International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V71I12P206

Volume 71 Issue 12, 50-60, December 2023
© 2023 Seventh Sense Research Group®

Original Article

Big Data Analytics Assisted Arithmetic Optimization with Deep Learning Model for Sentiment Classification

K. Manivannan[1], T. Suresh[2], M. Parthiban[3]

[1,2,3]*Department of Computer Science and Engineering, Annamalai University, Chidambaram, India.*

[1]*Corresponding Author: manivannan.vsbec@gmail.com*

Received: 28 July 2023 Revised: 12 October 2023 Accepted: 19 October 2023 Published: 06 December 2023

Abstract - *Sentiment Analysis (SA) may extract data from various text sources like blogs, reviews, and news; later, it categorizes them based on the polarity. Furthermore, big data is generated via social media and mobile networks. The implementation of SA on big data was found to be valuable for the business to take helpful commercial insights from textual-related content. Implementing SA on big data is utilized as a method to classify opinions into different sentiments. This article introduces a new Big Data Analytics Assisted Arithmetic Optimization with Deep Learning Model for Sentiment Classification (BDA-AODLSC) approach. The presented BDA-AODLSC approach exploits BDA tools for sentiment classification. Initially, the BDA-AODLSC approach performs data preprocessing to transform it into a compatible format, and the TF-IDF method is utilized for the word embedding process. An Attention-based Long Short-Term Memory (ALSTM) method is utilized for classifying sentiments, and its hyperparameters can be selected by an Arithmetic Optimization Algorithm (AOA). For managing big data, the Hadoop MapReduce tool is employed. A far-reaching analysis is accomplished to demonstrate the superior accomplishment of the BDA-AODLSC method. The extensive output demonstrates the significant accomplishment of the BDA-AODLSC method over other existing techniques.*

Keywords - *Sentiment Analysis, Deep Learning, Big Data Analytics, Arithmetic Optimization Algorithm, Hadoop MapReduce.*

1. Introduction

Recently, data have been steadily increasing. This increasing data, termed Big Data, is the foundation of the human life revolution in several domains [1]. Generally, the 5 main features of Big Data are veracity, volume, variety, value and velocity. The amalgamation of these 5 features is named 5 Vs. Where "volume" signifies the set of all generated data sets. The "variety" specifies the distinct formats of data from several sources. The "velocity" displays the high speed of data accumulation in the dataset [2]. The "veracity" signifies trustworthiness or data accuracy in the produced dataset. The "value" denotes each type of feature in the produced dataset. Big data analysis is increasing quickly in all industries or fields [3]. In medical science, big data analysis was utilized to cure and prevent distinct diseases like cancer.

The big data mining technique brings about a huge volume of data [4]. Additionally, the rapidity of management is considered. This technique involves optimized techniques (like PSO and GDA techniques), performance assessment technology (for instance, data envelopment analyses and fuzzy comprehensive evaluation), decision analysis technology (like the multi-criteria decision, gray decision, etc.) [5], scientific modelling or other prediction technology (like roughset, NN, NB, DT).

Classification and clustering were the 2 main classes of techniques for extracting data [6]. Yet, the rising data dimension produced the clustering methods and classification performances as the technique in this category functions on the data dimensions. Also, the disadvantage of a higher-dimension dataset involves degraded quality, redundant data, and a higher module construct period, which makes information analyses very complex [7]. To solve such issues, feature selection is exploited as a major preprocessing stage to choose subsets of attributes from the enormous dataset. It increases the performance of classification models and clustering, which induces noisy, foreign, and ambiguous elimination of data [8]. The FS technique depends upon search methods and a performance evaluation of a subset. In the preprocessing phase, selecting features was indispensable to remove duplications, reducing the quantity of data and unnecessary and irrelevant features [9]. It has several methods for choosing the attribute that helps select the actual dataset as a potential feature. Recently, EA has shown itself as attractive and efficient in solving difficulties using optimizations [10].

This article introduces a new Big Data Analytics Assisted Arithmetic Optimization with Deep Learning Model for Sentiment Classification (BDA-AODLSC) approach. The presented BDA-AODLSC approach exploits BDA tools for sentiment classification. Initially, the BDA-AODLSC approach performs data preprocessing to transform it into a compatible format, and the TF-IDF method is utilized for the word embedding process. An Attention-based Long Short-Term Memory (ALSTM) method is utilized for classifying sentiments, and its hyperparameters can be selected by an Arithmetic Optimization Algorithm (AOA). For managing big data, the Hadoop MapReduce tool is employed. A far-reaching analysis is accomplished to demonstrate the superior accomplishment of the BDA-AODLSC method.

