeted disease. The application of oversampling techniques to handle the binary class imbalance is a prudent approach, as it allows for the creation of a more balanced dataset, thereby enabling the development of robust and reliable predictive models. The fact that these datasets vary in terms of class imbalances, size, features, and numbers highlights the complex and diverse nature of medical data. This diversity emphasizes the importance of utilizing oversampling techniques to address the challenges posed by imbalanced datasets, enabling a more comprehensive analysis and interpretation of the data. By ensuring that the datasets are well-prepared and properly processed, we can effectively enhance the reliability and accuracy of the experimental results. By taking these factors into account, we can ensure that the experimental results accurately reflect the real-world challenges and complexities associated with medical data analysis. The insights gained from the application of oversampling techniques to these medical datasets can significantly contribute to the development of effective and reliable healthcare solutions, ultimately leading to improved medical

diagnosis and treatment strategies. As we continue to analyze and interpret the experimental results, it is essential to emphasize the significance of these findings in the broader context of advancing medical research and clinical practice.

5.2. Experimental Setup

It's great to hear that the experiments were conducted to comprehensively evaluate the efficiency and effectiveness of various algorithms in handling imbalanced clinical datasets. Assessing classifier accuracy, precision, recall, and F1 score/F measure is essential in understanding the performance of these algorithms in real-world scenarios. The choice of conducting experiments on the cloud, particularly using the Python programming language in the Google Colab environment, offers flexibility, scalability, and access to a range of resources necessary for complex computations and analyses. Using the imbalance ratio (IR) as a fundamental criterion for evaluating the algorithm is a practical approach, as it allows for a standardized metric to assess the impact of the data balancing techniques across different datasets. By focusing on IR, we can effectively measure the extent to which the imbalanced nature of the data is addressed by the various balancing techniques. The experimental workflow, as depicted in the figure, is critical in illustrating the step-by-step process of the proposed work, from data preprocessing and balancing to the application of different classifiers and the evaluation of their performance using various metrics. This visualization provides a clear overview of the entire experimental setup, making it easier to understand and replicate the process. It is crucial to document the experimental workflow thoroughly, including details of the datasets used, the specific parameters of the balancing techniques, the classifiers employed, and the specific evaluation metrics considered. This documentation helps in ensuring the transparency and reproducibility of the experiments, enabling other researchers to validate and build upon our findings. By conducting comprehensive empirical performance analysis, we contribute to the growing body of knowledge in the field of data imbalance handling in clinical datasets, providing valuable insights into the effectiveness of different approaches and their applicability to real-world scenarios. Such empirical analyses serve as a crucial foundation for the development of robust and reliable diagnostic tools and predictive models in the healthcare domain.

6. Result and Discussion

Experiments were carried out to evaluate three balancing approaches and five classification techniques on class unbalanced datasets. The classification results were evaluated using well-known performance indicators such as Precision, Recall, F1 score, and Average Precision-Recall. After pre-processing the brain tumor illness dataset, each of the three data balancing procedures - SMOTE, ADASYN, and SMOTEEN - was applied independently. The balanced dataset was then evaluated against five major classifiers, as seen in the figure above. Experiments were carried out to evaluate three balancing approaches and five classification techniques on class unbalanced datasets. The classification results were evaluated using well-known performance indicators such as Precision, Recall, F1 score, and Average Precision-Recall as shown in the table1, table2, table3 and table4. After pre-processing the brain tumor illness dataset, each of the three data balancing procedures - SMOTE, ADASYN, and SMOTEEN - was applied independently. The balanced dataset was then evaluated against five major classifiers, as seen in the figure above.

Precision values for logistic regression, decision tree, Gaussian naive bayes, random forest, and extra trees classifiers using ADASYN and SMOTEENN approaches are the same or higher than precision values for all classifiers using SMOTE techniques. The recall and F1 score values for the decision tree classifier with SMOTEENN are much lower than those for the SMOTE and ADASYN approaches. The recall value for extra trees classifier using ADASYN is higher than that of the other two approaches. Thus, among all ML procedures, the balancing technique SMOTEEN for brain tumour illness has the greatest average precision-recall value.

The outcome analysis as shown in the fig.3, fig.4, fig.5 and fig.6 clearly shows that the SMOTEENN balancing approach outperformed all other balancing strategies for the brain tumour dataset. This is because SMOTEENN uses SMOTE and ENN to combine oversampling and undersampling. ENN tries to eliminate cases in all classes, therefore every instance that is misclassified will be removed from the training set. Undersampling underperformed in many circumstances because it removed potentially important examples from datasets.

