International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Optimized Support Vector Machine for Big Data Classification NPC

[1*]Dr. R. Josphineleela, [2]Dr. Mudassir Khan, [3]Dr. R. Baskaran, [4]Dr. Syed Mohd Fazal Ul Haque, [5]Amjan Shaik , [6]Dr. Siva Shankar S.

Submitted: 12/02/2023 **Revised**: 13/04/2023 **Accepted**: 04/05/2023

Abstract: The rapid advancement of contemporary technology and smart systems has resulted in a large influx of big data. A phenomenon known as the class imbalance problem limits learning from many real-world datasets. When one class (the majority class) contains disproportionately more instances than the other class, the dataset is unbalanced (the minority class). Because of these datasets, traditional machine learning algorithms struggle to perform well on classification tasks. To compensate for the imbalance, NPC employs an innovative hybrid machine learning approach for grading the training samples.Both local and global data are used to generate the grades. The contribution of this article is a totally new classifier for efficiently dealing with the imbalance issue without the requirement for manually-set parameters or expert knowledge. To address this problem, in this research a novel approach Hybrid Support Vector Machine is designed by incorporating three major steps like pre-processing, dimension reduction and classification. Initially, the pre-processing phase is enabled by the data normalization process. The extensive sets of features are reduced using dimension reduction process and are achieved by using Quantum Theory-based Particle Swarm Optimization (QPSO). With this technique, a better solution can be obtained for classifying the big data; therefore, the existing problems related to accuracy metrics can resolved. Finally, a hybrid optimized support vector machine technique is proposed to accomplish the big data classification task. The suggested technique is compared to sample algorithms on unbalanced datasets in order to demonstrate the algorithm's efficacy.

Index terms: *Neighbours Progressive Competition, Big Data, Support Vector Machine, Quantum particle swarm optimization, Feature selection.*

1. Introduction

Recent technical advancements have paved the way for the use of massive amounts of data in real-world applications [1]. Because of this expansion, structured knowledge cannot be easily retrieved, nor can it be quickly identified or extracted. These discoveries have resulted in the advancement of data mining and data science as a recognised subject, which has proven to be critical in today's data-driven world [2]. The amount of information that must be handle in the actual world currently surpasses the processing power of conventional systems, and this is true of data mining as well [3]. The most important cause for an increased velocity of data on the Internet is the emergence of fresh technology, additional services such to Cloud Computing (CC) [4]. This incident indicates a significant problem for the data analytics community. As a result, big data are employed which has been well-known in recent years for delivering a wealth of data and prospects for improving decision-support

1 Department of cse, professor ,University:Panimalar engineering college.*
2Assistant Professor, Department of Computer Science, College of Science & Arts Tanumah,
King Khalid University, Abha, Saudi Arabia.
3Dean [R & D],Centre for Research & Development, Agni college of Technology,
Chennai -130.
4. Assistant Professor, Department of Polytechnic, Computer Science and Engineering.
Maulana Azad National Urdu(A Central University), Gachibowli, Hyderabad -32, Telangana.
5Professor & HoD-CSE, St. Peter's Engineering College, Hyderabad, TS, INDIA.
6Associate Professor & Dean, Foreign Affairs, Department of CSE, KG Reddy College of Engineering and Technology, Chilukuru Village, Hyderabad, Telangana.
Corresponding Authors Mail: josphineleelar@gmail.com

systems [5]. It supports higher volume, velocity, diversity of data [6, 30]. Its new higher-performance processing proficiencies enable the system to make decisions better and achieves process optimization. Big data is the term for this type of information where in some circumstances, as the pace, variety, and amount of data rise, the existing approaches may be unable to manage data storage and processing [7]. It also surpasses a system's ability to process it in terms of time and memory use. Big data is described by three data characteristics: velocity, diversity, and volume [8, 26]. Velocity denotes the rate at which data is processed and generated based on the applications that are required, whereas variety denotes the nature and forms of data [23]. On the other hand, volume measures the data's potential worth and it is the magnitude of the data [9]. Big data has been used in a variety of industries, including manufacturing, medical, and finance [27]. Moreover, it poses difficulties for digital world in terms of storing, transferring, processing, extracting, and delivering data. This tremendous expansion of data creates various issues in data processing rather than data storage and access [10].

The main difficulty in large data classification is creating a competent parallel classification algorithm that quickly and accurately determines the right hyper-parameters and feature subsets for learning [24]. Various large datasets with misleading and duplicate features were discovered during a comprehensive search in the solution space, generating confusion in the classification process as well as a protracted processing time. [31]. In big data categorization area, the researchers have made multiple attempts to develop effective solutions to this challenge. [11], [12]. Machine learning techniques are extensively used classifiers that have shown to be effective in a variety of applications, including security assessment, health care, and traffic congestion. The popularity of machine learning approaches stems mostly from the ease with which they may be learned [13]. SVM is a prominent classification approach, which has earned more success rate in the industrial big data field among a variety of machine learning approaches. [14, 32]. Despite the fact that the SVM produces promising results, building a competent SVM classifier that can properly forecast hiddenrecent samples remains a difficulty [25]. The ability of SVM to generalise is largely determined by feature selection and optimization of parameter. However, the parameter selection has become a significant study topic in SVMs since it has an impact on the SVM performance [28]. Two SVM parameters including the kernel and the error kernel parameters are needs to be adjusted for better performance [15]. As a consequence, an effective approach for building SVM classifiers integrated with the parallel hyper-parameter optimization and feature subset selection has been devised, allowing the parallel classification method to be employed in a range of practical applications with good results [29]. To resolve the problems in big data categorization, an innovative optimised SVM design was developed in this study.

The manuscript is organised as follows; section 2 is dedicated to the literature review. The third section outlines the research gap discovered in the previous study. The suggested technique is detailed in Section 4, and the results of the study and discussion are detailed in Section 5. The conclusion and future scope are depicted in the final part.

2. Literature Review

Various articles that are related to the topic of big data classification model in several fields and its overall outcome are discussed in this section.

Mahdi Saki et al. [16] introduced an edge processing unit in 2019 that involves both data categorization framework and data transfer unit. Data from the Internet of Things (IoT) is divided into critical and non-critical maintenance categories via a data categorization methodology. In addition, the data transmission units are used for data transfer communication.

Congshi Jiang and Yihong Li [17] designed a Radial Basis Function (RBF) neural system classification solution to address health large data classification difficulties in 2019. The dataset was processed using the manifold analysis technique. The similarity matrix and AP clustering were used to build the RBF neural network classifier.

Ismail Hababehet al. [18] developed an integrated strategy for large data categorization prior to duplication, analysis, and data mobility in 2019. The need of protecting big data mobility was demonstrated by categorising data into two types based on the risk effect of the content.

In 2020, Jesus Mailloet al. [19] proposed two big data parameters for solving issues in big data categorization: neighbourhood Density and decision tree progression. The huge data was managed in this case by activating the basic metrics. The presented big data parameters are available as a Spark-Package, enable for a rapid analysis of the structure of a classification dataset prior to using big data pre-processing, which is important for smart data.

Zhenchao Ma et al. [20] developed a generic multiple classification algorithm for heterogeneous and high-dimensional data using a Support Multi-mode Tensor Machine (SMTM) algorithm in 2021. To broaden the formulation of the STM, this method was implemented by using the multi-mode product.

Minjun Zhu and Qinghua Chen [21] introduced a semi-supervised network for big data picture categorization based on the neighbour structure learning framework in 2020. While learning the network representation, this model will include data from the node's neighbours. Structural Labeled Locally Deep Nonlinear Embedding (SLLDNE), on the other hand, combines the network structure, labels, and node properties into a deep neural network.

