International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8095
Article Received: *25 June 2023* ***Revised:*** *20 August 2023* ***Accepted:*** *05 September 2023*

Slime Mold Optimization with Relational Graph Convolutional Network for Big Data Classification on Apache Spark Environment

[1,*]Mr. K. Manivannan, [2]Dr. T. Suresh
Research Scholar, Department of Information Technology,
V.S.B. Engineering College, Karur, India.
manivannan.vsbec@gmail.com
[2]Associate Professor, Department of Computer Science and Engineering,
Annamalai University, Chidambaram, India.
sureshaucse@gmail.com

Abstract—Lately, Big Data (BD) classification has become an active research area in different fields namely finance, healthcare, e-commerce, and so on. Feature Selection (FS) is a crucial task for text classification challenges. Text FS aims to characterize documents using the most relevant feature. This method might reduce the dataset size and maximize the efficiency of the machine learning method. Various researcher workers focus on elaborating effective FS techniques. But most of the presented techniques are assessed for smaller datasets and validated by a single machine. As textual data dimensionality becomes high, conventional FS methodologies should be parallelized and improved to manage textual big datasets. This article develops a Slime Mold Optimization based FS with Optimal Relational Graph Convolutional Network (SMOFS-ORGCN) for BD Classification in Apache Spark Environment. The presented SMOFS-ORGCN model mainly focuses on the classification of BD accurately and rapidly. To handle BD, the SMOFS-ORGCN model uses an Apache Spark environment. In the SMOFS-ORGCN model, the SMOFS technique gets executed for reducing the profanity of dimensionality and to improve classification accuracy. In this article, the RGCN technique is employed for BD classification. In addition, Grey Wolf Optimizer (GWO) technique is utilized as a hyperparameter optimizer of the RGCN technique to enhance the classification achievement. To exhibit the better achievement of the SMOFS-ORGCN technique, a far-reaching experiments were conducted. The comparison results reported enhanced outputs of the SMOFS-ORGCN technique over current models.

Keywords- Apache Spark; Graph convolutional network; Slime mold optimization; Big data classification; Feature selection.

I. INTRODUCTION

Learning from very huge databases becoming a major problem for many current data mining and Machine Learning (ML) approach [1]. This issue can be typically named BD that is the disadvantages and difficulties of analyzing and processing vast amounts of data [2]. It has grabbed more interest in a large number of areas like financial businesses, bioinformatics, marketing, and medicine due to the massive collections of raw data which were stored [3]. Current advancements in Cloud Computing (CC) technologies enable to adapt of standard data mining methods to implement successfully massive quantities of information. The adaptation of data mining gadgets for BD perplexities may require remodelling of the approaches and their inclusion in parallel atmospheres. Lately, novel and more flexible workflows have occurred for extending the standard MapReduce method, namely Apache Spark, which was applied successfully over several ML and data mining perplexities [4]. Data pre-processing approaches, and highly concrete data reduction methods, were envisioned for cleaning and simplifying input data. Therefore, they try to quicken data mining approaches and even for enhancing their accurateness by reducing redundant and noisy data [5].

FS serves the main part in data mining, particularly in text classifier task which suffers from large dimensionality in several application fields like spam filtering, sentiment analysis, and emotion identification [6]. FS focuses on selecting informative and appropriate words from huge datasets. Thus, the FS could minimize space dimensionality, diminishes the run time in the classifier process, and enhance the efficacy of ML approaches [7]. So, FS was regarded as a serious approach due to it directly affects the classifier accuracy [8]. In addition, though many available FS approaches for text classifiers were filter-related, such techniques do not work if the datasets were huge because they were reliable on the serial programming method. Traditional FS methods are needed to read data into memory for scrutiny, however, a limited memory could not be able to deal with storage and processing of huge datasets [9, 10]. Therefore, FS approaches were required for distributed atmospheres, like Hadoop, a robust tool for distributed processing and storage of huge datasets.

In [11], an adjusted Multi-Layer Particle Swarm Optimization (MPSO) method has been effectively implemented for selecting the Feature Sets (FS). For data classification, the Multi-Layer Perceptron utilizing Stochastic Gradient Descent (MLP-SGD) structure was employed. The devised student performance methods have complied with the Radial Basis Function (RBF) method for minimizing the misclassified ones in the data which is collected, leading to enhanced classifier outcomes. The researchers in [12] modelled a New Map that reduced related extreme learning and parallel FS for microarray cancer data classifications. The next stage employs a wrapper method that utilizes the Adaptive Whale Optimization Algorithm (AWOA) having the Nelder–Mead Algorithm (NMA) for the accomplishment of the gene selection. Wrapper methods were utilized for describing the FS selection process as a search problem.

Hassib et al. [13] formulate a new classifier structure for BD which is made up of 3 developed stages. The primary phase is the FS phase which leverages the Whale Optimization Algorithm (WOA) to find the optimal FS. The next phase is the pre-processing phase which employs the SMOTE and the LSH-SMOTE technique to solve the class imbalance issue. In conclusion, the last stage was WOA + BRNN approach that uses the WOA to train a DL technique known as bidirectional RNN for the first time. Alarifi et al. [14] present a BD and ML approaches to measure Sentiment Analysis (SA) course to address these challenges. The data can be accumulated from a vast amount of data sets, useful in the effectual analysis of the system. In [15], an innovative binary variant of wrapper FS GWO and PSO was suggested. The K-NN method together the matrices of Euclidean separation were leveraged for discovering the optimum solutions. A tent chaotic map helps evade the model from locked to local optima issues.