Table 1:- Precision Value

	With SMOTE	Without SMOTE	With ADASYN	Without ADASYN	With ENN	Without ENN	With SMOTE-ENN	Without SMOTE-ENN
LR	0.7619	0.7619	0.7778	0.7778	0.7778	0.8667	0.7778	0.8889
DT	0.7447	0.7778	0.7843	0.7143	0.78	0.7576	0.8095	0.8333
GNB	0.8095	0.8139	0.9091	0.907	0.9091	0.8378	0.9091	0.8064
RF	0.8409	0.8181	0.8364	0.8461	0.849	0.933	0.8364	0.8333
XTree	0.8085	0.8444	0.807	0.8182	0.8103	0.871	0.8246	1.00

Table 2: F1- Score Value

	With SMOTE	Without SMOTE	With ADASYN	Without ADASYN	With ENN	Without ENN	With SMOTE-ENN	Without SMOTE-ENN
LR	0.7805	0.7805	0.8155	0.8155	0.8155	0.6582	0.8155	0.6316
DT	0.8046	0.8235	0.8	0.7368	0.7879	0.6098	0.7472	0.5479
GNB	0.8293	0.8434	0.8602	0.8478	0.8602	0.7209	0.8602	0.625
RF	0.8809	0.8571	0.8846	0.8713	0.8823	0.7088	0.8846	0.7692
XTree	0.8736	0.8941	0.868	0.8654	0.8785	0.675	0.8868	0.6933

Table 3: Recall Value

	With SMOTE	Without SMOTE	With ADASYN	Without ADASYN	With ENN	Without ENN	With SMOTE-ENN	Without SMOTE-ENN
LR	0.8	0.8	0.8571	0.8571	0.8571	0.5306	0.8571	0.4898
DT	0.875	0.875	0.8163	0.7143	0.7959	0.5102	0.6939	0.4082
GNB	0.85	0.875	0.8163	0.7959	0.8163	0.6326	0.8163	0.5102
RF	0.925	0.9	0.9388	0.8979	0.9184	0.5714	0.9388	0.7143
XTree	0.95	0.95	0.9388	0.9184	0.9592	0.551	0.9592	0.5306

Table 4: Average Precision-Recall Value

	With SMOTE	Without SMOTE	With ADASYN	Without ADASYN	With ENN	Without ENN	With SMOTE-ENN	Without SMOTE-ENN
LR	0.7148	0.7148	0.7588	0.7588	0.7588	0.7625	0.7588	07643
DT	0.7174	0.7463	0.7587	0.7277	0.7524	0.7023	0.7591	0.7217
GNB	0.767	0.778	0.8605	0.8534	0.8605	0.7669	0.8605	0.7272
RF	0.8173	0.789	0.8246	0.8256	0.8324	0.8096	0.8246	0.7794
XTree	0.7944	0.8285	0.7971	0.804	0.8036	0.7694	0.8172	0.8332

7. Conclusion

The problem of data imbalance is indeed a significant issue in various fields, particularly in medical diagnosis, where the scarcity of certain classes of data can significantly impact the performance of classification algorithms. Traditional algorithms tend to favour the majority class, leading to poor performance on the minority class. To mitigate this issue, various techniques have been developed, including the hybrid sampling approach using techniques like SMOTE, ADASYN, and SMOTE-ENN. SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling) are both oversampling techniques that aim to balance the class distribution by generating synthetic samples for the minority class. SMOTE creates synthetic samples along the line segments joining k minority class nearest neighbors, while ADASYN focuses on generating more synthetic data for those minority class samples that are difficult to classify. On the other hand, SMOTE-ENN combines the over-sampling approach of SMOTE with the under-sampling approach of Edited Nearest Neighbors (ENN) to further improve the balance in the dataset. The dynamic updating of the over-sampling rate during the hybrid sampling process in SMOTE-ENN is an important feature, as it allows the

algorithm to adapt to the specific characteristics of the dataset at hand. This adaptability can be crucial for achieving better performance in cases where the imbalance in the dataset is not uniform or changes over time. It's important to note that while these balancing algorithms can significantly improve the performance of classification algorithms on imbalanced datasets, there is no one-size-fits-all solution. The effectiveness of these techniques can vary depending on the specific characteristics of the dataset, the nature of the imbalance, and the specific classification algorithm being used. In the context of medical diagnosis, where accurate predictions are crucial, it's imperative to carefully evaluate and select the most appropriate balancing strategy based on the specific requirements and nuances of the dataset at hand. Additionally, considering the importance of data quality, it's essential to integrate machine learning (ML) approaches and advanced balancing algorithms to ensure reliable and accurate predictions in the medical domain. Regular monitoring and evaluation of the performance of these algorithms are necessary to ensure that the chosen approach is effectively addressing the data imbalance issue and improving the overall predictive accuracy.

