International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-2, December, 2019

Classification of Imbalanced Big Data using SMOTE with Rough Random Forest

Tanuja Das, Abhinandan Khan, Goutam Saha

Abstract: ***Learning from datasets is an important research topic today. Amongst the various data mining tools available for the purpose, none works satisfactorily in the case of imbalanced data mainly because this type of data gives rise to various minority classes, which may affect the learning process. In addition to the large volume, characteristics of Big Data also include velocity and variety. The Synthetic Minority Oversampling Technique (SMOTE) is a widely used technique to balance imbalanced data. Here, we have focussed on extending this concept to conform to the Big Data environment by combining it with the concepts of rough random forest (RRF). This hybrid approach comprising SMOTE and RRF algorithms for learning from imbalanced datasets has been applied on various benchmark datasets from the KEEL Dataset Repository. The results obtained are satisfactory. The velocity aspect of Big Data has been handled by this method on the dynamic dataset of the stock market. The results obtained have been verified using popular online websites related to stock markets.***

Keywords: ***big data, rough set theory, random forest, rough random forest, SMOTE, stock market data.***

I. INTRODUCTION

Extraction of knowledge from big datasets like those generated in areas such as healthcare, financial businesses, or telecommunications is the focus of researchers nowadays. However, the problem with these real-world datasets is that they are often imbalanced. For instance, a medical diagnosis dataset used by Kibler *et al*.(1987)contains samples that comply to the diagnosis of an anomalous disease, and only 5% of the sample correlate to positive diagnosis, i.e. most of the remaining samples belong to the benign group. Learning from these imbalanced datasets hampers the efficiency of machine learning systems that are designed to perform relatively well for balanced datasets. The flaw persists as these algorithms try to increase the accuracy on the majority cases (Provost, 2000) and, thus the performance suffers in the case of minority instances. Also, time-series forecasting of data, e.g., from financial businesses, is quite tough as these non-stationary instances makes the environment for prediction very complex (Chawla, 2009). The forecasting tasks which involve time-series are required to predict the unusual values. Thus, classification and extraction of information from these types of imbalanced data invite research initiatives, which will develop newer machine learning algorithms that are more relevant for the purpose.

From contemporary literature, it is found that several techniques already exists which tries improve the performance of a classifier from the imbalanced datasets, the most common being the practice of re-sampling strategies which learns from the data by changing its categories in favour of a given perspective. The Synthetic Minority Oversampling TEchnique (SMOTE) is often used to convert an imbalanced dataset into a balanced one. Though SMOTE improves the classification capability of imbalanced data substantially, it has been further enhanced using techniques like SMOTE with Tomek Links (SMOTE-TL) (Batista *et al*., 2004). Gustavo *et al*.(2004) applied SMOTE to oversample the data and next used data cleaning methods like Tomek links and Edited Nearest Neighbour Rule to remove noisy data lying on the wrong side of the decision border. SMOTE with Edited Nearest Neighbours (SMOTE-ENN) has been implemented by Batista *et al*.(2004). Gustavo *et al*. (2004), on the other hand, used SMOTE to oversample the data and then cleaned the data with the ENN prototype selection technique.

SMOTE with Rough Set Theory (SMOTE-RSB*) (Ramentol *et al*., 2012a) unclutters the data after application of SMOTE by eliminating samples that do not satisfy the rough lower approximation. Another approach is the SMOTE with Fuzzy Rough Set Theory, namely, SMOTE-FRST (Ramentol *et al*., 2012b), where the SMOTE-RSB technique utilizes the concept of fuzzy logic for rough set theory. Borderline SMOTE (SMOTE-BL1 and SMOTE-BL2) (Han *et al*., 2005) works by enhances the borderline. Another algorithm, namely, the Safe-level SMOTE (SMOTE-SL) (Bunkhumpornpat *et al*., 2009) assigns a safe level to each minority instance before creating artificial instances so that they are created only in the safe locality. A technique to improvise SMOTE has been given by Barua *et al*. (2014), wherein the method MWMOTE utilizes the most informative minority samples by assigning weights to them. Chennuru *et al*. (2017), in order to enhance the classification of minority instances, developed a method called MahalCUSFilter, which is basically an undersampling technique.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author
Tanuja Das, Department of Information Technology, Gauhati University Institute of Science and Technology, Guwahati, Assam – 781014, India. Email: tanujadas55@gmail.com
Abhinandan Khan*, Department of Computer Science and Technology, University of Calcutta, Acharya Prafulla Chandra Roy Siksha Prangan, JD-2, Sector-III, Saltlake, Kolkata – 700106, India.
Goutam Saha, Department of Information Technology, North-Eastern Hill University, Shillong – 793022, Meghalaya, India. Email: dr.goutamsaha@gmail.com

As the size of the data continues to grow exponentially, and the patterns of the data continuously varying over time, new problems and challenges of volume, velocity, and variety emerge in the data processing domain. Also, other aspects of Big Data like veracity or value must also be taken care of. In order to evolve the previous algorithms or to create new ones for Big Data, the scalability factor should be tackled well. In this paper, we have focussed on the extraction of information from imbalanced Big Data, especially focusing on its volume and velocity issues.

For classification purposes, Random Forest (RF) has widely been used. Literature has provided evidence about the superiority of RF technique as compared to other methods (Dietterich, 2000). The performance of RF is based on the performance of its individual decision trees. The accuracy of the RF algorithm will degrade over time if it is not adapted to the changing needs of the Big Data domain. Aiming at this problem, Liu (2014) presents an approach on how to realise the self-adaptation ability with the random forest method in similar situations. Jinmao *et al*. (2002) had shown that the efficiency of the decision tree combined with Rough Set Theory (RST) is more efficient than most of the other machine learning techniques. Rajhans *et al*. (2015) proposed the Rough Random Forest (RRF) for classification with an encouraging result. In this methodology, bagging and random feature selection has been used to form a number of rough decision trees. The authors used this approach to classify data, especially the sparse data that gave very good classification accuracy. Thus, in order to handle the large volume as well as velocity issues of Big Data, we have tried to combine the concept of SMOTE along with RRF. The original contributions of combining SMOTE with RRF can be summarized as follows. Although SMOTE is widely used by many researchers to develop models, solving class-imbalance problems for datasets with higher dimensions is still a challenging task. It is because the computational complexity and memory requirements of SMOTE increases when handling large-scale imbalanced data (Krawczyk, 2016). This motivates us to develop a modified approach that can provide a quality solution even if the dataset is very large.