This article develops a Slime Mold Optimization based FS with Optimal Relational Graph Convolutional Network (SMOFS-ORGCN) for BD Classification in Apache Spark Environment. The presented SMOFS-ORGCN model mainly focuses on the classification of BD accurately and rapidly. To handle BD, the SMOFS-ORGCN model uses an Apache Spark environment. In the SMOFS-ORGCN model, the SMOFS technique gets executed for reducing the profanity of dimensionality and to improve classification accuracy. In this article, the RGCN technique is employed for BD classification. In addition, Grey Wolf Optimizer (GWO) technique is utilized as a hyperparameter optimizer of the RGCN technique to enhance the classification achievement. To exhibit the better achievement of the SMOFS-ORGCN technique, far-reaching experiments were conducted.

II. THE PROPOSED MODEL

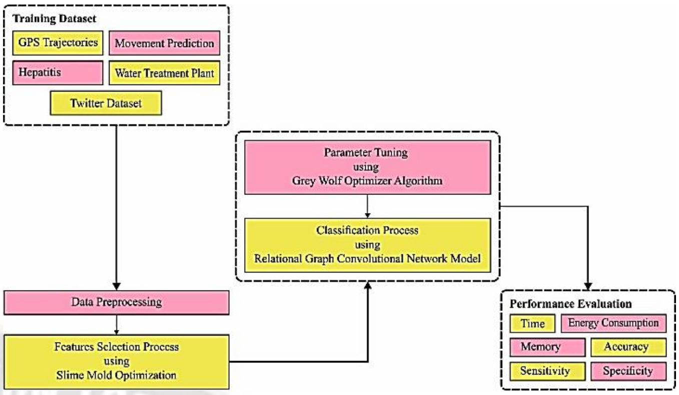

Figure 1. Procedures involved in the SMOFS-ORGCN model

In this article, a novel SMOFS-ORGCN model is introduced for the precise classification of BD. To handle BD, the SMOFS-ORGCN model uses an Apache Spark environment. The proposed model comprises SMOFS-based feature subset selection, RGCN and GWO-based classification and hyperparameter tuning. The workflow is illustrated in Fig. 1.

A. Apache Spark

For managing BD, the SMOFS-ORGCN approach uses an Apache Spark environment. The Apache ecosystem is displayed in Fig. 2. Apache Spark is a super-fast incorporated analytics engine and a popular open-source cluster-computing architecture [16] for large-scale data processing. Spark analytics platform has become more prominent over Hadoop MapReduce owing to the various advantages it provides.

Apache Spark contains a built-in stack of libraries as follows.

Support SQL interface to query data structure and process over a larger cluster

Spark Streaming makes them easier to construct scalable fault-tolerant streaming application

GraphX API for graphs and its parallel calculation

MLlib for scalable ML

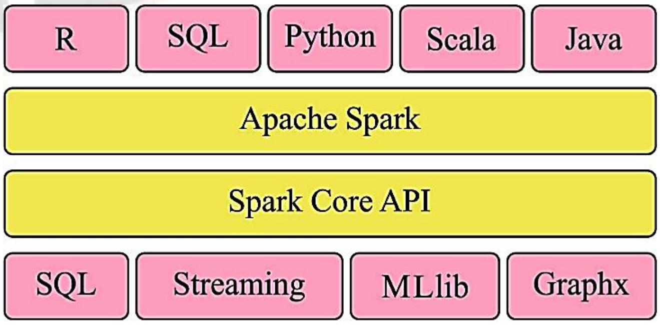

Figure 2. Apache Ecosystem

DOI: https://doi.org/10.17762/ijritcc.v11i11s.8095
Article Received: *25 June 2023* ***Revised:*** *20 August 2023* ***Accepted:*** *05 September 2023*

Spark Resilient Distributed Dataset (RDD) that assists in-memory fault-tolerant distributed computation, Directed Acyclic Graph (DAG) based query optimization, and scheduling technique for streaming and batch computations. Spark-based application is easier to program and utilize dissimilar programming languages such as SQL, Java, Python, and Scala R. Eventually, Spark framework is hosted over its cluster, Hadoop YARN, EC2, or container orchestration schemes such as Apache Mesos and Kubernetes. Also provides smooth incorporations with various file formats, open-source databases, and streaming apps.

B. Process involved in SMOFS Technique

In the SMOFS-ORGCN technique, the SMOFS technique gets executed for reducing the profanity of dimensionality and improve classification accuracy [17]. The SMO model is the new nature-inspired technique. It represents the mathematical modelling of mimicking the propagation wave of SM while making the optimum way to connect food. This technique adoptively mimics the procedure of producing negative and positive feedback in the propagation wave. This process is integrated into dissimilar optimization issues involving the engineering one. The major two phases in the SMO process are named warp and approaching foods.