2. Literature Review

Hou et al. [11] presented an online client technique related to the ML approach for IoT unstructured BDA and utilized it in other BDA setups. Utilize online data given by the user to apply background Data Mining (DM), a parallel way for verifying its efficacy utilizing ML techniques like the K-Nearest Neighbour (KNN) approach. Chittam et al. [12] modelled big data storage for the data of the material science and its data processing variable for addressing the challenging task of data tabulation from scientific studies; DM approaches for retrieving the data from databases to execute BDA, and an ML predictive technique for determining material strength insights. Three techniques were presented depending on RF techniques, LR, and SVM. Such devices are tested and trained through a 10-fold cross-validation approach.

Huang et al. [13] explored the numerical modelling and simulation of online shopping client reliability depending on BDA and ML. This study primarily utilizes the ML clustering technique or simulating consumer loyalty. Call k-means collaborative mining technique depends on the Hash framework for executing DM on a multi-dimensional hierarchical tree of corporate credit risk, incessantly adjusting the supporting thresholds for various DM levels per particular needs and choosing effectual association rules until satisfactory outcomes were gained. Pegalajar et al. [14] formulated ML techniques for forecasting electricity demand. Particularly, this study was carried out through data from the Spanish Electricity Network. To this end, the author modelled the application of a set of ML approaches utilizing several structures. Specifically, the author applied 6 different predictive techniques: Multi-layer Perceptron, Linear Regression (LR), Gradient Boosting Regression, Random Forests (RF), three types of recurrent neural networks and Regression Trees.

Cui [15] established vigorous association rules among distinct surveillance parameters depending on old surveillance data in standard structural situations for forecasting whether the structural condition is standard. Also, this study constructed a system function module per actual requirements, gained the complete system structure, and applied the system function modules jointly with methods. Rudnichenko et al. [16] focused on the aspects of DSS for the ML techniques selection in big data withdrawal advancement. The study presented the outcomes of devising and utilizing a decision support system to evaluate ML approaches to solve DM problems.

3. The Proposed Model

This article presents a new BDA-AODLSC approach for accurate sentiment classification. This BDA-AODLSC approach exploits BDA tools for sentiment classification. Initially, the BDA-AODLSC technique performs data preprocessing to transform it into a compatible format, and the TF-IDF model is utilized for the word embedding procedure. For sentiment classification, the ALSTM model is utilized, and its hyperparameters can be selected by the AOA. Fig. 1 represents the workflow of the BDA-AODLSC technique.

3.1. Hadoop MapReduce

For managing big data, the Hadoop MapReduce (MR) tool is employed. As soon as the data size is increased, the conventional data analytics system has become increasingly complex to process and store an enormous quantity of information [17]. The best solution to manage the ample quantity of information is saving it in Hadoop file systems, viz., HDFS. The HDFS make the Hadoop work effectively due to its fault-tolerant model, which makes it reliable. HDFS is searched as a typical file mechanism which handles huge data sets. The default block size is 64 MB, which outperforms while handling a massive data set. The dataset in the HDFS is stored in the actual dataset and its metadata, namely file size, location, and so on. HDFS was a storage unit of Hadoop and followed master–slave framework. The master node encompasses 3 components named: the secondary name node, job tracker and name node, while the slave node encompasses the data node and task tracker.

MR allows data processing at a large scale. The structure of MR gives data processing in a distributed and parallel manner. Also, this makes sure that the network operates in a fault-tolerant method, which facilitates I/O scheduling. MR processes information in the value and key pair form. The mapping elements among two connected data items are called key-value pairs. Here, the key is an identifier which recognizes the value exclusively. The MR architecture encompasses a mapper phase that takes a raw dataset, sorts it out into key or value pairs, and reduces the phase which processes the data simultaneously. Also, it enables the presence of different mapper and reducer functions inside similar tasks. Likewise, the resultant from the reducer class could be provided as input to mapper classes. MR function is performed by programming languages like Python, Ruby, Java, etc. Now, the Java API is used for performing the MR function.