Here, we have used the Synthetic Minority Oversampling TEchnique (SMOTE) methodology in combination with an ensemble learning model, known as the RRF, to extract information classes from it. RRF has been constructed by amalgamating three basic building techniques, namely RST, Decision Tree, and RF. Here, the reducts of decision trees extracted from RRF have been selected on the basis of the boundary region of attributes. We have found much-improved results using these combined techniques. For imbalanced Big Data classification, we have used the Sliding Window (Bhatotia *et al*.,2012) based implementation of SMOTE+RRF. The whole work has been implemented using the R platform.

Section 2 of this paper discusses the basics of SMOTE, RST, RF, RRF, SMOTE for Big Data and RRF for Big Data. Section 3 discusses the methodology adopted for this work. Section 4 discusses the results of our method on the benchmark datasets and provides a comparison with a few previously used techniques available in the literature. The paper concludes with Section 5.

II. BACKGROUND

In this section, we have given a background of our research, starting with an overview of SMOTE in Section 2.1, followed by a concise idea of RST, RF, and RRF in Sections 2.2, 2.3, and 2.4, respectively. Sections 2.5 and 2.6 have been dedicated to the SMOTE algorithm and the RRF adaptations for Big Data.

A. SMOTE: Synthetic Minority Oversampling TEchnique

In SMOTE, "synthetic" instances are generated by over-sampling he minority class which is motivated by an approach that showed great potential in handwritten character recognition (Ha et al., 1997). SMOTE (Chawla et al., 2002) and its variants are extensively used in solving the class imbalance problems. This is because the algorithm is simple as well as robust, which is suitable for dealing with different types of data. The main motive behind writing this algorithm is to gain insight as to how this algorithm oversamples the minority class samples and provides supervised inclination of the bias towards them. The algorithm SMOTE (Chawla et al., 2002) is as given below:

Algorithm SMOTE(T,N,k)

Input: number of minority class samples T; the amount of SMOTE $N\%$; and number of nearest neighbours k

Output: $\left(\frac{N}{100}\right) * T$ synthetic minority class samples

//If N is less than 100%, randomize the minority class samples as only a random percentage of them will be SMOTE-d.

If $N < 100$
    randomise the T minority class samples
    $T = \left(\frac{N}{100}\right) * T$
    $N = 100$
End If
$N = (int)\left(\frac{N}{100}\right)$
//The amount of SMOTE is assumed to be in integral multiples of 100.
k: number of nearest neighbours
$numattrs$: number of attributes
$Sample[\,][\,]$: array for original minority class samples
$newindex$: keeps a count of the number of synthetic samples generated, initialised to 0
$Synthetic[\,][\,]$: array for synthetic samples
//Compute k nearest neighbours for each minority class sample only.
For $i \leftarrow 1$ to T
    Compute k nearest neighbours for i, and save the indices in the $nnarray$
    Populate (N,i,$nnarray$)

```
15. End For
```

Algorithm Populate(N,i,$nnarray$)

```
//Function to generate synthetic samples.
1. While N ≠ 0
2.      Choose a random number between 1 and
        k, call it nn. This step chooses one
        of the k nearest neighbours of i.
3.      For attr ← 1 to numattrs
4.          Compute dif =
            Sample[nnarray][nn][attr]- Sample [i][attr]
5.          Compute gap = random number
            between 0 and 1
6.          Synthetic[newindex][attr] =
            Sample[i][attr] + gap × dif
7.      End for
8.      newindex++
9.      N = N − 1
10. End while
11. Return //End of Populate.
```

As shown in Fig. 1, for each minority class sample, neighbours from the k nearest neighbours are randomly chosen depending upon the amount of over-sampling required. Then over-sampling is done by taking each minority class sample and producing synthetic samples along the boundary region of the k minority class nearest neighbours.

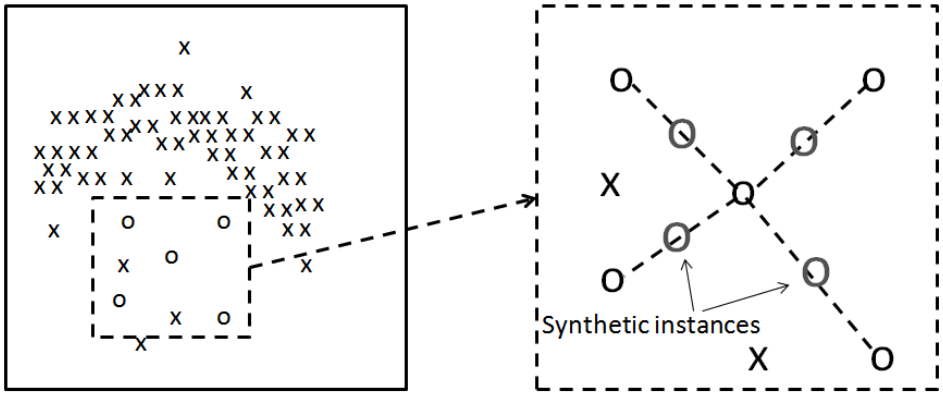

Fig. 1. Generation of Synthetic Instances with the help of SMOTE.

B. Rough Set Theory

The rough set theory is generally applied for data analysis. It was first popularized by Z. Pawlak in 1982 (Pal *et al.*,1999). It is an addendum of approximation technique to the set theory concept in order to handle imprecise and uncertain data.

In RST, the information system (Pal *et al.*,1999) is characterized by $I = (U,A,V,f)$, where U is a non-empty finite set of patterns; A is the non-empty finite set of attributes of each pattern such that $A = (C \cup D)$ where C is a subset of conditional attributes and D is a subset of decision attributes, V is the range of attributes defined as $V_a(x)$: the value of some pattern $x \in U$ corresponding to feature $a \in A$ and f is a generic function defined as $\{f: U \to V_a\}$ for every $a \in A$. The core concepts adopted in RST are defined below:

Equivalence Relation:

$$\{(x,y) \in U^2 | \forall a \in P: a(x) = a(y)\} \quad (1)$$

Lower Approximation:

$$\underline{P}X = \{x \in U | [x]P \subseteq X\} \quad (2)$$

The lower approximation forms the basis of decision making, and may not be equal for two equivalence classes.

Upper Approximation:

$$\overline{P}X = \{x \epsilon U | [x]P \cap X\} \neq \varphi \quad (3)$$

The Lower approximation is always a subset of Upper approximation. The presence of a pattern in the upper approximation is heavily dependent on its absence in any of the lower approximations.