Approaching food: Here, the slime is approaching food as per the odour in the atmosphere, and such behaviours are arithmetically defined by:

$$\mathrm{X(t+1)} = \begin{cases} \overrightarrow{X_b}(t) + \overrightarrow{vb} \times (\overrightarrow{W} \times \overrightarrow{X_A(t)} - \overrightarrow{X_B}(t)), r < p \\ \overrightarrow{vc} \times \vec{X}(t), r \geq p \end{cases} \tag{1}$$

Here, $\overrightarrow{vb}$ denotes a variable that ranges from -a to $a, \overrightarrow{vc}$ is a variable that reduces from [0,1], in a linear formation, X_b is the present individual position corresponding to higher odour smell, t represents the existing iteration, X characterizes the position of the SM, X_A and X_B are arbitrarily chosen from the SM, and W epitomizes the weight of SM. The equation of p is characterized by:

$$\text{p= } \tanh\,[S(i) - DF] \text{ where i.e., } 1, 2, 3, \cdots, n \tag{2}$$

$where\ S(i)$ epitomizes the fitness of $\vec{X}$ and DF is the optimal fitness over each iteration. From the abovementioned equations, $v\vec{b}$ ranges from -a to a, and it is defined by:

$$a = arctanh(-(\frac{t}{\max t}) + 1) \tag{3}$$

The $\overrightarrow{W}$ equation is determined as follows:

$$\overline{W(SmeelIndex(t))} = \begin{cases} 1 + r \times \log(\frac{bF - S(i)}{bF - wF} + 1), & condition \\ 1 - r \times \log(\frac{bF - S(i)}{bF - wF} + 1), & others \end{cases} \tag{4}$$

where SmeelIndex denotes the series of fitness values. r represents the arbitrary number in [0,1] , bF signifies the optimum fitness acquired in the existing iteration method, $and\ wF$ characterizes the worst fitness attained in the existing iterative method.

Warp food: Here, the behaviour of the slime can be expressed in the following.

$$\overrightarrow{X^*} = \begin{cases} rand\ \times (UB - LB) + LB, rand\ < z \\ X(\overrightarrow{t}) + \overrightarrow{vb} \rightarrow\times (W \rightarrow\times \overrightarrow{X_A}(t) - \overrightarrow{X_B}(t))\ , r < p \\ \overrightarrow{vc} \times \vec{X}(t), r \geq p \end{cases} \tag{5}$$

In Eq. (5), LB and UB symbolize the lower and upper restrictions of the searching space, $and\ rand$ and r symbolize arbitrary parameter ranges within [0,1].

The Fitness Function (FF) utilized in the proposed technique is developed to have a stability among the classification accuracy (maximum) and the amount of chosen factors in all the solutions (minimum) attained via the chosen feature, Eq. (6) characterizes the FF to assess a solution.

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|C|} \tag{6}$$

In Eq. (6), $\gamma_R(D)$ characterizes the classifier error rate of the provided classification. $|R|$ denotes the chosen subset's cardinality and $|C|$ indicates the overall amount of factors in the data, α, and β indicates two variables subsequent to the subset length and classifier quality's criticalness. $\in [1,0]$ and $\beta = 1 - \alpha$.

C. BD classification using RGCN Model

In this work, the RGCN module is utilized for the categorization of BD [18]. A Graph Convolution Network (GCN) is a topological networking mechanism that depends on the graph concept that is initially suggested to manage non-Euclidean datasets. In this work, the convolution function in GCN is implemented for realizing the differentiable data transmission technique of neighbouring graph nodes. Usually, the transferred data is the hidden layer of the node that is a higher-dimension feature vector. GCN has the benefit of data processing. Each datum, word, and symbol in the text is assumed as a network node. According to the word co-occurrence relationships and the relationships among the dataset, a text graph is constructed for a certain corpus, and later a text GCN (text-GCN) mechanism is constructed. Consider that a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains an edge $(v_j, v_j) \in \mathcal{E}$ and node $v_j \in \mathcal{V}$.

All the nodes v_i encompass a self-loop edge, such as $(v_i, v_j) \in \mathcal{E}$. Consider $X \in R^{n\times m}$ as a matrix encompassing the eigenvector of n node, whereby m denotes the dimension of the eigenvector and all the rows of $x_v \in R^m$ denote the eigenvectors of node v. Given that D indicates the degree matrix of $\mathcal{G}$, as well as A, represents the adjacent matrix of graph $\mathcal{G}$, whereby $D_{ii} = \Sigma_j A_{ij}$. Single convolution layers the GCN captures near-domain data. Once numerous GCN layers are arranged, large data is collected. For a one-layer GCN, k dimension node feature matrix $L^{(1)} \in R^{n\times k}$ is evaluated by:

$$L^{(1)} = \rho(\tilde{A}XW_0) \tag{7}$$

In Eq. (7), $\tilde{A} = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ denotes the normalized symmetric adjacent matrix and $W_0 \in R^{m\times k}$ shows the weight matrix. From the abovementioned, high-order neighbourhood data is integrated by stacking numerous GCN layers.

$$L^{j+1} = \rho(\tilde{A}L^j W_j) \tag{8}$$

Whereby j denotes the layer count and $L^0 = X$.