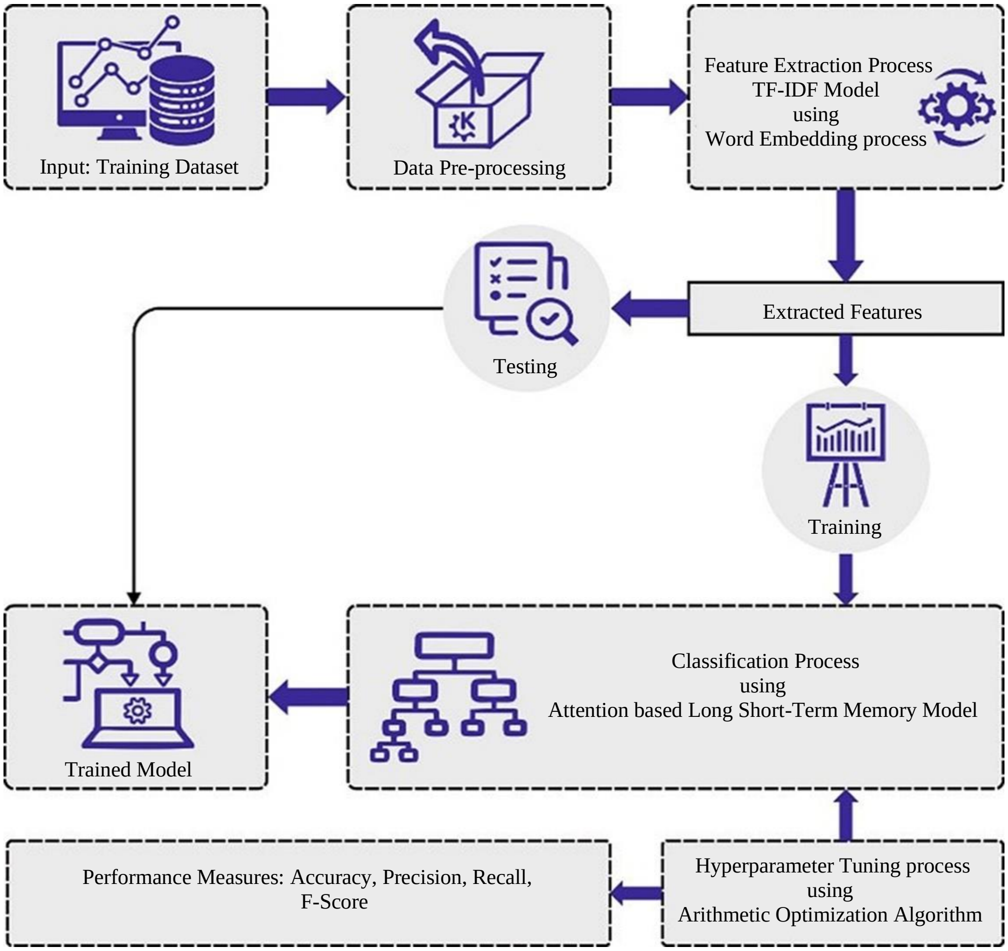
Input: Training Dataset
Data Pre-processing
Feature Extraction Process
TF-IDF Model
using
Word Embedding process
Extracted Features
Testing
Training
Classification Process
using
Attention based Long Short-Term Memory Model
Trained Model
Performance Measures: Accuracy, Precision, Recall, F-Score
Hyperparameter Tuning process
using
Arithmetic Optimization Algorithm

Fig. 1 Workflow of BDA-AODLSC model

3.2. Data Preprocessing and Word Embedding

The data preprocessing technique is used to eliminate incomplete and noisy data. Preprocessing has played a major role in optimizing classification performance [18]. In this study, the data used has an enormous quantity of redundant data that plays no part in the anticipation. The training and testing duration increases while the data set is bigger. Hence, eradicating redundant data might quicken the training process.

Preprocessing includes the steps performed for cleaning data such that the learning efficacy of the model is improved. The Natural Language Tool Kit (NLTK) of Python was implemented for these purposes. It is a collection of text-processing repositories that are exploited for different processing jobs, and the study applied NLTK 3.5b1 with Python 3.