Boundary Region: It determines whether a set is rough or crisp and is characterized by:

$$\overline{P}X - \underline{P}X \quad (4)$$

Absence of any elements in the boundary region implies that the particular set is crisp.

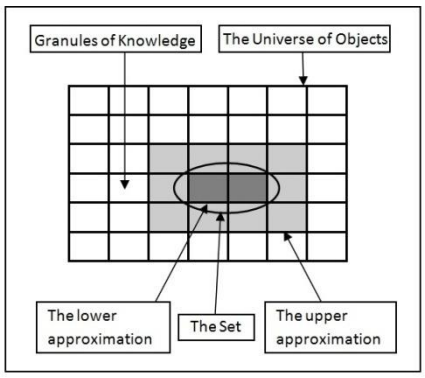

Fig. 2.The simplified concept of the Rough Set Theory.

C. Random Forest

The Random forest was proposed by Breiman (Fawagreh et al., 2014) which is basically a learning method based on aggregation. Random Forest algorithm works as a large collection of decorrelated decision trees. The RF is made up of basically two notions: a subset of samples and then a random subset of features. The RF uses the CART algorithm to build decision trees.

The core concept supporting the RF is the unification of a large number of decision trees composed by utilizing a subset of training samples and then selecting the best feature as a split point among a random subset of features at each split node. Suppose there are m decision trees then the sample data is partitioned into m subsets of samples for training decision tree. The estimation of the best split is done on the basis of the number of randomly chosen features in the dataset which accounts for boosting the performance of the RF. The determinative decision is taken after the majority voting from the constructed trees.

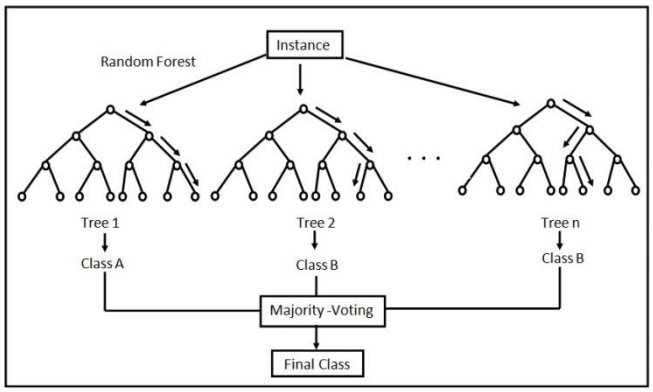

Fig. 3.Simplified Random Forest.

Individual decision trees are generated as described below:

- A dataset [*inbag*] is formed by sampling with replacement members from the training set; this technique is often referred to as "bootstrapping". The number of examples in the [*inbag*] dataset is equal to that of the training dataset. This new dataset may contain duplicate examples from the training set. Using the bootstrapping technique, usually, one-third of the training set data is not present in the [*inbag*]. This left-over data is known as the out-of-bag data [*oob*].
- A random number of attributes are chosen for each tree. These attributes form the nodes and leaves using standard tree-building algorithms.
- Each tree is grown to the fullest extent possible without pruning.

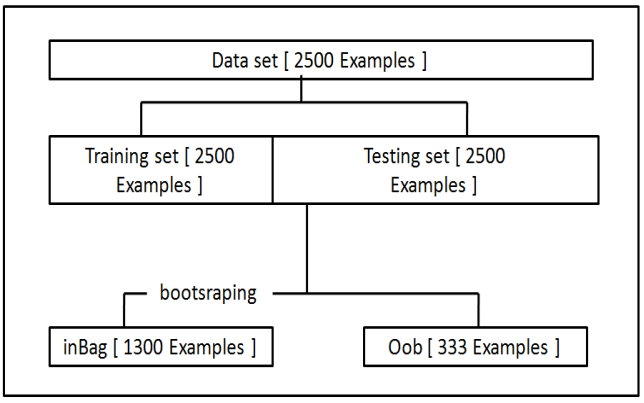

Fig. 4.Sample with Replacing.

The above method is continued several times to generate multiple individual random tree learners. Testing of the respective trees and the entire forest as well is done using the out-of-bag examples. The out-of-bag error estimate gives the average misclassification which is an imporant attribute for performance analysis.

D. Rough Random Forest

The RRF is an ensemble learning model which is constructed using three basic building blocks: RST, Decision Tree and RF. Here, bagging and random feature selection are used to form a number of rough decision trees which is constructed by the overall boundary region approach (Gondane *et al*.,2015). The method of constructing the overall boundary region is given as follows:

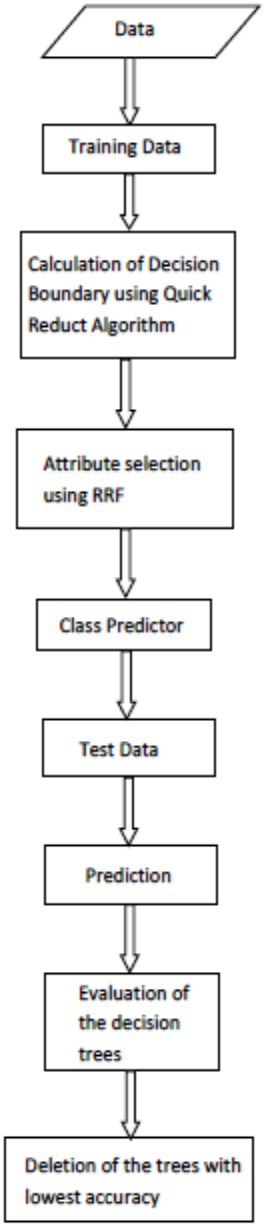

Fig. 5.Modified Rough Random Forest for Big Data.

Let there be n decision classes in the data sample, and B is some conditional attribute ($B \in A$), then its boundary region ($br_B(d = i)$) based on any decision class i can be defined as:

$$br_B(d = i) = \overline{B}(d = i) - j = 1n\underline{B}(d = i), \quad (5)$$

where $\overline{B}(d=i)$ and $\underline{B}(d=i)$ indicates the Lower and Upper approximation of attribute B based on decision class equal to i.The overall boundary region for making a decision of selecting best-split feature unbiased as:

$$\underline{B}(d=i) = i = 1$$

$$BR_B = U - i = 1\, nnbr_B(d=i), \qquad (6)$$

where U: set of all elements; $br_B(d=i)$: boundary region of attribute B based on decision class i; BR_B: overall boundary region of B.