Consequently, the forward propagation method in the R-GCN is determined by

$$h_i^{(l+1)} = \text{ReLU}\,(\textstyle\sum_{u\in\mathcal{N}(v_i)} \frac{1}{c_i} W^{(l)} h_u^{(l)}) \tag{9}$$

In Eq. (3), $\mathcal{N}_r(v_j)$ characterizes the collection of neighbouring nodes that relationships are r for node i, l signifies the layer count and c_j represents a normalized constant. R-GCN structure is presented for resolving heterogeneous graph challenges from which dissimilar edges have dissimilar descriptions of relation.

D. Hyperparameter Tuning by Implementing GWO

In the last stage, the GWO model is utilized as a hyperparameter optimizer of the RGCN technique to enhance the classification performance [19]. GWO is a populace-based metaheuristic model derivative from the social activities of grey wolves that would rather exist in a group composed of 5-12 wolves. The grey wolf has a strict social order. First, the grey wolf approach, tracks and chase the target. Then, harass, pursue and encircle the target until it stops moving. The last stage is to attack the target.

The GWO process models the two social behaviours of wolf group hunting and social hierarchy. Wolf in the pack characterizes a possible solution to the optimization issue. The optimum solution is termed alpha (α). The 2nd and 3rd optimal solutions are named beta (β) and delta δ correspondingly. The residual solution is named omega (ω). The hunting can be controlled by α, β, and δ. ω follow the three candidates and it is given as follows.

$$\vec{X}(t+1) = \vec{X}(t) + \vec{A}.\vec{D} \tag{10}$$

In Eq. (11), t indicates the iteration count, $\vec{A}$ and $\vec{C}$ refer to the coefficient vector, $\overrightarrow{X_p}$ and $\vec{X}$ represents the location of the prey and grey wolf.

$$\vec{D} = |\vec{C}.\vec{X}(t) - \vec{X}(t) \tag{11}$$

$$\vec{A} = 2a.\vec{r}_1 - a \tag{12}$$

$$\vec{C} = \overrightarrow{2r_2} \tag{13}$$

Now a decreased sequentially from two to zero, r_1 and r_2 denote arbitrary vectors within [0,1]. Replicating the hunting behaviour of grey wolfs has improved acquaintance regarding the likely location of prey. Then, other wolves upgrade the location of the searching agent depending on the position of the optimal searching agent. The location of the wolf is upgraded as follows

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{14}$$

Here $\vec{X}_1, \vec{X}_2$ and $\vec{X}_3$ are described as follows:

$$\vec{X}_1 = \vec{X}_{1\alpha} - \vec{A}_1.(\vec{D}_\alpha) \tag{15}$$

$$\vec{X}_2 = \vec{X}_{2\alpha} - \vec{A}_2.(\vec{D}_\beta) \tag{16}$$

$$\vec{X}_3 = \vec{X}_{3\alpha} - \vec{A}_2.(\vec{D}_\gamma) \tag{17}$$

From the equation, $\vec{X}_\alpha$, $\overrightarrow{X_\beta}$ and $\vec{X}_\gamma$ denote the position of the optimal solution, and $\vec{D}_\alpha \vec{D}_\beta$, and $\vec{D}_\gamma$ are determined as follows.

$$\vec{D}_\alpha = |\vec{C}_1.\vec{X}_\alpha - \vec{X}\,| \tag{18}$$

$$\vec{D}_\beta = |\vec{C}_2.\vec{X}_\beta - \vec{X}\,| \tag{19}$$

$$\vec{D}_\gamma = |\vec{C}_3.\vec{X}_\gamma - \vec{X}\,| \tag{20}$$

where a parameter regulates the balance between the exploitation and exploration as given below.

$$a = 2 - r\frac{2}{\text{Max}\,lter} \tag{21}$$

where t indicates the iteration count and $MaxlTer$ represents the maximal iterating amount. The GWO model derives an FF to accomplish an enhanced performance of the categorization. It describes a positive integer for characterizing the improved accomplishment of the solution candidate. Also, the mitigation of the classifier rate of error is considered as the FF. The optimal solution has a minimum error rate and the worse solution achieves an enhanced rate of error.

$$fitness(x_i) = ClassifierErrorRate(x_i)$$

$$= \frac{No.of\ misclassified\ instances}{Overall\ instances} * 100 \qquad (22)$$

III. RESULTS AND DISCUSSION

In this segment, the investigational validation of the SMOFS-ORGCN approach is examined by employing four datasets. The results are inspected under a distinct number of Data Size (DS). Table 1 offers the comprehensive FS outputs of the SMO-FS approach on four datasets. The investigational values implied that the SMO-FS methodology has chosen an optimum number of features on each dataset. On the GPS Trajectory (GPS-T) dataset, this approach has elected 7 features out of fifteen. Moreover, On GPS Movement Prediction (GPS-MP) dataset, the SMO-FS model has elected 1 feature out of 4 features. In addition, On the GPS Water Treatment Planet (GPS-WTP) dataset, the SMO-FS model has elected 18 features out of 38 features.