Long-Hao Yang et al. [22] proposed Apache Spark, a cluster computing system to help with large data categorization, in 2018. The Micro-EBRBS parallel rule generation and inference methodologies for big data multiclass classification issues were used to develop the described solution.

Table 1: Analysis of existing studies

Authors [citations]	**Year**	**Methodology**	**Challenges**
Mahdi Saki *et al.,* [16]	2019	Edge Processing Unit	High computational burden.
Congshi Jiang and Yihong Li [17]	2019	A neural network classification technique based on the RBF.	During classification, each node in the hidden layer calculates the RBF function for the input sample vector, RBF networks take longer to train than Multilayer Perceptron.
Ismail Hababehet al., [18]	2019	Integrated Methodology	This method is unable to handle file characteristics for large data formats like txt, csv, log, xls, and sql files.
Jesus Mailloet al., [19]	2020	Decision Tree Progression and Neighbourhood Density	Other classifier families are affected by the decrease of the dataset.
Zhenchao Ma *et al.,* [20]	2020	Support for the SMTM (Multi-mode Tensor Machine) algorithm	This method has a long learning curve.

Minjun Zhu and Qinghua Chen [21]	2020	Deep Nonlinear Embedding with Structural Labeling (SLLDNE)	Because the majority of nodes are only connected by a few edges, learning effective vector representations with limited data is difficult.
Long-Hao Yang *et al.,* [22]	2018	Apache Spark	This strategy does not address the practical issue of data that is ambiguous and imbalanced.

The above section describes numerous approaches and frameworks utilized for big data classification in various field. The above discussed approaches are executed efficiently for big data classification. Whereas, there are some research gaps that are represented as future work. The proposed technique in [18] efficiently classifies enormous data, however it is unable to handle file properties for such massive data types. Although the Neighbourhood Density and Decision Tree Progression techniques in [19] did well, the smaller dataset had an impact on other classifier families. The Support Multi-mode Tensor Machine (SMTM) technique in [20] takes a long time to train. The superior results are revealed by the Structural Labelled Locally Deep Nonlinear Embedding (SLLDNE) in [21]. Even still, finding adequate vector representations with little information is challenging since most nodes are only connected by a few edges. In [22] the Apache Spark outperformed well still, This strategy does not address the practical issue of data that is ambiguous and imbalanced.Therefore, the aforementioned gaps are rectified by developing novel technique for classify the big data in this work.

3. Research Gap

Big Data deals with massive volumes of data have opened up tremendous opportunities for knowledge discovery. A common challenge in real-world datasets is the class imbalance problem, which limits the amount of knowledge that can be recovered. Classifiers face difficulty in dealing with imbalance data class. Moreover, the "minority class" is the one with the fewest samples, while the "majority class" is the one with the most samples. In traditional machine learning research, balanced data sets are those in which the quantity of distinct class samples is roughly equal. As a result, when presented with unbalanced data sets, the classifier fails to learn the minority class characteristics well, resulting in many minority samples being misclassified. Many scholars have been working on finding a suitable solution to the challenge of unbalanced data categorization throughout the years. Resampling the training samples in order to produce a more balanced dataset is how data level techniques function. This is accomplished by either over-sampling minority class data, or under-sampling majority class samples, using hybrid models that combine over-sampling and under-sampling methodologies. For dealing with the problem of class imbalance, preprocessing approaches are utilized. As a result, one unavoidable downside of this strategy is that it alters the distribution of data in the first place. Furthermore, under-sampling can result in the loss of crucial information from the majority class, whereas over-sampling raises the complexity of the training computation and, in some situations, raises the risk of overfitting. Many of the algorithms discussed above rely on global knowledge to make more accurate conclusions; nevertheless, their learning models are sometimes overly complicated. Furthermore, several of them need the tuning of a hugequantity of parameters, creating them computationally costly and time intensive. The chief contribution of the proposed stud is illustrated as follows,

- The suggested method's major goal is to forecast whether medical data is normal or unhealthy based on this concern.

- Pre-processing, size reduction, and classification are the three steps of the novel technique. Prior to further processing, the details are pre-processed.
- The data is submitted to a dimension reduction technique after preliminary processing to decrease the maximum size data to low-size data.
- The suggested solution employs the QPSO method to minimise data dimension.
- The selected characteristics are then fed into the HSVNN separator input to divide the data into normal and ill categories.

4. Research Methodology

Data mining technique of classification entails obtaining a general rule or classification procedure from a group of study instances. The phrase "big data classification" indicates the method of extracting variables from massive datasets in order to improve health-care quality. Many of the characteristics in conventional medical databases are acquired for purposes other than data categorization, and medical data sets are generally considered as having a large number of dimensions. Even if a medical diagnosis is available, it is preferable to pick low-cost, low-risk clinical studies that are critical in establishing the stage of the disease. Figure 1 shows a detailed representation of the proposed path.

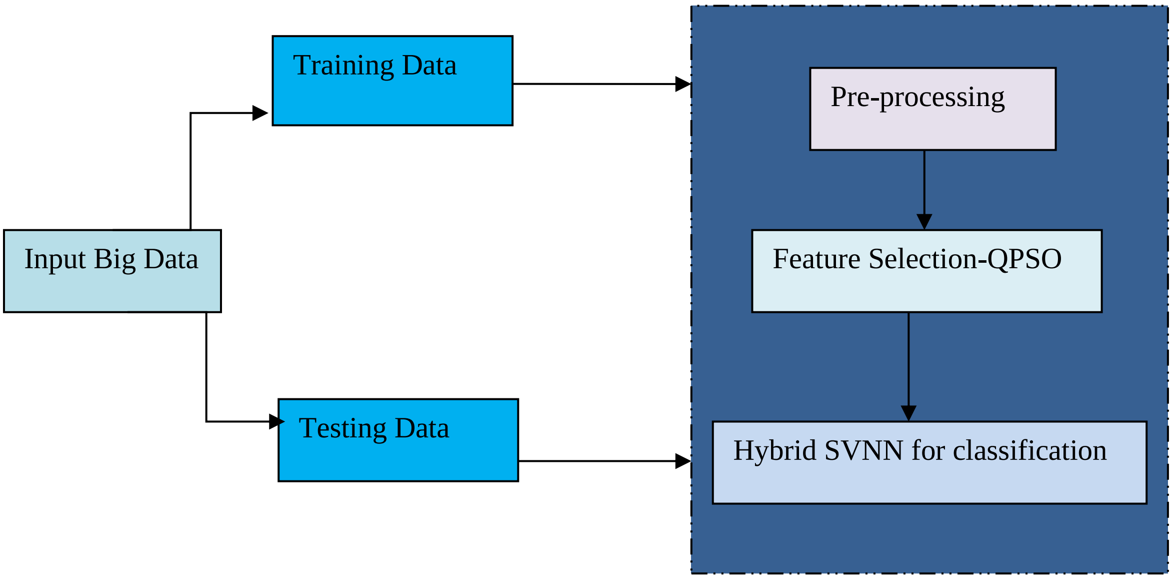

Fig 1: Overall structure of proposed data classification

4.1 Pre-processing

In this work, multiple forms of contaminated databases are used to classify medical data. The number of records in a database is n, and each record contains m characteristics. The database records in this scenario may be incomplete, noisy, or duplicate. This will have an impact on the sections' correctness. As a result, the input information details are pre-processed before beginning the categorization process. Consider dataset D, which contains the number of electronic records as well as the number of m feature sets. The input features' values are first transformed to numeric values. The numerical values are then normalised. $A^{ij} = \frac{A^{ij}-M^{i}}{(max-min)featurej}$ is the mathematical expression for the normalising function.