8. References and Footnotes

Author contributions

Prabhat Kumar Sahu: Investigate conceptualization and design, Data collection, Analysis and interpretation of the results and Draft manuscript preparation **Mitrabinda Khuntia:** Investigate conceptualization and design, Data collection, Analysis and interpretation of the results and Draft manuscript preparation **Binod Kumar Pattanayak:** Analysis and interpretation of the results. All authors examined the findings and approved the final paper version.

Funding Information :

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Conflicts of Interest:

The authors declare no conflict of interest

Ethics:

Ethical considerations in research are a set of principles that guide your research designs and practices. These principles include voluntary participation, informed consent, anonymity, confidentiality, potential for harm, and results communication. Data, results, methods and procedures, and publication status are reported honestly. They are not fabricate, falsify, or misrepresent data. We have tried to avoid bias in experimental design, data analysis, data interpretation, peer review, personnel decisions, grant writing, expert testimony, and other aspects of research. We have carefully and critically examine your own work and the work of your peers. Keep good records of research activities.

References

[1] Akram, F., Liu, D., Zhao, P., Kryvinska, N., Abbas, S., & Rizwan, M. (2021). Trustworthy intrusion detection in e-healthcare systems. *Frontiers in public health*, *9*, 788347.

[2] Javed, A. R., Shahzad, F., ur Rehman, S., Zikria, Y. B., Razzak, I., Jalil, Z., & Xu, G. (2022). Future smart cities: Requirements, emerging technologies, applications, challenges, and future aspects. *Cities*, *129*, 103794.

[3] Zhang, L., Zhong, Q., & Yu, Z. (2021). Optimization of tumor disease monitoring in medical big data environment based on high-order simulated annealing neural network algorithm. *Computational Intelligence and Neuroscience*, *2021*, 1-9.

[4] Ali, T. M., Nawaz, A., Ur Rehman, A., Ahmad, R. Z., Javed, A. R., Gadekallu, T. R., ... & Wu, C. M. (2022). A sequential machine learning-cum-attention mechanism for effective segmentation of brain tumor. *Frontiers in Oncology*, *12*, 873268.

[5] Senan, E. M., Jadhav, M. E., Rassem, T. H., Aljaloud, A. S., Mohammed, B. A., & Al-Mekhlafi, Z. G. (2022). Early diagnosis of brain tumour mri images using hybrid techniques between deep and machine learning. *Computational and Mathematical Methods in Medicine*, *2022*.

[6] Rathod, R., & Khan, R. A. H. (2021). Brain tumor detection using deep neural network and machine learning algorithm. *PalArch's Journal of Archaeology of Egypt/Egyptology*, *18*(08), 1085-1093.

[7] Alanazi, M. F., Ali, M. U., Hussain, S. J., Zafar, A., Mohatram, M., Irfan, M., ... & Albarrak, A. M. (2022). Brain tumor/mass classification framework using magnetic-resonance-imaging-based isolated and developed transfer deep-learning model. *Sensors*, *22*(1), 372.

[8] Kumar, T. S., Arun, C., & Ezhumalai, P. (2022). An approach for brain tumor detection using optimal feature selection and optimized deep belief