TABLE I. FS OUTCOMES OF SMO-FS TECHNIQUE

Applied Dataset	Overall Features	Chosen Features
GPS-T	15	07
GPS-MP	04	01
GPS-WTP	38	18
Hepatitis	19	08
Twitter	02	01

Fig. 3 provides the maximum cost investigation of the SMO-FS approach with other FS techniques. The figure represented that the SMOFS-ORGCN method has obtained a lower best cost over other techniques on each dataset.

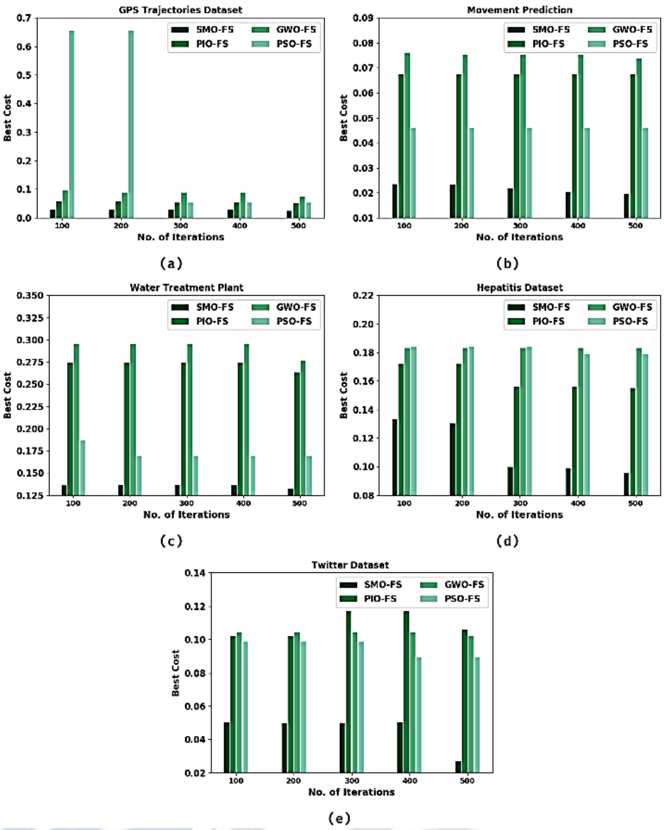

Figure 3. Best Cost of SMO-FS with other FS models a) GPS-T b) GPS-MP c) GPS-WTP d) Hepatitis e) Twitter Dataset

Table 2 gives a time and memory evaluation of the SMOFS-ORGCN method under diverse DS. The outcomes implied that the SMOFS-ORGCN method required optimal time and memory. For instance, with 50Mb data, the SMOFS-ORGCN approach has obtained a time of 36s and memory of 1037875bytes. Likewise, with 100Mb data, the SMOFS-ORGCN approach has acquired a time of 37s and memory of 1174355bytes. Similarly, with 150Mb data, the SMOFS-ORGCN approach has acquired time and memory of 95s and 1256658bytes.

TABLE II. TIME AND MEMORY EVALUATION OF THE SMOFS-ORGCN MODEL

DS (Mb)	Time (Sec)	Memory (byte)
50	36	1037875
100	37	1174355
150	95	1256658
200	135	1304980
250	165	1369464
300	227	1402799

A relative throughput investigation of the SMOFS-ORGCN method with and without Hadoop optimization under different DS is given in Table 3 and Fig. 4. The outputs portrayed that the

***Article Received:** 25 June 2023 **Revised:** 20 August 2023 **Accepted:** 05 September 2023*

SMOFS-ORGCN method has acquired efficient throughput under optimization. As a sample, with a DS of 50Mb, the SMOFS-ORGCN method has offered a throughput of 8094kbps and 1864.84kbps under HO and HWO respectively. For example, with a DS of 150Mb, the SMOFS-ORGCN method has granted throughput of 9561kbps and 7532.94kbps under HO and HWO correspondingly. Meanwhile, with a DS of 300Mb, the SMOFS-ORGCN method has presented a throughput of 12818kbps and 1694.39kbps under HO and HWO correspondingly.

TABLE III. THROUGHPUT (KBPS) EVALUATION OF PROJECTED MODEL ON DISSIMILAR DS

DS (Mb)	Hadoop with Optimization (HO)	Hadoop without Optimization (HWO)
50	8094.00	1864.84
100	6717.00	1048.48
150	9561.00	7532.94
200	7751.00	5294.85
250	10322.00	2289.61
300	12818.00	1694.39