Where;

A^{ij} → i^{th} row j^{th} column attribute value

M^{i} → Mean value of the i^{th} column attribute

After normalisation, pre-processed data is provided to the parameter reduction process.

4.2 Dimension reduction using Quantum PSO

In this phase, the two significant processes are dimension reduction and feature selection process which is utilized for fetching only the essential data from the dataset. The initial process is executed by adopting the dimension reduction process with the utilization of Principle Component Analysis (PCA). For extracting effective set of features from high dimension feature vector, dimensionality reduction approach is utilized with the help of PCA.With this technique, the low dimensional feature component from the dataset is extracted from the massive dataset for further process.Thus, error accomplished while inducing higher dimension feature space to classifier are reduced.

Then the feature selection process is seen as a combinatorial enhancement issue that points to find optimal functions for a subset of the original data set that still reliably describes the original data. The whole feature representation process consists of two main stages: (i) determining the minimum reductions and (ii) evaluating the selection of traits. This makes them quicker than machine learning techniques that assess subsets of functions through a learning algorithm validation system. This article has utilized an improved PSO procedure to improve the presentation of traditional PSO. For an effective capacity choice cycle, this article utilized a PSO calculation dependent on the quantum hypothesis. Here this technique incorporated with the itemized data about PSO and QPSO cognizance calculations.

4.2.1 Mathematical modelling of PSO algorithm

The pre-processed data parameter is exposed to the reduction procedure after the initial phase of pre-processing. A feature selection approach is utilised to minimise the size. Feature selection is the process of selecting the appropriate characters to reduce the size of the character and construct an efficient prediction system. Similarly, a slew of criteria pose as substantial roadblocks to categorization. As a result, feature selection is a critical step in the classification process. PSO techniqueis an effective tool for managing optimization problem and is an appropriate model for feature selection approach. The calculation is an arbitrary pursuit and equal optimization calculation, which is straightforward, simple to actualize, and exceptionally proficient, and requires less boundary change. This calculation is reasonable for logical exploration as well as for designed application. It begins from stochastic arrangement with looks for ideal arrangement repeatedly; what's more, it assesses nature of arrangement thinking about wellness.

The calculation begins from an arbitrary arrangement, looks through the ideal arrangement iteratively, and assesses the nature of the arrangement by thinking about the wellness. Assume that a measurement space comprises a populace for particles. The area of particle I in an $n-$dimensional space is X. Particle speed V. In the functional cycle, particle speed and position can be reprednted as follows:

$$\left\{\begin{array}{c} V_{id}^{k} = wV_{id}^{k} + c_1 r_1 \left(P_{id}^{k} - X_{id}^{k}\right) + c_2 r_2 (P_{gd}^{k} - X_{id}^{k}) \\ X_{id}^{k+1} = X_{id}^{k} + V_{id}^{k+1} \end{array}\right. \qquad (1)$$

w is inertial weight, $d = 1, 2, \cdots$; k is refered as number of emphases; c term defines quickening elements, r are irregular number going on [0,1]. Equation (1) demonstrates the particle speed was refreshed the three sections: initially, the segment is memory of previousmovement of the particles, which is called the "inertial" part; latter segment is differs from present state of particle to ideal state throughwhih it has passed, it may be considered "self-accepting"; then the third part is separate the current state of particle and best position in the assembly, it can be considered as "social experience";. The cycle stream of PSO calculation as per the following statements:

1) Induce particle swarm which includesspeed of each particle and irregular position.

2) To calculate the fitness value for every particle in each cycle.

3) Update pi and PG as indicated by fitness to update position and speed of the particles.

4) Decide if most extreme number of cycles or worldwide ideal position was reached, whether fulfilled the minimum limit.

If satisfied, complete the iteration; otherwise repeat the steps of 2–4.

4.2.2. Quantum theory-based PSO

To select the optimal set of feature space, a novel Quantum Theory based PSO technique is utilized. The reason behind improving the defaultPSO is that to improve the performance of low convergence rate. And the application of quantum theory to identify particle behaviour is used to develop PSOs by increasing performance analysis of classic PSOs. This is because of the way that particle collection in quantum space is depicted by a bound state made by a potential fascination field in particle sports focus. The particlesquantum states can some likelihood be situated anytime in the arrangement space. Particles with an aggregated state can look for the solution in all possible space, but do not deviate endlessly. In the quantum PSO model, the condition of particle canportrayedthrough the wave work Ψ (k, t) rather than speed and position. The dynamic behavior of particles is different from the behavior of prototype PSO particles. In this specific circumstance, the likelihood of a particle showing up at position X is dictated by the likelihood thickness work Ψ (k, t) ^ 2, the state in which relies upon expected field in the particle $\Psi(x,t)^2$, Solution encoding is a prominent procedure for all optimization algorithms. The QPSO algorithm arbitrarily initializes the values with the N solution, which signifies the **N** particle in the entire population. The initial solution is given in the format table 2.

	F₁	**F₂**	**F₃**		**F_V**
S_1	1	0	1	……	1
S₂	1	1	0	……	1
S_3	0	1	0	…..	0
S_5	1	1	1	……	1

Table 2: Sample solution encoding

The particles travel according to the subsequent iterative equations

$$X_i(t+1) = g_i(t) + \propto . |m_{i.best} - X_i(t)| . ln\frac{1}{u_i(t)}, \; ifs \geq 0.5 \quad (2)$$

$$X_i(t+1) = g_i(t) + \propto . |m_{i.best} - X_i(t)| . ln\frac{1}{u_i(t)}, \; ifs < 0.5 \quad (3)$$

where u_i , s is arbitrary numbers evenly dispersed in intermission [0,1]. The parameter ∝isknown as compression-expansion ratio. Also

$$g_i(t) = \varphi_i(t) . G_{i.best}(t) + \left(1 - \varphi_i(t)\right) . G_{global}(t) \quad (4)$$

$$m_{i.best} = \frac{1}{Q}\sum_{i=1}^{Q} G_{i.best}\,(t) \quad (5)$$

Where φ_i is an equally shared random number in [0, 1] interval, Q indicates particles count, and m_ (i.best) was determined bycenter of best locations for all particles in population.

The achievement QPSO model is when absolute deviation among C (t + 1) and C (t) is 10 times slower when the algorithm is closed; Otherwise, G_mak reaches highest amount of iterations. Where δ is the learning threshold.

Pseudocode of the proposed Feature selection process

```
Initialize the positions and the $\boldsymbol{G_{t.best}}$positions of all the particles
```

```
Do Calculate $\boldsymbol{m}_{\boldsymbol{t.best}}$ using Equation (5) for particle i, where i = 1, 2. . . Q
Select a suitable value ∝
For particles i = 1 to Q
Calculate the cost value of particle i according to Equation (3)
Update $\boldsymbol{G}_{\boldsymbol{t.best}}$using Equation (4)
Update $\boldsymbol{G}_{\boldsymbol{global}}$as follows
If $\boldsymbol{C}(\boldsymbol{G}_{\boldsymbol{i.best}}(\boldsymbol{t}) < \boldsymbol{C}(\boldsymbol{G}_{\boldsymbol{global}}(\boldsymbol{t}-\boldsymbol{1}))$ then $\boldsymbol{G}_{\boldsymbol{global}}(\boldsymbol{t}) = \boldsymbol{G}_{\boldsymbol{i.best}}(\boldsymbol{t})$else$\boldsymbol{G}_{\boldsymbol{global}}(\boldsymbol{t}) = \boldsymbol{G}_{\boldsymbol{i.best}}(\boldsymbol{t}-\boldsymbol{1})$
For dimension 1 to d
$\boldsymbol{\varphi}$= rand (0,1)
u = rand (0,1)
If s = rand (0,1) ≥ 0.5
Update particle positions using Equation (2)
else
Update particle positions using Equation (3)
Until the terminal condition is met
```

To evaluate the best answer, the fitness value is calculated for initial solutions. Maximum fitness is seen to be a good therapy. The fitness function is calculated using the equation below.

$$Fitness = \max(Accuracy) \tag{6}$$

$$Accuracy = \frac{tp+tn}{tp+tn+fp+fn} \tag{7}$$

The technique is completed after an optimum objective function is established. The best fitness solution has been picked for further processing. The provided feature is sent into the HSVNN classifier.