In the RRF, n bootstrap samples are generated from the training data from the n rough decision trees. Then by using the i-th bootstrap sample, each rough decision tree i is trained.

Testing is done by passing the test data through each rough decision tree of the rough random forest. The forest then chooses the classification having the most votes overall the trees in the forest.

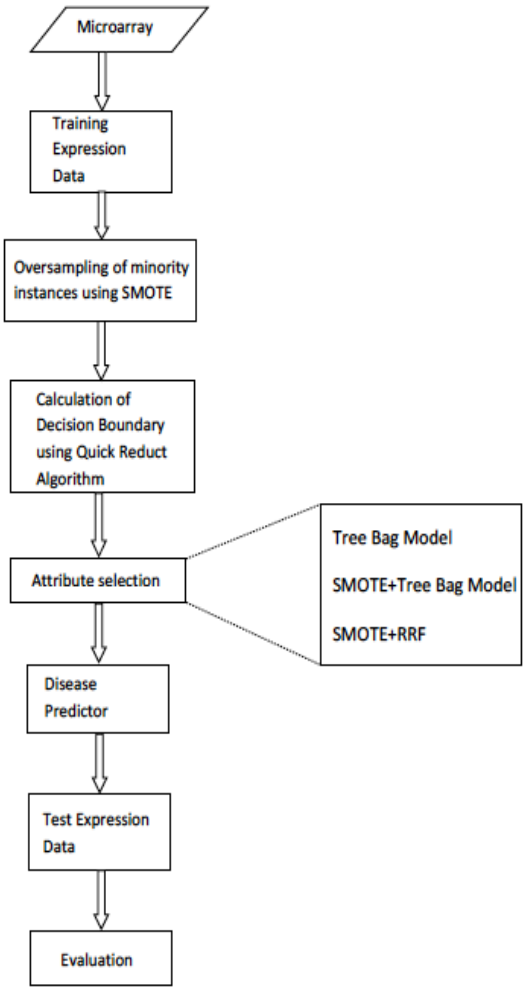

Fig. 6.Minority Class classification system.

E. SMOTE Algorithm for Big Data

The SMOTE Algorithm for streaming Big Data can be implemented using the Sliding Window Approach. In this approach, as the window slides with time, the SMOTE technique gives an output when new data appear on the stream. It emits this output by processing data in the window.

However, such computations are very costly as they require a complete re-computation over the full window. Thus, the scalability of SMOTE can be improved using the technique called Slider. This method skips complete re-computation by re-using previously run sub-computations whenever new data is added at the end of the window or old data is dropped from its beginning (Bhatotia*et al.*,2012). Bhagat *et al.*,in2015, modified the oversampling technique to adapt to big data using MapReduce so that this technique can handle as large data-set as needed.

F. Rough Random Forest for Big Data

In the big data environment, the data update rate is very fast, so are the update rates of data characteristics and modes hidden in data. Decision trees based on training sets data will become out of date and less accurate in classifying data after a certain period of time. Thus, to handle big data, the Rough Random forest procedure needs to be modified so that it can quickly classify new streaming data. It is done by checking the accuracy rate of each tree during the implementation process, regularly updating the forest and eliminating trees with the lowest accuracy rate (Liu, 2014). Also, the number of trees is to be scaled so that while ensuring efficiency, accuracy should not be compromised. The improved Rough Random Forest for Big Data has been shown in Fig. 6

III. EXPERIMENTAL PROCEDURE

In this section, we have investigated the procedures to classify the minority instances of the imbalanced datasets. The datasets used have been enlisted along with their relative distribution of majority-minority instances in Table 1. We have done our experiment in two steps. In the first step, the basic SMOTE algorithm enhanced with RRF has been implemented. In the second step, we have extended the procedure for Big Data environment.

A. Basic SMOTE Enhanced with RRF

The procedure is as shown in Fig. 7. This phase consists of two stages:

Stage 1: SMOTE has been applied to introduce new artificial minority class samples to the training set. The particular idea has been applied as follows: the k-nearest neighbours of sample X in the minority class have been obtained, and then n samples selected randomly and recorded as X_i. Finally, the new sample X_{new} have been defined by interpolation as follows:

$$X_{new} = X_{origin} + rand \times \left(X_i - X_{origin}\right),\ i = 1,2,\ldots,n \qquad (7)$$

Here, while applying SMOTE, the minority class examples have been over-sampled 100% of its original size. Then, the five nearest neighbours have been calculated for generating synthetic examples. The percentage of False Positive (FP) and True positives (TP) have been averaged over five-fold cross-validation runs for each of the data combinations. But one of the flaws of the SMOTE algorithm is that it fails to locate all the boundary points. Also, the k-means algorithm is useful only for spherical datasets, and its execution requires quite an amount of time.

Thus, we need to move on to our next step of experiment containing the RRF construct, which is efficient for large datasets and suitable datasets of any shape. *Stage 2*: Next, the RRF theory has been applied to get the decision boundary using the quick reduct algorithm. In RRF, each decision tree has been trained in the following way. Each bootstrap sample of the dataset has been divided into training and testing decision tables. Using RST, we have used global discernibility method to convert the real-valued attributes to nominal ones in both the training and test decision tables. Then, feature selection has been performed on the training decision table based on the reducts calculated. Using this feature selection method, we have obtained a subset of attributes which gave the same quality as the complete feature set. Subsequently, testing has been done by introducing the test data to the rough random forest and then the final decision class is calculated based on the majority voting from the individual decision trees. Here, we have considered approximately two-thirds of the dataset selected arbitrarily as the training set and used the remaining one-third of the same dataset said to be Out-of-bag (OOB) samples for testing the validity of the trained dataset. Thus, this OOB data has been put for testing each decision tree. The AUCs (Batista *et al*.,2004) have been calculated using the trapezoidal rule. Also, to predict the accuracy of this idea, we have compared this model with other models like the Tree Bag Model and SMOTE+Tree Bag model. The AUCs have been used to check the accuracy of each model.