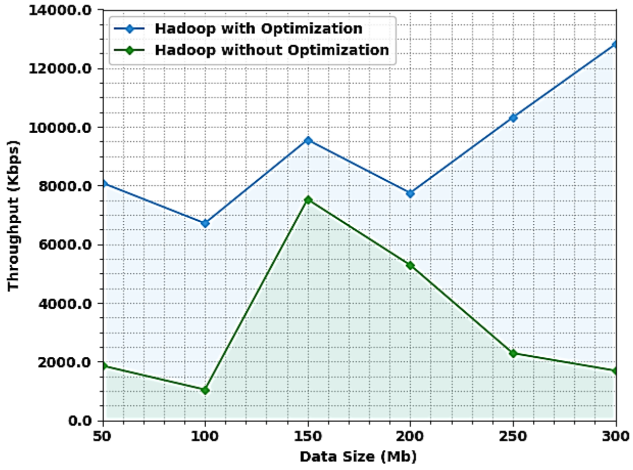

Figure 4. Throughput evaluation of SMOFS-ORGCN technique

Table 4 and Fig. 5 exhibit the SMOFS-ORGCN technique has attained effectual ECON under optimization. Additionally, with a DS of 50Mb, the SMOFS-ORGCN technique has presented ECON of 91.17 and 202.58 under HO and HWO respectively. Moreover, with a DS of 150Mb, the SMOFS-ORGCN model has attained ECON of 1217.74 and 180.46 under HO and HWO correspondingly.

TABLE IV. ENERGY CONSUMPTION EVALUATION OF PROPOSED MODEL ON DISSIMILAR DS

DS (Mb)	Hadoop with Optimization	Hadoop without Optimization
50	91.17	202.58
100	60.16	163.32
150	121.74	180.46
200	179.32	298.31
250	189.55	320.89
300	154.17	253.09

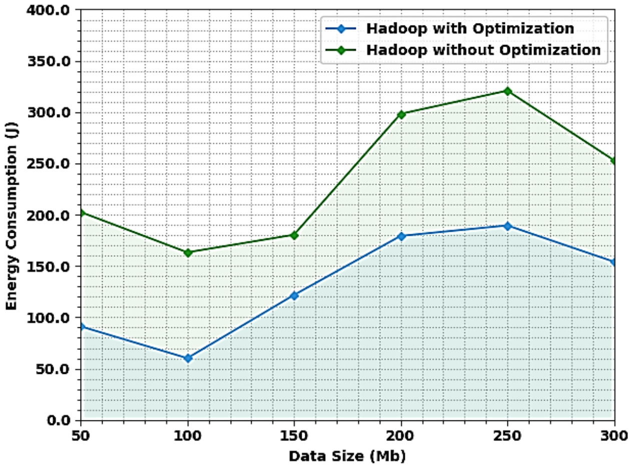

Figure 5. ECON analysis of SMOFS-ORGCN technique

Meanwhile, with a DS of 300Mb, the SMOFS-ORGCN model has granted ECON of 154.17kbps and 253.09kbps under HO and HWO correspondingly.

Fig. 6 shows the comparative $sens_y$ examination of the SMOFS-ORGCN methodology with other recent models on the applied datasets [20, 21]. The figure portrayed that the SMOFS-ORGCN methodology has given an outcome in maximum $sens_y$ values on each dataset.

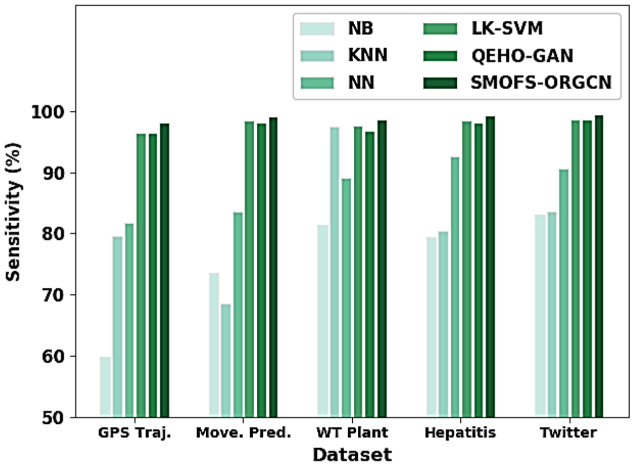

Figure 6. Comparative $Sens_y$ examination of SMOFS-ORGCN with existing models

As a sample, with the GPS-T dataset, the SMOFS-ORGCN methodology has gained a greater $sens_y$ of 98.12% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methodology have reported a lower $sens_y$ of 59.78%, 79.65%, 81.74%, 96.55%, and 96.42% respectively. Also, with the GPS-MP dataset, the SMOFS-ORGCN approaches has attained a greater $sens_y$ of 99.11% while the NB, KNN, NN, LK-SVM, and QEHO-GAN approaches have reported a lower $sens_y$ of 73.52%, 68.46%, 83.54%, 98.44%, and 98.16% correspondingly. Moreover, with the GPS-WTP, the SMOFS-ORGCN technique has attained a higher $sens_y$ of 98.66% while the NB, KNN, NN, LK-SVM, and QEHO-GAN techniques have reported lower $sens_y$ of 81.36%, 97.46%, 89.09%, 97.65%, and 96.87% correspondingly. Finally, with the Hepatitis dataset, the SMOFS-ORGCN technique gained a greater $sens_y$ of 99.38% but the NB, KNN, NN, LK-SVM, and QEHO-GAN techniques have reported a lesser $sens_y$ of 79.46%, 80.45%, 92.54%, 98.47%, and 98.12% correspondingly.