4.3. Classification using HSVNN classifier

After the features have been chosen, the prediction process is provided the important features that have been chosen. This research recommends the HSVNN classifier for predicting. HSVNN is a classifier that combines SVM and ANN. The HSVNN classifier's operating concept is explained in detail.

4.3.1 Artificial neural network

A computer structure inspired by biological neuron features in the human brain is an ANN. The number of nodes connects each of the three levels that make up NN. Each neuron applies an "activation function" to its net contribution while computing its NN outcome signal. The ANN model constitutes three different layers namely, the input layer, a hidden layer, and an output layer. Data, signals, characteristics, or estimates from external sources are received via the input layer. These data sources (tests or models) are usually integrated inside the limitations of the activation functions (AF). The centre layer defines the concealed layer between the input and output layers. From this level, information is delivered to the output level. The neurons in the output layer are still functioning normally. The network's ultimate output is determined by this layer, which is made up of neurons. The structural representation of NN is shown in Figure 3.

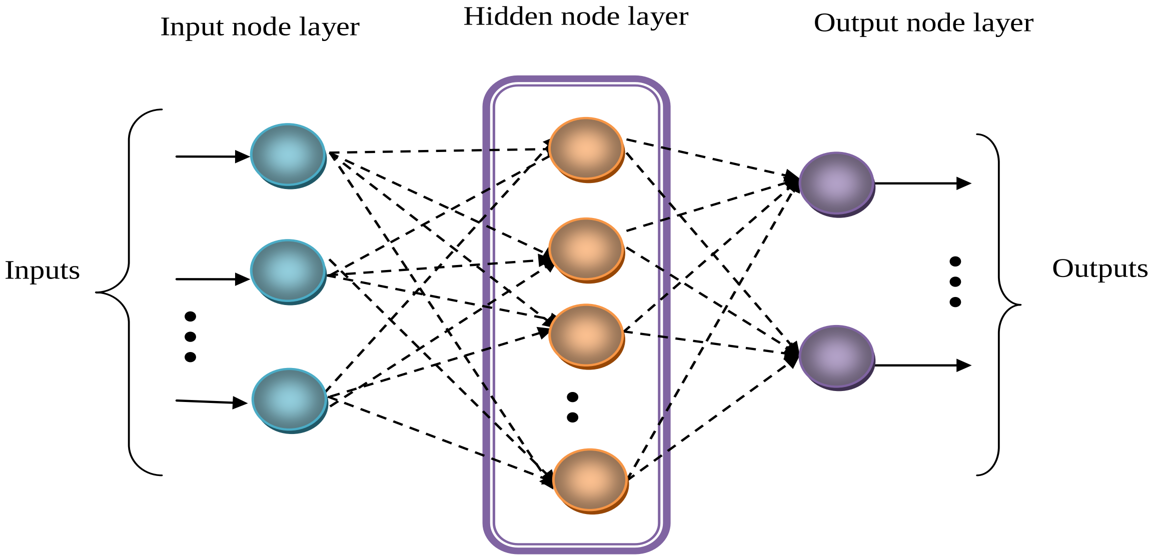

Fig 2: Formation of Neural Network

Consider the symbol $a_1, a_2, ..., a_u$ as a vector of input nodes, the hidden layer nodes as $b_1, b_2, ..., b_v$ a vector, and the output layer nodes as $c_1, c_2, ..., c_w$ a vector. The weight among hidden and input layers is written as W_{ij}^h; moreover, the weight amid the hidden and output layers is expressed as W_{jk}^o. Table 1 shows the back-propagation neural network learning algorithm.

```
Samples, learning rate $\eta$, goal value O, and sigmoid activation function are all inputs.
Output: Structure that has been trained
Begin
Initialize the weight values arbitrarily as $W_{ij}^h$, $W_{jk}^o$, for iteration =1,2,....,iteration$_{max}$.
Do the following for sample =1,2,...,N: / feed forward stage:
        For $j = 1,2 ..., v$ do
    $h_j = f\left(\sum_{i=1}^{u} a_i * W_{ij}^h\right)$
        For $k = 1,2 ..., w$ do
    $c_k = f\left(\sum_{j=1}^{u} h_j * W_{jk}^o\right)$
    If
    $Error = \frac{1}{2}\sum_{k=1}^{w}(o_k - c_k)^2 > threshold$
    then
            // Back propagation stage:
    $\Delta W_{jk}^o = -(o_k - c_k) * h_j$
    $\Delta W_{ij}^h = -h_j(1 - h_j)\ a_i \sum_{k=1}^{w}\left[(o_k - c_k) * W_{jk}^o\right]$
    $W_{jk} = W_{jk} - \eta\Delta W_{jk}$
    $W_{ij}^h = W_{ij}^h - \eta\Delta W_{ij}^h$
        Else
            // learning finish
```

During the feed forward phase, the weight, sigmoid capacity, and attributes of the previous level are used to calculate the values of each level. The method checks for an inaccuracy within the threshold between the yield estimate and the target estimate in the back propagation setting. Otherwise, all weights will be replaced by equations, and the learning process will be disrupted once more. The learning process does not stop until the defect is either inside the threshold or surpasses the maximum severity. The final weight on each connection is utilised to form an intelligent structure after learning.

$$\Delta W_{jk}^o = -(o_k - c_k)h_j \qquad (8)$$

$$\Delta W_{ij}^{h} = -h_j\left(1-h_j\right)a_i \sum_{k=1}^{w}\left[\left(o_k - c_k\right)*W_{jk}^{o}\right]$$

(9)

4.3.2 SVM

The SVM approach focuses on training events on the outskirts of class descriptions to discover the optimal split hyper planes across classes. Support vectors are the examples used in this training. Other than support vectors, rejection vectors are used to train other vectors. The goal of SVM is to increase generalisation by using a discriminating function to prolong the classification gap. Figure 4 depicts the hyperplane.

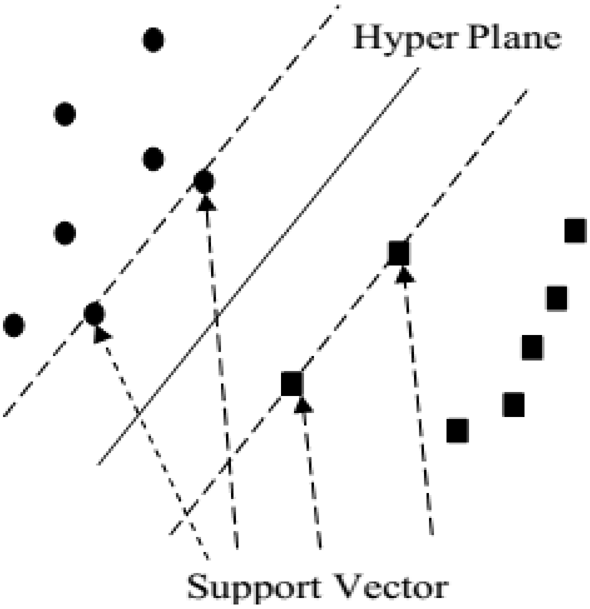

Fig 3: Support Vector Machine

Let the training sample be for all linear classification problems is $\{u_i, v_i\}, \left(i = 1,2,\ldots,m\right)$. Equation gives the arithmetical term for the ideal hyperplane.