network. *Biomedical Signal Processing and Control*, *73*, 103440.
[9] Alsaif, H., Guesmi, R., Alshammari, B. M., Hamrouni, T., Guesmi, T., Alzamil, A., & Belguesmi, L. (2022). A novel data augmentation-based brain tumor detection using convolutional neural network. *Applied Sciences*, *12*(8), 3773.
[10] Al-Shoukry, S., Rassem, T. H., & Makbol, N. M. (2020). Alzheimer's diseases detection by using deep learning algorithms: a mini-review. *IEEE Access*, *8*, 77131-77141.
[11] Gab Allah, A. M., Sarhan, A. M., & Elshennawy, N. M. (2021). Classification of brain MRI tumor images based on deep learning PGGAN augmentation. *Diagnostics*, *11*(12), 2343.
[12] Kumar, V., Aydav, P.S.S., Minz, S., 2021. Multi-view ensemble learning using multiobjective particle swarm optimization for high dimensional data classification. J. King Saud Univ.-Comput. Informat. Sci.
[13] Anwar, H., Qamar, U., Muzaffar Qureshi, A.W., 2014. Global optimization ensemble model for classification methods. Sci. World J. 2014.
[14] Shahzad, R.K., Lavesson, N., 2013. Comparative analysis of voting schemes for ensemble-based malware detection. J. Wireless Mobile Netw., Ubiquitous Comput. Dependable Appl. 4 (1), 98–117.
[15] Prusa, J., Khoshgoftaar, T.M., Dittman, D.J., 2015. Using ensemble learners to improve classifier performance on tweet sentiment data. 2015 IEEE International Conference on Information Reuse and Integration. IEEE, pp. 252–257.
[16] Ekbal, A., Saha, S., 2011. A multiobjective simulated annealing approach for classifier ensemble: Named entity recognition in indian languages as case studies. Expert Syst. Appl. 38(12), 14 760–14 772.
[17] Krawczyk, B., Minku, L.L., Gama, J., Stefanowski, J., Woz´ niak, M., 2017. Ensemble learning for data stream analysis: A survey. Informat. Fusion 37, 132–156.
[18] Dong, X., Yu, Z., Cao, W., Shi, Y., Ma, Q., 2020. A survey on ensemble learning. Front. Comput. Sci. 14 (2), 241–258.
[19] Tsai, C.-F., Lin, Y.-C., Yen, D.C., Chen, Y.-M., 2011. Predicting stock returns by classifier ensembles. Appl. Soft Comput. 11 (2), 2452–2459.
[20] Abellán, J., Mantas, C.J., 2014. Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. Expert Syst. Appl. 41 (8), 3825–3830.
[21] Catal, C., Tufekci, S., Pirmit, E., Kocabag, G., 2015. On the use of ensemble of classifiers for accelerometer-based activity recognition. Appl. Soft Comput. 37, 1018–1022.
[22] Da Silva, N.F., Hruschka, E.R., Hruschka Jr, E.R., 2014. Tweet sentiment analysis with classifier ensembles. Decis. Support Syst. 66, 170–179.
[23] Aburomman, A.A., Reaz, M.B.I., 2016. A novel svm-knn-pso ensemble method for intrusion detection system. Appl. Soft Comput. 38, 360–372.
[24] Haralabopoulos, G., Anagnostopoulos, I., McAuley, D., 2020. Ensemble deep learning for multilabel binary classification of user-generated content. Algorithms 13 (4), 83.
[25] Alharbi, A., Kalkatawi, M., Taileb, M., 2021. Arabic sentiment analysis using deep learning and ensemble methods. Arabian J. Sci. Eng. 46 (9), 8913–8923.
[26] Can Malli, R., Aygun, M., Kemal Ekenel, H., 2016. Apparent age estimation using ensemble of deep learning models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 9–16.
[27] Ortiz, A., Munilla, J., Gorriz, J.M., Ramirez, J., 2016. Ensembles of deep learning architectures for the early diagnosis of the alzheimer's disease. Int. J. Neural Syst. 26 (07), 1650025.
[28] Tasci, E., Uluturk, C., Ugur, A., 2021. A voting-based ensemble deep learning method focusing on image augmentation and preprocessing variations for tuberculosis detection. Neural Comput. Appl., 1–15.
[29] Xu, S., Liang, H., Baldwin, T., 2016. Unimelb at semeval-2016 tasks 4a and 4b: An ensemble of neural networks and a word2vec based model for sentiment classification. In: Proceedings of the 10th international Workshop on Semantic Evaluation (SemEval-2016), pp. 183–189.
[30] J. Xiao, Svm and knn ensemble learning for tra_c incident detection, Physica A: Statistical Mechanics and its Applications 517 (2019) 29–35.
[31] R. Polikar, Ensemble learning, in: Ensemble machine learning, Springer, 2012, pp. 1–34.
[32] T. G. Dietterich, et al., Ensemble learning, The handbook of brain theory and neural networks 2 (2002) 110–125.
[33] B. Zenko, L. Todorovski, S. Dzeroski, A comparison of stacking with meta decision trees to bagging, boosting, and stacking with other methods, in: Proceedings 2001 IEEE International Conference on Data Mining, IEEE, 2001, pp. 669–670.

[34] X. Dong, Z. Yu, W. Cao, Y. Shi, Q. Ma, A survey on ensemble learning, Frontiers of Computer Science 14 (2020) 241–258.
[35] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.
[36] He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)* (pp. 1322-1328). Ieee.
[37] Rahman, M. M., & Davis, D. N. (2013). Addressing the class imbalance problem in medical datasets. *International Journal of Machine Learning and Computing*, *3*(2), 224.
[38] Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, *6*(1), 20-29.
[39] Kovács, G. (2019). An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets. *Applied Soft Computing*, *83*, 105662.
[40] Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, *21*(3), 660-674.
[41] Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. In *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings* (pp. 986-996). Springer Berlin Heidelberg.
[42] Khuntia, M., Sahu, P. K., & Devi, S. (2022). Prediction of Presence of Brain Tumor Utilizing Some State-of-the-Art Machine Learning Approaches. *International Journal of Advanced Computer Science and Applications*, *13*(5).
[43] Khuntia, M., Sahu, P. K., & Devi, S. (2022). Novel Strategies Employing Deep Learning Techniques for Classifying Pathological Brain from MR Images. *International Journal of Advanced Computer Science and Applications*, *13*(11).
[44] Khuntia, M., Sahu, P. K., & Devi, S. (2023). Deep Learning Approaches for Brain Tumor Diagnosis using Fused Layer Accelerator. *Journal of Computer Science, 19(2*).