Fig. 7 displays the relative $spec_y$ evaluation of the SMOFS-ORGCN methodology with other recent methodologies on the applied datasets [20, 21]. The figure denoted the SMOFS-ORGCN methodology has resulted in maximum $spec_y$ values on each dataset. For example, with the GPS-T dataset, the SMOFS-ORGCN method has gained a greater $spec_y$ of 99.76% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methods have reported lower $sens_y$ of 76.74%, 78.64%, 78.64%, 82.47%, and 98.94% correspondingly. Further, with the GPS-MP dataset, SMOFS-ORGCN approaches have attained a higher $spec_y$ of 99.77% while the NB, KNN, NN, LK-SVM, and QEHO-GAN approaches have reported lower $spec_y$ of 76.38%, 70.48%, 82.94%, 97.42%, and 98.94% correspondingly. Moreover, with the GPS-WTP dataset, the SMOFS-ORGCN methods has gained higher $spec_y$ of 99.51% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methods have reported lower $sens_y$ of 82.47%, 98.74%, 87.60%, 96.49%, and 97.89% correspondingly. At last, for instance, with the Hepatitis dataset, SMOFS-ORGCN model has reached greater $spec_y$ of 99.72% while the NB, KNN, NN, LK-SVM, and QEHO-GAN models have reported lower $spec_y$ of 82.71%, 73.54%, 91.90%, 98.54%, and 99.16% correspondingly.

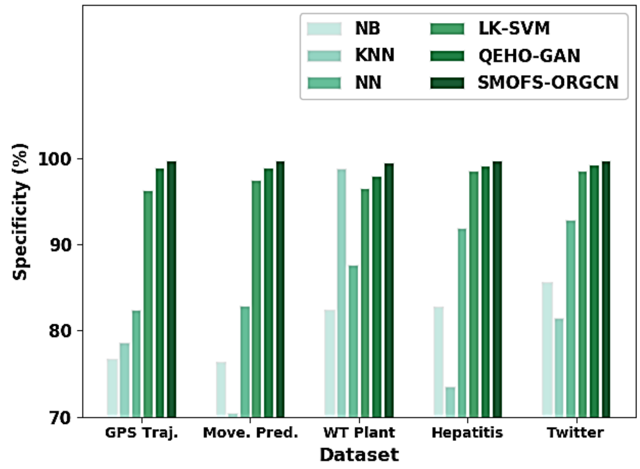

Figure 7. Comparative $Spec_y$ examination of SMOFS-ORGCN with existing models

Fig. 8 depicts the comparative $accu_y$ scrutiny of the SMOFS-ORGCN methodology with other recent models on the applied datasets [20, 21]. The figure denoted the SMOFS-ORGCN methodology has given an outcome in maximum $accu_y$ values on every dataset. As a sample, with GPS-T data, the SMOFS-ORGCN approach has reached a greater $accu_y$ of 99.10% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methods have reported lower $accu_y$ of 80.36%, 70.35%, 81.46%, 96.39%, and 98.32% correspondingly. Also, with the GPS-MP dataset, the SMOFS-ORGCN approach has reached a greater $accu_y$ of 99.95% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methods have reported lower $accu_y$ of 79.51%, 69.91%, 83.14%, 97.40%, and 98.91% correspondingly.

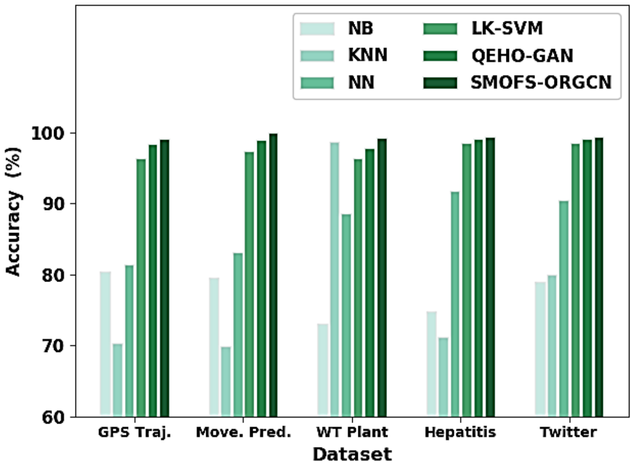

Figure 8. Comparative $Accu_y$ examination of SMOFS-ORGCN with existing models

In addition, with the Water treatment dataset, the SMOFS-ORGCN method has acquired a higher $accu_y$ of 99.18% while the NB, KNN, NN, LK-SVM, and QEHO-GAN methods have reported lower $accu_y$ of 73.02%, 98.72%, 88.58%, 96.36%, and 97.83% respectively. At last, with the Hepatitis dataset, the SMOFS-ORGCN approach has gained a higher $accu_y$ of 99.45% while the NB, KNN, NN, LK-SVM, and QEHO-GAN approaches have reported lower $accu_y$ of 74.83%, 71.21%, 91.69%, 98.53%, and 99.10% correspondingly. These outputs and argument portrayed the superior achievement of the SMOFS-ORGCN approach in the BD classification process.