$$f(u) = \omega.\varphi(u) + a \qquad (10)$$

Here;

a → Threshold value

ω → Weight factor

SVM classification based on a linear decision function may be formally expressed as,

$$f(u) = \mathrm{sgn}\left(\sum_{i=1}^{m} v_i . b_i . r\left(u_i, u\right) + a\right) \qquad (11)$$

Here;

b_i → Lagrange multiplier

$r(u_i, u)$ → Kernel function

u_i, v_i → Support vectors among any two classes

4.3.3 Classification using HSVNN classifier

In this article, the classification is done with the HSVNN classifier. The following is a step-by-step guide to classification:

Step 1: The ANN input layer receives the specified features (inputs) first. The input layer neuron is represented as $I_1, I_2, \ldots, I_a$, the hidden layer neuron as $H_1, H_2, \ldots, H_b$, and the output layer neuron is represented as $O_1, O_2, \ldots, O_c$ in the ANN. Likewise, the weight function linking the input and hidden layers is denoted by W_{ij}^{1}.

Step 2: To estimate the outcome of the hidden layer, the weight value of the input parameters are multiplied between the input and the hidden layers. The mathematical function for hidden layer output is as follows:

$$H_j = B_1 + \sum_{i=1}^{n} I_i W_{ij}^{1} \qquad (12)$$

Here W_{ij}^{1} denotes the weight rate supplied between the input and hidden layers and n represents the nth input attribute, and is the bias value.

Step 3: After the calculation of the hidden layer outputs, the activation function is applied to Hj. The mathematical function is as follows:

$$F\left(H_j\right)=\frac{1}{1+e^{-H_j}} \qquad (13)$$

Step 4: After the estimation of the hidden layer outputs, it is sent to the SVM. Further, the SVM model was trained using the input data. SVM provides the benefit of minimising the experience classification error while raising the geometric margin at the same time. The input data is mapped to a high-dimensional feature area, which is divided by a hyperplane. Two parallel hyperplanes are produced on either side of the hyperplane that splits the data. The distance between two parallel hyperplanes is increased by using a dividing hyperplane.

Take a look at the data points $\{U_i, V_i\},\ (i=1,2,\ldots,n)$. The hyperplane was utilized to divide the data into two groups. The hyperplane is determined using a subsequent equation.

$$f(U)=\omega.\varphi(U)+B \qquad (14)$$

The weight factor is indicated by ω while the threshold value is denoted by B. The result of the HSVNN is represented in below derivation.

$$O_i = W_{ij}^{2} * \log sig\left[C_1 + \sum_{i=1}^{n} I_i W_{ij}^{1} \right] + C_2$$

(15)

Where W_{ij}^{2} signifies the weight value multiplied by the hidden layer's output. C_1 and C_2, indicate the bias values in the hidden and output layers, respectively.

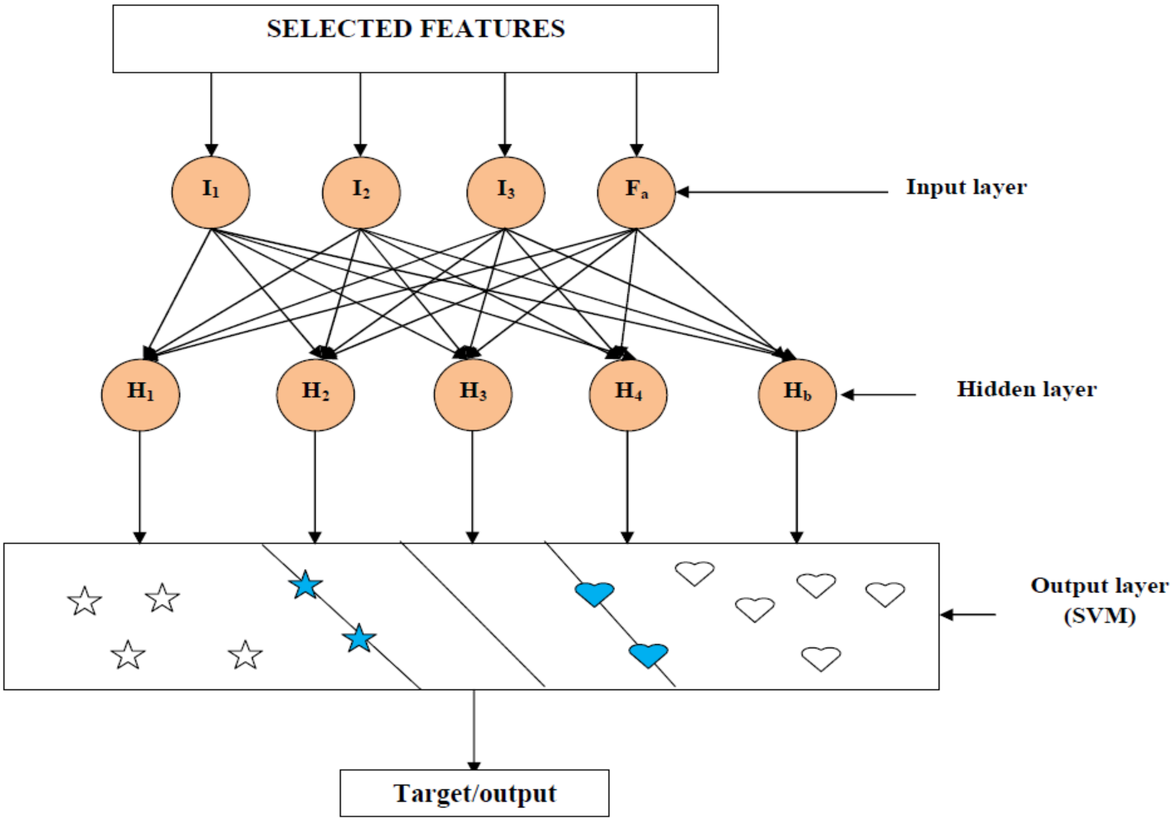

Fig 4: Working process of HSVNN

Equation is used to represent the error function used between the target output and the categorised output.

$$E = \varepsilon_{\max} + \varepsilon_{\min} + \frac{1}{N}\sum_{i=1}^{n}\left|O_{Class} - O_{Tar}\right| \quad (16)$$

Where N indicates the number of features, and O_{Class}andO_{Tar}stand for the categorised and target outputs, respectively. . $\varepsilon_{\min}$ and $\varepsilon_{\max}$ signify the weight vector's minimum and maximum eigenvalues, respectively, and ε may be calculated using equation

$$\varepsilon = Eigen\left(W_{ij} * W_{ij}^{T}\right) \quad (17)$$

$$\varepsilon_{\max} = \max(\varepsilon);\ \varepsilon_{\min} = \min(\varepsilon) \quad (18)$$

The HSVNN system is stored when the training procedure is completed. The trained model's structure is used in the testing and detection procedure.

4.3.4Testing process

We classify medical data as normal or abnormal during the testing procedure. In the experiment, the dataset is divided as 4:1 ratio for training and testing, respectively. The trained network is utilised for prediction when the training procedure is completed. HSVNNs are trained at a variety of facilities during the training process. The score value (t) is acquired after the execution procedure. The value is used to forecast the score.

4. Experimental results and discussion

The evaluation metrics are calculated using the confusion matrix produced from the experimental findings. In this study, the assessment parameters include accuracy, precision, recall, F1 score, and specificity.