B. Basic SMOTE Enhanced with Modified RRF for Big Data Environment

Here, we have combined the basic ideas for SMOTE and Rough Random Forest for Big Data. First, we have used the SMOTE to oversample minority instances. The next step involves the modification of RRF for Big Data environment. Here, SMOTE+RRF model is working in two steps: the first step has been used for building of the classification model and the second step has been used for the prediction of the class labels related to the dataset using already generated the model. In the model-building phase, a training dataset has been used for the creation of the model using, at first, the sliding window approach, and then the divide-and-conquer approach. Using the first approach, for each data in the window, SMOTE has been applied. Then, parallel computation has been done by each decision tree of the random forest for the corresponding data block in the window. The aggregation of the entire trees constitutes the forest. After building the model, classification has been implemented to predict the class labels of the test datasets. Here, partition of data has been done, and the data blocks are transferred across all the nodes. Then, the classification for corresponding data blocks has been done based on a voting scheme. In this experimental setup, initially, the entire data in the window has been used. And as the streaming data arrives, new data is added, and old data removed, consequently, the output is updated. Since the dataset in each SMOTE+RRF model represents particularly a limited portion of samples from the whole dataset, a specific number of tress has been selected from the set and appended to the resulting decision tree set. The screening process is done based on the maximum accuracy obtained by the individual decision trees.

IV. RESULTS AND DISCUSSIONS

Comparison of the results has been made based on Tree Bag model, SMOTE and SMOTE+ RRF. Clearly, results obtained by us show remarkable improvement.

A. Datasets Used

Imbalanced Datasets from KEEL Dataset Repository

For performance analysis of the different algorithms used for imbalanced data, we have selected some datasets publicly available from the KEEL Dataset Repository (Alcalá-Fdez *et al*.,2011). To validate the algorithm for varying degree of imbalance, we have divided our datasets into the following scenarios:

- Imbalance ratio between 1.5 and 4.
- Imbalance ratio higher than 20.

The ratio of the majority and minority instances of the various datasets used have been shown in Table 1.

Table-I: The Ratio of the Dataset Distribution.

Dataset	Majority class	Minority class
Wdbc	1.86	1
Transfusion	3.20	1
Hepatitis	3.80	1
Wine-red	29.17	1
Thyroid	36.94	1
Mammography	42.10	1

Stock Market Data

The stock market produces a massive amount of valuable trading data. Here, we have used the S&P 500 market index, which is freely available in the Yahoo finance site (Up, 2012). The class imbalance problem in market data occurs when there is an overload of buy or sell orders for a stock at a particular time. The maximal shift in share price is due to such kind of imbalances, which occurs relatively quickly. These shifts cause a stock's price to either go up or go down, the knowledge of which can be utilised by traders (Duggan, 2017). The structure of the Stock Market Data has been given in Table 2. On a typical trading day, stock information of previous day like the open, high, low, close price of stock along with volume traded has to be analysed which provides a lot of hidden information to predict stock trends. To quickly visualise this information, OHLC (open, high, low, close) charts or Candlestick charts are commonly used. The dataset consists of 70,000 entries until the latest date (2018-10-22). We have used the sliding window approach to include the most recent data to implement the model.

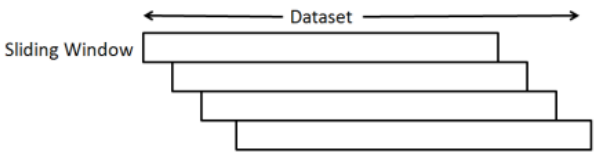

Fig. 7.Sliding Window Approach.

Table-II: Structure of the Stock Market Data.

Date	Open	High	Low	Close	Adj Close	Volume
2017-07-24	2472.04	2473.10	2466.32	2469.91	2469.91	3.01E+09
2017-07-25	2477.88	2481.24	2474.91	2477.13	2477.13	4.11E+09
2017-07-26	2479.97	2481.69	2474.94	2477.83	2477.83	3.56E+09
2017-07-27	2482.76	2484.04	2459.93	2475.42	2475.42	4.00E+09
2017-07-28	2469.12	2473.53	2464.66	2472.10	2472.10	3.29E+09
2017-07-31	2475.94	2477.96	2468.53	2470.30	2470.30	3.47E+09
2017-08-01	2477.10	2478.51	2471.14	2476.35	2476.35	3.46E+09
2017-08-02	2480.38	2480.38	2466.48	2477.57	2477.57	3.48E+09
2017-08-03	2476.03	2476.03	2468.85	2472.16	2472.16	3.65E+09
2017-08-04	2476.88	2480.00	2472.08	2476.83	2476.83	3.24E+09
2017-08-07	2477.14	2480.95	2475.88	2480.91	2480.91	2.93E+09
2017-08-08	2478.35	2490.87	2470.32	2474.92	2474.92	3.34E+09
2017-08-08	2465.35	2474.41	2462.08	2474.02	2474.02	3.31E+09

B. Results Obtained

For performance analysis on the algorithms for the datasets from KEEL Dataset Repository as well as the Stock Market Data, comparisons based on AUC scores obtained using the Tree Bag Model, SMOTE+Tree Bag Model, and SMOTE+RRF has been done.

Qualitative Analysis for Datasets from KEEL Dataset Repository

The AUC values of the six datasets obtained have been shown in Table 3.

The receiver operating characteristic curve or ROC is a standard evaluation metric for binary classification problems. It plots the true positive rate vs the false positive rate with respect to varying thresholds during sample classification. If the classifier is accurate, the true positive rate will increase quickly, and the area under the curve or AUC will be close to 1. On the contrary, if the true positive rate increases linearly with respect false positive rate giving the AUC approximately around 0.5, then it implies it is just a random classifier.

Figures 8-13 show the results obtained from plotting the various ROC curves for the different datasets.

Table-III: Results using Different Techniques (in %age).

Dataset	Tree Bag Model	SMOTE+Tree Bag Model	SMOTE+ RRF
Wdbc	96.94	99.30	99.58
Transfusion	67.84	70.58	76.42
Hepatitis	82.68	84.84	99.85
Wine-red	98.90	99.66	99.78
Thyroid	87.49	98.53	99.71
Mammography	68.52	98.59	99.61

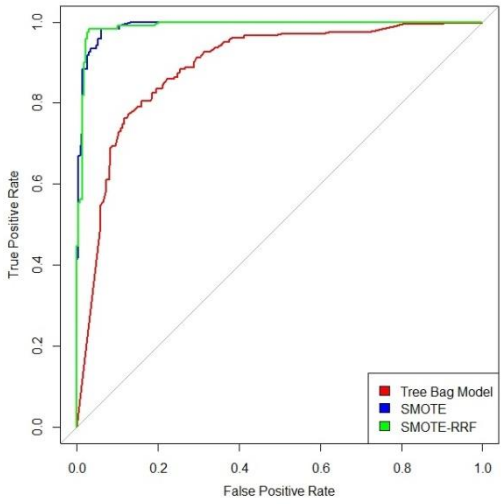

Fig. 8. ROC curves for WDBC Dataset.