IV. CONCLUSION

In this article, a new SMOFS-ORGCN technique is presented for performing BD classification accurately and rapidly. To handle BD, the SMOFS-ORGCN model uses an Apache Spark environment. In the SMOFS-ORGCN model, the SMOFS technique gets executed for reducing the profanity of dimensionality and to improve classification accuracy. In this article, the RGCN technique is employed for BD classification. In addition, Grey Wolf Optimizer (GWO) technique is utilized as a hyperparameter optimizer of the RGCN technique to enhance the classification achievement. To exhibit the better achievement of the SMOFS-ORGCN technique, far-reaching experiments were conducted. The comparison results reported enhanced outputs of the SMOFS-ORGCN technique over current models. In the future, optimal feature reduction and outlier removal approaches can be developed to boost the classification performance of the SMOFS-ORGCN technique.

REFERENCES

[1] BenSaid, F. and Alimi, A.M., 2021. Online feature selection system for big data classification based on multi-objective automated negotiation. Pattern Recognition, 110, p.107629.

[2] Rong, M., Gong, D. and Gao, X., 2019. Feature selection and its use in big data: challenges, methods, and trends. IEEE Access, 7, pp.19709-19725.

[3] Abdulwahab, H.M., Ajitha, S. and Saif, M.A.N., 2022. Feature selection techniques in the context of big data: taxonomy and analysis. Applied Intelligence, pp.1-46.

[4] AlNuaimi, N., Masud, M.M., Serhani, M.A. and Zaki, N., 2020. Streaming feature selection algorithms for big data: A survey. Applied Computing and Informatics.

[5] Joseph Manoj, R., Praveena, A. and Vijayakumar, K., 2019. An ACO–ANN-based feature selection algorithm for big data. Cluster Computing, 22(2), pp.3953-3960.

[6] Ghaddar, B. and Naoum-Sawaya, J., 2018. High dimensional data classification and feature selection using support vector machines. European Journal of Operational Research, 265(3), pp.993-1004.

[7] 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.

[8] Soheili, M. and Eftekhari-Moghadam, A.M., 2020. DQPFS: Distributed quadratic programming based feature selection for big data. Journal of Parallel and Distributed Computing, 138, pp.1-14.

[9] Rashid, A.N.M., Ahmed, M., Sikos, L.F. and Haskell-Dowland, P., 2020. Cooperative co-evolution for feature selection in Big Data with random feature grouping. Journal of Big Data, 7(1), pp.1-42.

[10] Shehab, N., Badawy, M. and Ali, H.A., 2022. Toward feature selection in big data preprocessing based on hybrid cloud-based model. The Journal of Supercomputing, 78(3), pp.3226-3265.

[11] Thenmozhi, L. and Chandrakala, N., 2022. Developed Modified Particle Swarm Optimization For Feature Selection On Learning Based Big Data In Cloud Computing. JOURNAL OF ALGEBRAIC STATISTICS, 13(1), pp.310-320.

[12] Hira, S. and Bai, A., 2022. A Novel Map Reduced Based Parallel Feature Selection and Extreme Learning for Micro Array Cancer Data Classification. Wireless Personal Communications, 123(2), pp.1483-1505.

[13] 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.

[14] Alarifi, A., Tolba, A., Al-Makhadmeh, Z. and Said, W., 2020. A big data approach to sentiment analysis using greedy feature selection with cat swarm optimization-based long short-term memory neural networks. The Journal of Supercomputing, 76(6), pp.4414-4429.

[15] El-Hasnony, I.M., Barakat, S.I., Elhoseny, M. and Mostafa, R.R., 2020. Improved feature selection model for big data analytics. IEEE Access, 8, pp.66989-67004.

[16] Esmaeilzadeh, A., Heidari, M., Abdolazimi, R., Hajibabaee, P. and Malekzadeh, M., 2022, January. Efficient large scale nlp feature engineering with apache spark. In 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0274-0280). IEEE.

[17] Zubaidi, S.L., Abdulkareem, I.H., Hashim, K.S., Al-Bugharbee, H., Ridha, H.M., Gharghan, S.K., Al-Qaim, F.F., Muradov, M., Kot, P. and Al-Khaddar, R., 2020. Hybridised artificial neural network model with slime mould algorithm: a novel methodology for prediction of urban stochastic water demand. Water, 12(10), p.2692.

[18] Chen, Z., Huang, K., Wu, L., Zhong, Z. and Jiao, Z., 2022. Relational Graph Convolutional Network for Text-Mining-Based Accident Causal Classification. Applied Sciences, 12(5), p.2482.

[19] Mirjalili, S., Mirjalili, S.M. and Lewis, A., 2014. Grey wolf optimizer. Advances in engineering software, 69, pp.46-61.

[20] Lakshmanaprabu, S.K., Shankar, K., Khanna, A., Gupta, D., Rodrigues, J.J., Pinheiro, P.R. and De Albuquerque, V.H.C., 2018. Effective features to classify big data using social internet of things. IEEE access, 6, pp.24196-24204.

[21] Kaur, I., Lydia, E.L., Nassa, V.K., Shrestha, B., Nebhen, J., Malebary, S. and Joshi, G.P., 2021. Generative Adversarial Networks with Quantum Optimization Model for Mobile Edge Computing in IoT Big Data. Wireless Personal Communications, pp.1-21.