- Removal of numerical value
- Removal of Missing value
- Conversion of Lowercase
- Stemming
- Punctuation mark removal

Next, the preprocessed data is fed as input to the TF-IDF method for deriving relevant feature vectors from a dataset. TF-IDF was an arithmetical measure proposing to demonstrate the word's significance from the file. It is performed as a weight element from text mining and data retrieval processes. The effectiveness of word was pointed with upgraded file; however, significance can be equally decreased as the corpus frequency when a word has a maximum frequency from a study, whereas a lower frequency from other research can be considered a keyword with discriminative ability. TF indicates the word numbers from the file:

$$TF = \frac{Overall\ words\ appearing\ from\ document}{Total\ words\ from\ the\ document} \tag{1}$$

IDF can be defined as a typical word. IDF is mainly implemented if the file has limited entries, and IDF is considered massive, and entry with optimum objectives.

$$IDF = \log\left(\frac{Total\ no.of\ documents\ in\ corpus}{Total\ no.\ of\ documents\ containing\ the\ term\ +\ 1}\right) \tag{2}$$

The composition of TF and IDF was termed TF-IDF:

$$TF\text{-}IDF\ = TF \times IDF \tag{3}$$

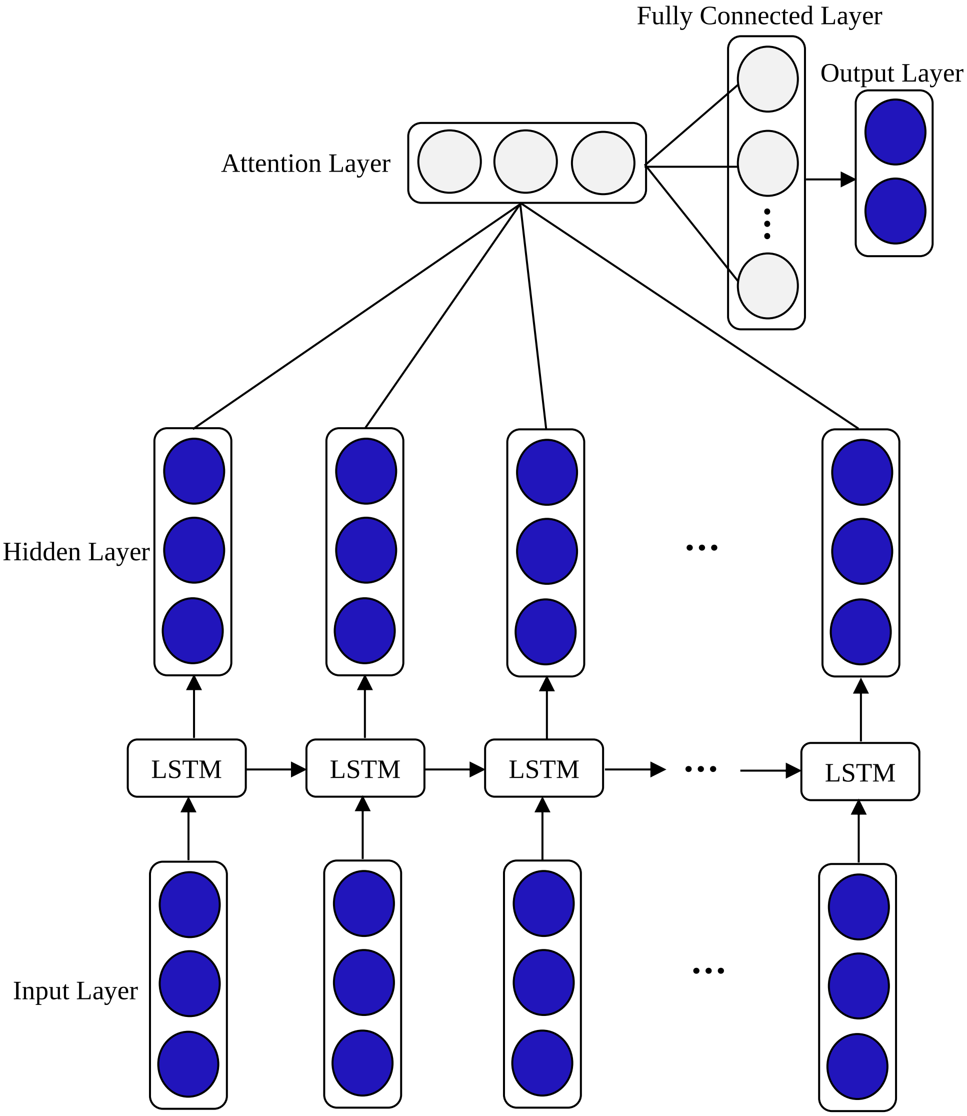

Fig. 2 Structure of ALSTM

Hence, the TF-IDF measure can be directly and inversely proportional to the word frequencies exhibited in the document and from the entire corpus.