4.1 Dataset Description and evaluation metrics

This study makes use of a large binary classification dataset. On July 14, 2014, in Vancouver, Canada, this dataset was utilised in the Evolutionary Computation for Big Data and Big Learning competition. ECBDL14 is the dataset URL from http://cruncher.ncl.ac.uk/bdcomp/. This dataset includes 631 features as well as numerical and categorical metrics and contains around 66 million samples. A total of 75% of the samples are utilised for training, 25% for testing, and 1% for threshold. There are 65003913 training cases and 2897917 test examples in this dataset, totalling 631 features. The instance per split, on the other hand, is around 1984 instances.

- **Accuracy:** It's the proportion of the number of correctly categorised cases to the total number of cases.

$$Accuracy = \left(\frac{(Tp+Tn)}{(Tp+Tn+Fp+Fn)}\right) \quad (19)$$

- **Precision:** It's the proportion of exactlly categorised positive events to the total number of positively expected events.

$$Precision = \left(\frac{Tp}{(Tp+Fp)}\right) \quad (20)$$

- **Recall:** It's the proportion of correctly categorised positive cases among all positive instances.

$$Recall = \left(\frac{Tp}{(Tp+Fn)}\right) \quad (21)$$

- **F1_Score:** It is the average harmonic between recall and precision

$$F1_{Score} = \left(\frac{2\ (Recall \times precision)}{Recall + precision}\right) \quad (22)$$

- **Specificity:** It's the proportion of properly categorised negative events to the overall number of negative occurrences.

$$Specificity = \left(\frac{Tn}{(Tn+Fp)}\right) \quad (23)$$

In the confusion matrix, the numbers of true positives, true negatives, false positives, and false negatives are denoted as TP, TN, FP, and FN, respectively.

The Python programme is used to execute the experiment on a system with 12 GB RAM and an Intel TM core (7M) i3-6100CPU @ 3.70 GHz processor. Python programming was used to create and train the suggested framework. The OpenCV, NumPy, TensorFlow, Keras, and matplotlib python libraries were utilised in this research. OpenCV is used for image processing, such as reading and writing to images. The result generation is handled by sklearn, which uses NumPy for matrix manipulation and TensorFlow for backend packages, Keras for high-level interface, and Matplotlib for graph display.

4.2 Results and discussion

Table 3: confusion table

True Class			
outcome of classifier		Positive	Negative
	Positive	TP	FP
	Negative	FN	TN
Total		Positive	Negative

It is usually important to identify one class as positive and the other as negative when dealing with two-class categorization problems. Take a look at the testing set, which contains both positive and negative samples. Although the goal of every classification algorithm is to assign a class to each sample, certain classifications may be wrong. To assess the classifier's performance, the number of TP, TN, FP, and FN samples is tallied.

4.3 Comparative Analysis

To show the proposed model's performance efficiency, the gathered results from the recommended work are compared to other current approaches. As a consequence, in this part, the findings of the proposed model were compared to those of current techniques.

Table 4: Statistical outcome analysis of proposed and existing methods

Techniques	**Logistic Regression (LR)**	**Naive Bayes (NB)**	**KSVM**	**Proposed method**
Performance metrics				
Accuracy	81.09	86.95	88.19	95
Precision	84.45	88.24	91.18	93.27
Recall	83.13	90	88.13	90
F1_ score	84.23	89.16	89.43	90.34
Specificity	80.12	86.09	87.72	89.21

The suggested methodology is used to compare the different strategies in Table 2. The metrics employed in the comparison study are accuracy, precision, recall, specificity, and F1-score. When compared to current approaches such as Logistic Regression, Nave Bayes, and KSVM, the suggested methodology's accuracy, precision, recall, specificity, and F1-score are 95, 93.27, 90, 90.34, and 89.21 percent, respectively.

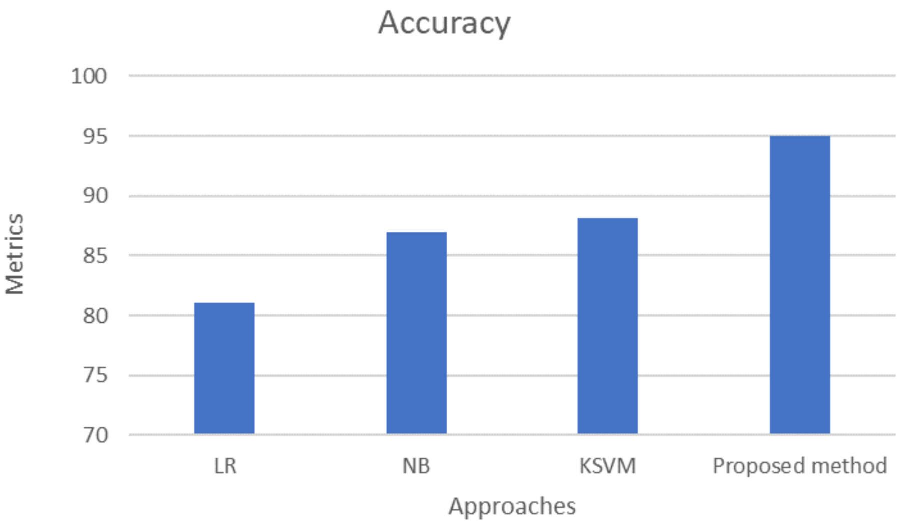

Fig 5: comparative analysis various approaches in terms of accuracy

The graph above depicts a comparison of several methodologies in terms of accuracy. The comparison graph shows that the suggested technique is capable of effectively classifying large amounts of data. The suggested technique has a 95 percent accuracy rate, which is higher than the existing system.

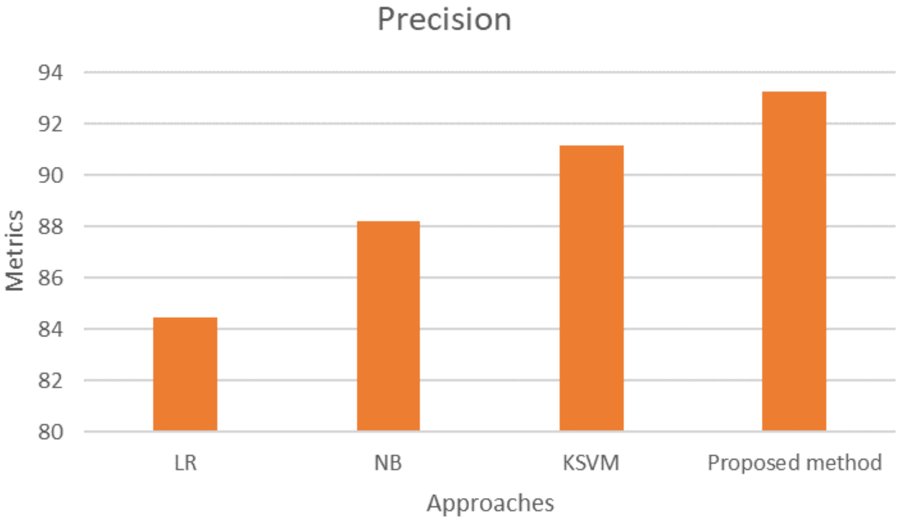

Fig 6: comparative analysis various approaches in terms of Precision

The graphic above shows a validation of system precision percentage with different machine learning technologies. In terms of huge data categorization, the suggested strategy outperformed well, as seen in the comparison graph. The suggested approach has an accuracy of 93.27 percent, which is higher than the other existing methods.