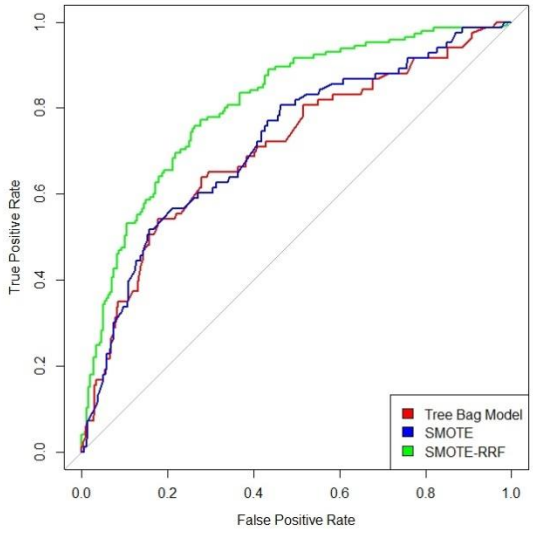

Fig. 9.ROC curves for the Blood Transfusion Dataset.

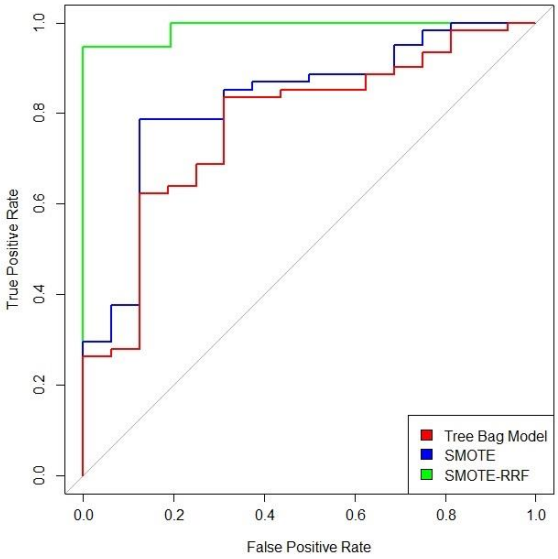

Fig. 10. ROC curves for the Hepatitis Dataset.

Qualitative Analysis for Stock Market Data

We have used SMOTE enhanced with Rough Random Forest to predict stock trends. Oversampling has been performed on the data to 100% of its original size. The training data consists of data from '1970-02-18' to '2001-06-20 '(31 years), and the testing data consists of data from '1987-10-22' to '2018-10-22' (31 years). To predict the price time-series, several indicators,*T*, have been defined in (Jula *et al*.,2016) that mostly aims to obtain some characteristics of the price's series, such as their variability extent or finding out some specific pattern, etc.

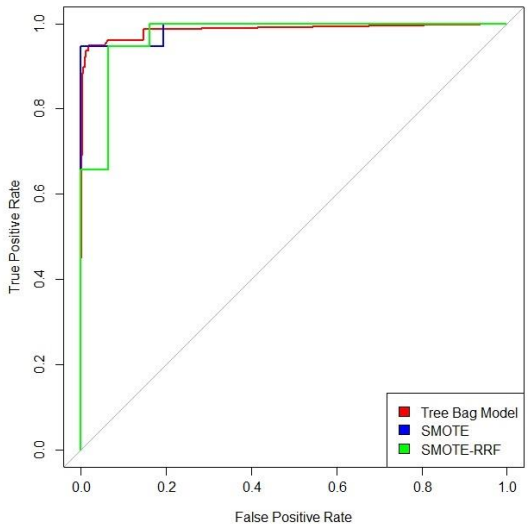

Fig. 11. ROC curves for the Wine Dataset.

Using the concept of SMOTE enhanced with RRF, we have obtained the essential variables from the dataset as shown in Fig. 14. Taking a threshold value of 10, we have selected only a subset of features for stock prediction. Here, the aim is to forecast the correct trading signal at a given time. The predictions of the model, *T* have been translated into trading signals as:

$$signal = \begin{cases} sell if T > 0.1 \\ hold if\ -0.1 \le T \le 0.1 \\ buy if T > 0.1 \end{cases} \quad (8)$$

From this model, T values provides information for making better investments. The signal for a particular day is determined by calculating the T value and using the thresholds in Eqn. (8). In order to evaluate the performance of the model, the model obtained from the training data has been stored in an object 'tradeRecord' and then this object has been used on the test data to obtain results of the trading activity.

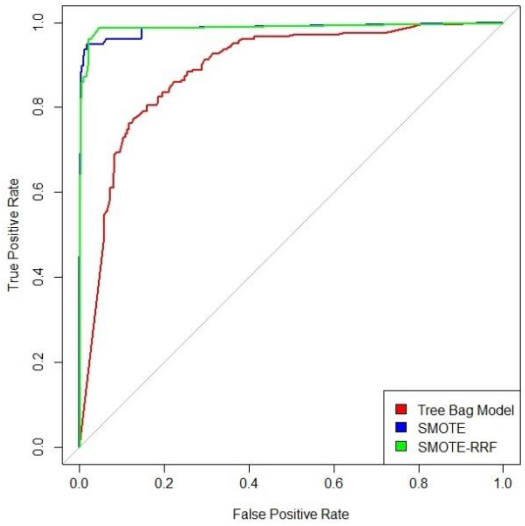

Fig. 12. ROC curves for the Hypothyroid Dataset.

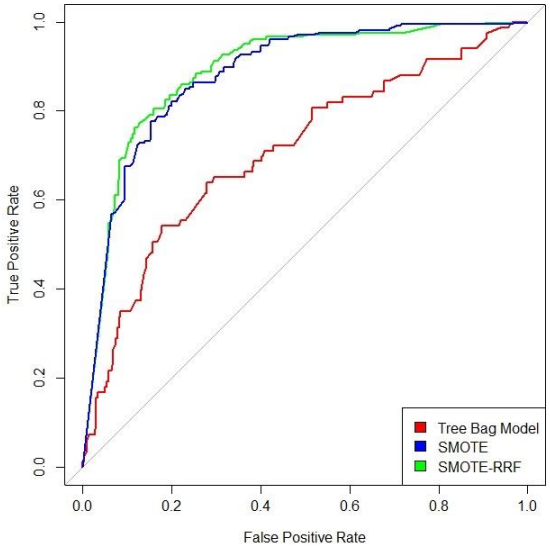

Fig. 13. ROC curves for the Mammography Dataset.