3.3. Sentiment Classification by Implementing ALSTM Model

The ALSTM method can be exploited to classify the sentiments properly. CNN is well-developed for accommodating and converting a single image to vector representation [19]. RNN has been exploited for detecting images according to the factor duration; however, in certain instances, it was seen as untrustworthy in real-time due to gradient vanishing across long-term windows during the backpropagation stage of the gradient. Then, the LSTM method is exploited to enhance the model efficacy, which could assist in eliminating the abovementioned problems by swapping hidden units by memory cells. Fig. 2 defines the infrastructure of ALSTM. Such function is named time dispersed layer for weight modifications and internal state building, usually performed by backpropagation method. This implies that adding a layer subsequently leads to numerous applications of the similar layer and a sequence of image "feature" or "interpretation" to function on the LSTM model.

(i) Block input (b_g) in

$$b_g = \tanh\left(W_b * X_g + U_b * out_{g-1} + d_b\right) \tag{4}$$

(ii) Input gate (i_g) in

$$i_g = \sigma(W_j * X_g + U_j * out_{g-1} + d_i \tag{5}$$

(iii) Forget gate (f_g) in

$$f_g = \sigma(W_f * X_g + U_f * out_{g-1} + d_f \tag{6}$$

(iv) Memory state (m_g) in

$$m_g = i_g \odot Z_g + f_g \odot m_{g-1} \tag{7}$$

(v) Output gate (O_g) in

$$O_g = \sigma\left(W_o * X_g + U_b * out_{g-1} + d_o\right) \tag{8}$$

(vi) and hidden state (h_g) in

$$h_g = O_g \odot \tanh\left(C_g\right) \tag{9}$$

Purposing at the challenge that the features could not be efficiently emphasized during the procedure of text classifiers, the presented system generates an LSTM text classifier system dependent upon an attention process that concentrates on the data of text data and enhances the appearance capability of text features [20]. During this method, the word effect weighted was defined by the relationship betwixt the output h_t of all the Hidden Layers (HLs) and context vector s.

The computation of the attention process is recognized in 2 steps:
Step1. Compute the attention distribution on every input data, specifically, proceeds the context vector s and the outcome h_t of HL as input units, arrive at single-layer perceptron, and gain an understood representation u_t of outcome with computation.

$$u_t = \tanh\left(\alpha h_t + \beta s\right) \tag{10}$$

whereas α and β denote the weighted matrix; h_t implies the resultant of HL; s represents the query vector. Next, θ_t has attained with softmax operation that is computed as:

$$\theta_t = \text{softmax}\left(u_t\right) \tag{11}$$

whereas the probability vector collected of θ_t refers to the attention distribution of words.

Step2. Compute the weighted sum of input data based on θ_t attention distribution, specifically the attention distribution θ_t stands for the correlation betwixt the time t data from inputted data vector H and query s if the query θ_t was provided.

The input data was summarized by weight summation to obtain the attention values. The particular computation as:

$$S = \sum_{t=1}^{N} \theta_t \, h_t \tag{12}$$

Text classifier method dependent upon attention-oriented LSTM utilizes softmax as a resultant layer for normalized computation, and integrated with cross-entropy loss function, the main function was demonstrated as:

$$Loss \; = -\sum_{i=1}^{K} Y_j \; \log\left(y_j\right) \tag{13}$$

In which K signifies the text counts from the corpus, y_j implies the real probability distributing vector of the present text classifier, y_j denotes the likelihood distribution vectors of the present text predicting with the classifier method, and the dimensional vectors are equivalent to the count of classifier labels.

3.4. Parameter Tuning utilizing AOA

Lastly, the hyperparameter optimization of the ALSTM algorithm takes place with the help of AOA. The candidate solution set is randomly generated with a population-oriented method initiated by the enhancement technique [21]. The optimization rule set gradually increased the solution set generated where the main function assesses it. For certain problems, the global optimization technique attains probability. The algorithm comprises two main categories: exploitation and exploration. In the exploration phase, the latter is the improved accuracy of the attained solution. Based on the AOA, the subsequent subsections describe intensification (exploitation) and diversification (exploration). Multiplication, division, subtraction, and addition are the foremost arithmetical operators. Together with analysis, algebra, and geometry, the most important section of current mathematics can be a building block of a number system named arithmetics. From the candidate solution set, the better fundamentals subjected to the specific criterion can be defined by the AOA as mathematical optimization.