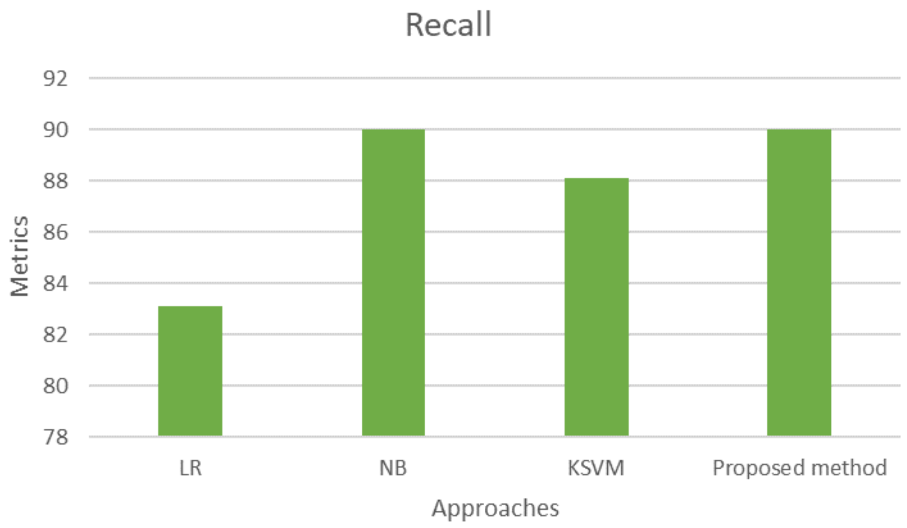

Fig 7: comparative analysis various approaches in terms of Recall

The graphic above shows a comparison of many machine learning algorithms in terms of recall. The comparison graph shows that the suggested technique is capable of effectively classifying large amounts of data. The suggested technique has a recall of 90%, which is significantly higher than the existing system.

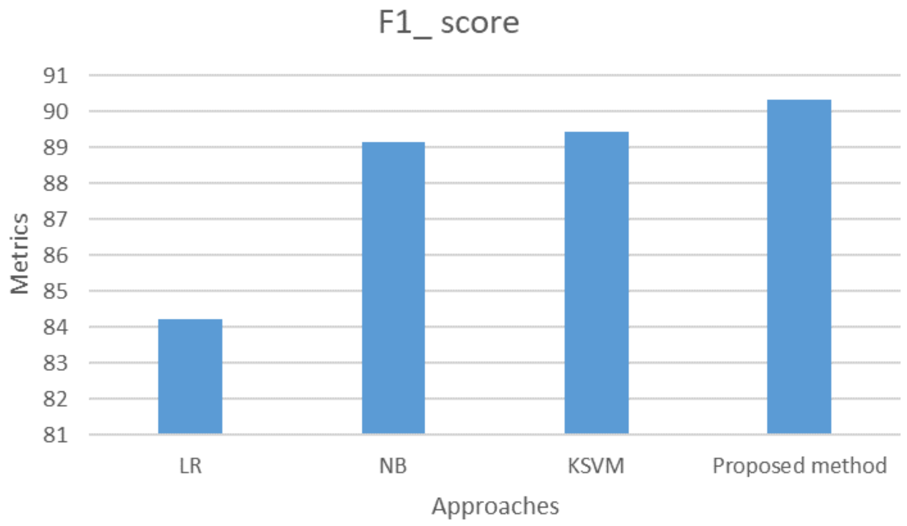

Fig 8: comparative analysis various approaches in terms of F1-score

The graphic above shows a comparison of many machine learning methods in terms of F1-score. In terms of huge data categorization, the suggested strategy outperformed well, as seen in the comparison graph. The suggested technique has an F1-score of 90.34 percent, which is higher than the other existing system.

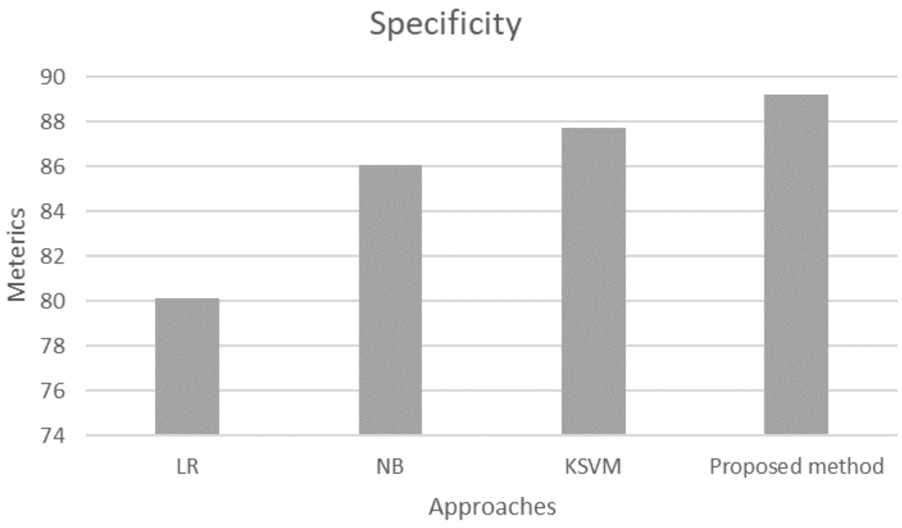

Fig 9: comparative analysis various approaches in terms of specificity

The comparison of several machine learning algorithms in terms of specificity is depicted in the diagram above. The comparison graph shows that the suggested technique is capable of effectively classifying large amounts of data. The suggested technique has a specificity of 89.24 percent, which is higher than the other existing system.

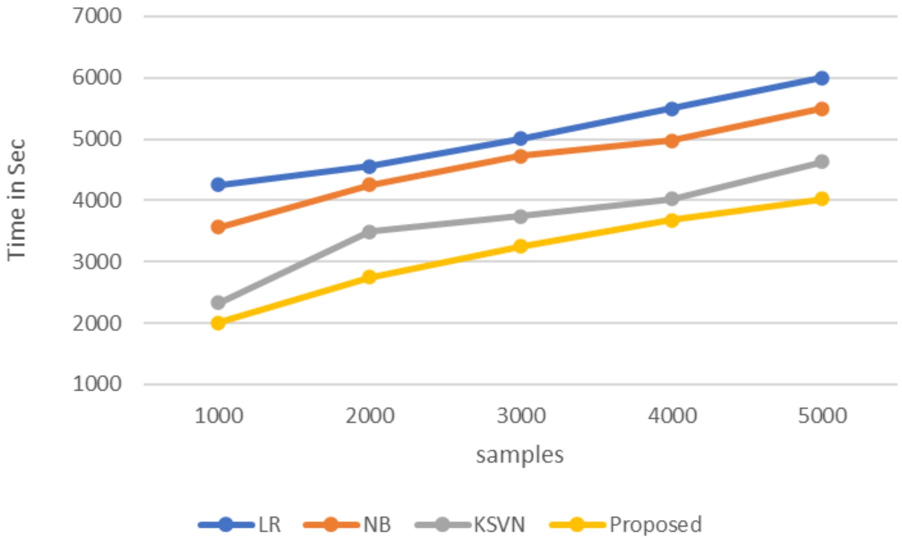

Fig 10: execution time comparison outcomes

On a graph, Figure 10 shows the execution time comparison results for three different classifiers. The proposed classifier's execution time results improve over time when compared to the other three classifiers. The recommended classifier requires less time to execute than the NB, LR, and KSVN classifiers.