In order to compare the performance of the technique of SMOTE+RRF for this data, we have compared its results with the well-known Tree Bag Model and the SMOTE+Tree Bag Model. The AUC scores for the Tree Bag Model, SMOTE+Tree Bag Model, and SMOTE+RRF respectively are 82.51%, 94.43% and 99.56%.

Fig. 15 shows the trading activity obtained from our results dated from '01-08-2018' to '22-10-2018'.The trading graphs obtained from the results are tested with the graphs available on the financial website "investing.com" (Investing.com, 2018, November 29).

Fig. 16 shows the snapshot of the trading activity obtained from "investing.com" dated from '13-08-2018' to '22-10-2018'. Comparing the two graphs, we can observe that the trading signals obtained from our method quite precisely match with the signals obtained from the stock market website.

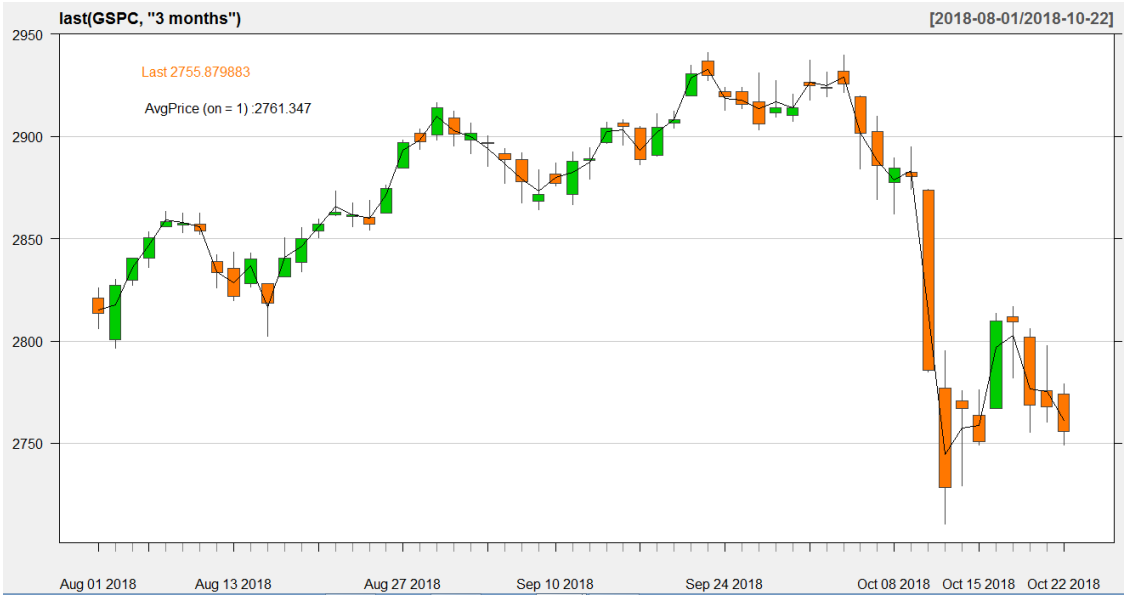

Fig. 14. Trading patterns as obtained from SMOTE improved with RRF.

Fig. 15. Results as verified from www.investing.com.

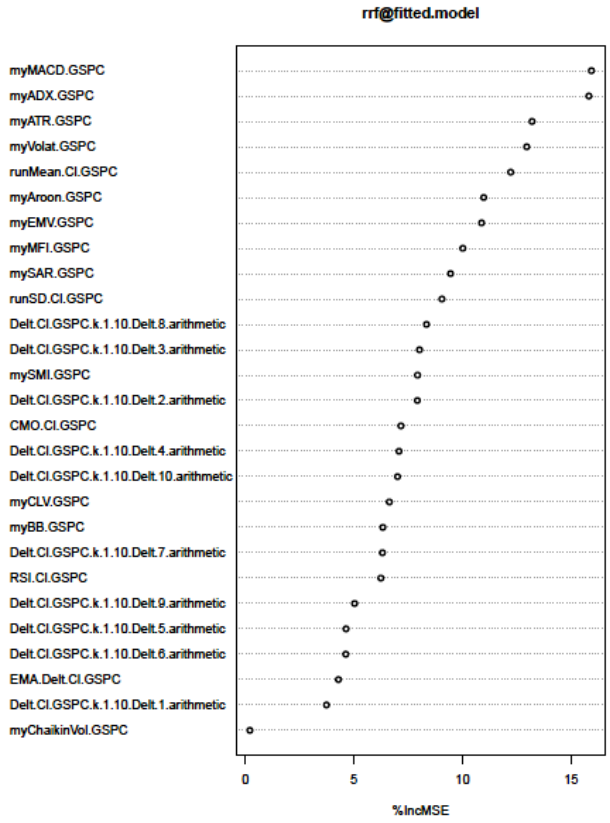

Fig. 16. Variable importance according to SMOTE enhanced with RRF.

V. CONCLUSION

Information extraction from imbalanced datasets is indeed a critical problem in today's age, where nearly all of the datasets are imbalanced. In this work, we have applied the SMOTE algorithm in conjunction with the RRF method for information retrieval. The proposed system has been carried out on a variety of datasets from the KEEL Dataset Repository with varying amount of imbalance. We have compared the validity of the proposed system in terms of the ROC curve and the corresponding AUC scores. Based on the previous techniques available, this method gives improved results, which proves that SMOTE can boost the performance of classifiers in presence of minority class instances. In today's world, big data can change the world as it can synthesise vast amounts of information. In this regard, we have tried to extend our method to a parallel environment on the S&P 500 stock market dataset to predict trading signals. This parallel environment ensures that the challenge of imbalance in the time series forecasting approach is met. The results obtained are excellent as confirmed by the popular website on stock market trading. In the future, we intend to explore whether this method can be adapted to other types of big datasets.

ACKNOWLEDGMENT

The authors would like to acknowledge TEQIP III scheme of the Government of India for providing the required funds for the publication of this work.

REFERENCES

1. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., & Herrera, F.: Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. Journal of Multiple-Valued Logic & Soft Computing. 17, (2011).
2. Barua S. *et al*.: MWMOTE--majority weighted minority oversampling technique for imbalanced dataset learning. IEEE Transactions on Knowledge and Data Engineering.26, 2, 405-425, (2014).
3. Batista, G. E., Prati, R. C., & Monard, M. C.: A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD ExplorationsNewsletter.*6*(1), 20-29, (2004).
4. Bhagat, R. C., & Patil, S. S.: Enhanced SMOTE algorithm for classification of imbalanced big-data using random forest. Advance Computing Conference (IACC),2015 IEEE International. 403-408, (2015).
5. Bhatotia, P., Dischinger, M., Rodrigues, R., & Acar, U. A.: Slider: Incremental sliding-window computations for large-scale data analysis. CITI, Universidade Nova de Lisboa, Lisbon. (2012).
6. Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C.: Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. Pacific-Asia conference on knowledge discovery and data mining, Springer, Berlin, Heidelberg. 475-482, (2009).
7. Chawla, N. V.: Data mining for imbalanced datasets: An overview. Data mining and knowledge discovery handbook, Springer, Boston, MA. 875-886, (2009).
8. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P.: SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research.16, 321-357, (2002).
9. Chennuru, V. K., & Timmapparedy, S. R.: MahalCUSFilter: A Hybrid Undersampling Method to Improve the Minority Classification Rate of Imbalanced Datasets. In International Conference on Mining Intelligence and Knowledge Exploration (pp. 43-53). Springer, Cham, 2017.

10. Dietterich, T. G.: An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine learning. 40(2), 139-157, (2000).
11. Duggan W.: How to trade opening and closing imbalances (2017, March 29), https://www.lightspeed.com/active-trading-blog/trade-opening-closing-imbalances/
12. Fawagreh, K., Gaber, M. M., & Elyan, E.: Random forests: from early developments to recent advancements. Systems Science & Control Engineering: An Open Access Journal, 2(1), 602-609, (2014).
13. Gondane, R., & Devi, V. S.: Classification using rough random forest. International Conference on Mining Intelligence and Knowledge Exploration, Springer, Cham. 70-80, (2015).
14. Ha, T. M., & Bunke, H.: Off-line, handwritten numeral recognition by perturbation method. IEEE Transactions on Pattern Analysis & Machine Intelligence. 5, 535-539, (1997).
15. Han, H., Wang, W. Y., & Mao, B. H.: Borderline-SMOTE: a new over-sampling method in imbalanced datasets learning. International Conference on Intelligent Computing, Springer, Berlin, Heidelberg. 878-887, (2005).
16. Investing.com (2018, November 29) https://www.investing.com/
17. Jula N. M. & Jula N.: Using R for analysing Financial Markets. Challenges of the Knowledge Society, 990, (2016). http://cks.univnt.ro/uploads/cks_2016_articles/index.php?dir=11_IT_in_social_sciences%2F
18. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence, 5(4), 221-232, (2016).
19. Kibler, D., & Aha, D. W.: Learning representative exemplars of concepts: An initial case study. Proceedings of the fourth international workshop on Machine Learning.24-30, (1987).
20. Liu, Y.: Random forest algorithm in big data environment. Computer Modelling & New Technologies, 18(12A), 147-151, (2014).
21. Pal, S. K., &Skowron, A.: Rough-fuzzy hybridization: A new trend in decision making. Springer-Verlag New York, Inc. (1999).
22. Provost, F.: Machine learning from imbalanced data sets 101. Proceedings of the AAAI'2000 workshop on imbalanced data sets. 1-3, (2000).
23. Ramentol, E., Caballero, Y., Bello, R., & Herrera, F.: SMOTE-RSB*: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory. Knowledge and information systems.33(2), 245-265, (2012a).
24. Ramentol, E., Verbiest, N., Bello, R., Caballero, Y., Cornelis, C., & Herrera, F.: SMOTE-FRST: a new resampling method using fuzzy rough set theory. Uncertainty Modeling in Knowledge Engineering and Decision Making. 800-805, (2012b).
25. Up, A.N.R.C.: Yahoo! Finance. (2012).Yahoo! Finance, (2012).
26. Wei, J., Huang, D., Wang, S., & Ma, Z.: Rough set based decision tree. Intelligent Control and Automation, 2002. Proceedings of the 4th World Congress on IEEE. 1, 426-431, 2002.

AUTHORS PROFILE

Tanuja Das received the B.Tech degree in Information Technology from North Eastern Hill University Shillong in 2012 and the M.Tech degree in Information Technology from Tezpur University in 2014. She worked as a Visiting Scientist in the Machine Intelligence Unit of ISI, Kolkata for a period of three months from 1st Jan 2015 to 31st March 2015. Currently, she is pursuing her PhD degree at the Department of Information Technology, North Eastern Hill University Shillong. She is currently working as an Assistant Professor in the Department of Information Technology, Gauhati University Institute of Science and Technology, Guwahati, Assam, India. Her research interests include Bioinformatics and Big Data.

Abhinandan Khan received the B.Tech degree in Electronics and Communication Engineering from the West Bengal University of Technology, India in 2011, and the ME degree in Electronics and Telecommunication Engineering from Jadavpur University, India in 2013. He is currently pursuing Ph.D. at the Department of Computer Science and Engineering, University of Calcutta, India. Mr Khan received the University Gold Medal for securing the highest marks among all post-graduate engineering courses at Jadavpur University. He is also a recipient of the Senior Research Fellowship (NET) from the Council of Scientific & Industrial Research (CSIR), Government of India. Mr Khan's research interests include computational biology, bioinformatics, computational intelligence, etc. Mr Khan has published 16 research articles.

Goutam Saha received the BE degree in Electrical and the ME degree in Electronics and Tele-Communication from Calcutta University in 1984 and 1989, respectively, and the PhD degree from IIT-Kharagpur in 2000. He was also a Post-Doctoral Research Associate in IIT-Kharagpur in 2001 and a Post-Doctoral Research Fellow in Ben-Gurion University, Israel in 2002.Currently, he is a Professor in the Department of Information Technology and holds the chair of Dean, School of Technology, North-Eastern Hill University, Shillong. His research interests include Embedded Systems, System Biology, Bioprocess Controller Design, Internet of Things, Computational Biology and Bioinformatics, etc. He has published more than 30 research papers in international journals and conferences, 3 book chapters and 1 book of international repute. He has also claimed a patent. He is a member of the Sixth Reconstituted Task Force Committee on Bioinformatics, Computational and System Biology (BCSB) of the Department of BioTechnology (DBT), Government of India.