5. Conclusion

In this paper, the hybrid Support Vector Machine is associated with the Artificial Neural Network based optimization technique for obtaining theeffective big data classification process. It is an important area of research that has explored various strategies to enhance the performance to determine big data classification problem during imbalance class. This work mainly dealt with the problems of classification based on the large dataset using the hybrid model to reduce features and classification. Finally, the model for validating the proposed approach to validating time and memory usage in classification task is better than the existing study. Moreover, the proposed work is evaluated with various performance metrics for analyzing the proposed technique with the existing technique; therefore, the obtained resultsare proven to a better technique than the existing technique. In future work, the proposed work to be enhanced by adopting the hybrid technique for fault prediction with the utilization of a meta-heuristic algorithm. Besides, future studies will analyze other functional features of the fault prediction method for example Completeness, reliability, and fairness.

Compliance with Ethical Standards

Conflict of interest

The authors declare that they have no conflict of interest.

Human and Animal Rights

This article does not contain any studies with human or animal subjects performed by any of the authors.

Informed Consent

Informed consent does not apply as this was a retrospective review with no identifying patient information.

Funding: Not applicable

Conflicts of interest Statement: Not applicable

Consent to participate: Not applicable

Consent for publication: Not applicable

Availability of data and material:

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Code availability: Not applicable

6. Reference

[1] Devi, S. G., &Sabrigiriraj, M. (2019). A hybrid multi-objective firefly and simulated annealing based algorithm for big data classification. *Concurrency and Computation: Practice and Experience*, *31*(14), e4985.

[2] Xing, W., & Bei, Y. (2019). Medical health big data classification based on KNN classification algorithm. *IEEE Access*, *8*, 28808-28819.

[3] Pintye, I., Kail, E., Kacsuk, P., &Lovas, R. (2021). Big data and machine learning framework for clouds and its usage for text classification. *Concurrency and Computation: Practice and Experience*, *33*(19), e6164.

[4] Li, H., Li, H., & Wei, K. (2018). Automatic fast double KNN classification algorithm based on ACC and hierarchical clustering for big data. *International Journal of Communication Systems*, *31*(16), e3488.

[5] Vennila, V., & Kannan, A. R. (2019). Hybrid parallel linguistic fuzzy rules with canopy mapreduce for big data classification in cloud. *International Journal of Fuzzy Systems*, *21*(3), 809-822.

[6] Fernández, A., del Río, S., Bawakid, A., & Herrera, F. (2017). Fuzzy rule based classification systems for big data with MapReduce: granularity analysis. *Advances in Data Analysis and Classification*, *11*(4), 711-730.

[7] Ulfarsson, M. O., Palsson, F., Sigurdsson, J., & Sveinsson, J. R. (2016). Classification of big data with application to imaging genetics. *Proceedings of the IEEE*, *104*(11), 2137-2154.

[8] Banchhor, C., &Srinivasu, N. (2020). Integrating Cuckoo search-Grey wolf optimization and Correlative Naive Bayes classifier with Map Reduce model for big data classification. *Data & Knowledge Engineering*, *127*, 101788.

[9] Al-Sharo, Y. M., Shakah, G., Alkhaswneh, M. S., Alju-Naeidi, B. Z., &Alazzam, M. B. (2018). Classification of big data: machine learning problems and challenges in network intrusion prediction. *Int. J. Eng. Technol*, *7*(4), 3865-3869.

[10] L'heureux, A., Grolinger, K., Elyamany, H. F., &Capretz, M. A. (2017). Machine learning with big data: Challenges and approaches. *Ieee Access*, *5*, 7776-7797.

[11] Hassanat, A. B. (2018). Two-point-based binary search trees for accelerating big data classification using KNN. *PloS one*, *13*(11), e0207772.

[12] Al-Thanoon, N. A., Algamal, Z. Y., &Qasim, O. S. (2021). Feature selection based on a crow search algorithm for big data classification. *Chemometrics and Intelligent Laboratory Systems*, *212*, 104288.

[13] Segatori, A., Marcelloni, F., &Pedrycz, W. (2017). On distributed fuzzy decision trees for big data. *IEEE Transactions on Fuzzy Systems*, *26*(1), 174-192.

[14] Ma, Z., Yang, L. T., & Zhang, Q. (2020). Support Multimode Tensor Machine for Multiple Classification on Industrial Big Data. *IEEE Transactions on Industrial Informatics*, *17*(5), 3382-3390.

[15] Ali, A. H., & Abdullah, M. Z. (2020). A parallel grid optimization of SVM hyperparameter for big data classification using spark Radoop. *Karbala International Journal of Modern Science*, *6*(1), 3.

[16] Saki, M., Abolhasan, M., & Lipman, J. (2019). A novel approach for big data classification and transportation in rail networks. *IEEE Transactions on Intelligent Transportation Systems*, *21*(3), 1239-1249.

[17] Jiang, C., & Li, Y. (2019). Health big data classification using improved radial basis function neural network and nearest neighbor propagation algorithm. *IEEE Access*, *7*, 176782-176789.

[18] Hababeh, I., Gharaibeh, A., Nofal, S., & Khalil, I. (2018). An integrated methodology for big data classification and security for improving cloud systems data mobility. *IEEE Access*, *7*, 9153-9163.

[19] Maillo, J., Triguero, I., & Herrera, F. (2020). Redundancy and complexity metrics for big data classification: towards smart data. *IEEE Access*, *8*, 87918-87928.

[20] Ma, Z., Yang, L. T., & Zhang, Q. (2020). Support Multimode Tensor Machine for Multiple Classification on Industrial Big Data. *IEEE Transactions on Industrial Informatics*, *17*(5), 3382-3390.

[21] Zhu, M., & Chen, Q. (2020). Big data image classification based on distributed deep representation learning model. *IEEE Access*, *8*, 133890-133904.

[22] Yang, L. H., Liu, J., Wang, Y. M., & Martínez, L. (2018). A micro-extended belief rule-based system for big data multiclass classification problems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

[23] Lakshmanaprabu, S.K., Shankar, K., Ilayaraja, M., Nasir, A.W., Vijayakumar, V. and Chilamkurti, N., 2019. Random forest for big data classification in the internet of things using optimal features. International journal of machine learning and cybernetics, 10(10), pp.2609-2618.

[24] Wang, L., Qian, Q., Zhang, Q., Wang, J., Cheng, W. and Yan, W., 2020. Classification model on big data in medical diagnosis based on semi-supervised learning. The Computer Journal.

[25] Azar, A.T. and Hassanien, A.E., 2015. Dimensionality reduction of medical big data using neural-fuzzy classifier. Soft computing, 19(4), pp.1115-1127.

[26] Lee, C.H. and Yoon, H.J., 2017. Medical big data: promise and challenges. Kidney research and clinical practice, 36(1), p.3.

[27] Fernández, A., del Río, S., Chawla, N.V. and Herrera, F., 2017. An insight into imbalanced big data classification: outcomes and challenges. Complex & Intelligent Systems, 3(2), pp.105-120.

[28] Yang, Y., 2020. Medical multimedia big data analysis modeling based on DBN algorithm. IEEE Access, 8, pp.16350-16361.

[29] Pramanik, P.K.D., Mukhopadhyay, M. and Pal, S., 2021. Big data classification: Applications and challenges. In Artificial Intelligence and IoT (pp. 53-84). Springer, Singapore.

[30] Hassib, E., El-Desouky, A., Labib, L. and El-kenawy, E.S.M., 2020. WOA+ BRNN: An imbalanced big data classification framework using Whale optimization and deep neural

network. Soft Computing, 24(8), pp.5573-5592.

[31] Hassib, E.M., El-Desouky, A.I., El-Kenawy, E.S.M. and El-Ghamrawy, S.M., 2019. An imbalanced big data mining framework for improving optimization algorithms performance. IEEE Access, 7, pp.170774-170795.

[32] Jayasri, N.P. and Aruna, R., 2021. Big data analytics in health care by data mining and classification techniques. ICT Express.