Eq. (14) demonstrates candidate solution set (Y). In all the iterations, the best-attained solution is considered the optimal candidate solution.

$$Y = \begin{bmatrix} y_{1,1} & \cdots & \cdots & y_{l_i i} & y_{l,m-1} & y_{l_i m} \\ y_{2,1} & \cdots & \cdots & y_{2,i} & y_{2,m-1} & y_{2,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_{M-1,1} & \cdots & \cdots & y_{M-l,i} & \cdots & y_{M-l,m} \\ y_{M1} & \cdots & \cdots & y_{M,i} & y_{M,m-1} & y_{M,m} \end{bmatrix} \tag{14}$$

The search phase is chosen beforehand the AOA initializes. The Math Optimizer Accelerated (M_{OA}) function is calculated as follows:

$$\begin{aligned} M_{OA}(c\_iteration) & \\ &= Minimum \\ &+ c\_iteration \\ &\times \left(\frac{\text{Max}imum - \text{Min}imum}{M\_iteration}\right) \end{aligned} \tag{15}$$

In the t^{th} iteration, the function value is represented by $M_{OA}(c\_iteration)$ thus, maximal and minimal iteration is evaluated. In this section, The AOA of exploration or diversification behaviours are introduced. The highest distributed value is attained by division or multiplication operators based on arithmetical operators. The AOA exploration operator can define a better solution based on multiplication and division. The arithmetical operator behaviours are based on the simple rule.

$$y_{j,k}(c\_iteration+1) = \begin{cases} B(y_i) \div (M_{OA}+\delta) \times \left((U_i-L_i)\times\alpha+L_i\right) R_2 < 0.5 \\ B(y_i) \times M_{OA} \times \left((U_i-L_i)\times\alpha+L_i\right) otherwise \end{cases} \quad (16)$$

Following, the j^{ih} and k^{th} solution position is represented by $y_{j_{i,k}}$. The smaller integer was δ, with the controlling variable was α. The j^{th} location of upper and lower boundaries is U_i and L_i.

$$M_{OA}(c\_-iteration) = 1 - \frac{c\_iteration1/\beta}{M\_iteration1/\beta} \quad (17)$$

Where $M_{OA}(c_{-iteration})$ denotes the function value at P^h iteration. Furthermore, the sensitive variable is β. High dense output is attained by the addition or subtraction of arithmetical operators. Due to lower dispersion, subtraction and addition are simply targeted approaches. Afterwards, a small iteration deduces near-optimum solutions. Eq. (18) defines the exploitation phase.

$$y_{\mathrm{j},k}(c\_iteration+1) = \begin{cases} B(y_i) - M_{OA} \times \left((U_i-L_i)\times\alpha+L_i\right) & R_3 < 0.5 \\ B(y_i) \times M_{OA} \times \left((U_i-L_i)\times\alpha+L_i\right) & otherwise \end{cases} \quad (18)$$

Apply the searching operators of exploitation to prevent being stuck in the local optima. R_1, R_2, and R_3 are randomly generated values within [0, 1]. The optimum solution can be attained by helping the exploitation search phase. Algorithm 1 defines the AOA pseudocode.

Algorithm 1: AOA pseudocode

```
Input: Initialized AOA parameter with the maximal iteration amount
Output: Attain the better solution
Whereas (c_iteration < M_iteration) do
Assess the fitness function
Attain the optimum solution
Upgrade the value of $M_{OA}$
For ($j = 1$ $to$ $s_{olution}$) do
    For ($j = 1$ $to$ $s_{\mathrm{olution}}$) do
        $R_1$, Rand $R_3$ are randomly generated values within [0,1]
        If $R_1 > M_{OA}$
            Then
            Diversification phase
            If $R_2 > 0.5$
            Then
            The division math operator is employed
            Upgrades the $j^{th}$ solution location
            Else
            The multiplication math operator was upgraded to the $k^{th}$ solution location
                End If
            Else
            Intensification phase
                If $R_3 > 0.5$
                Then
                The subtraction math operator was employed
                Upgrades the $j^{rh}$ solution location
                Else
                The addition math operator is employed
                Upgrades the $k^{th}$ solution location
                End If
            End If
            End For
            End For
             (c_iteration < c_iteration +1
End While
```

The fitness selection is an essential aspect of the AOA method. Solution encoding is employed to assess the aptitude (goodness) of the candidate solution. Presently, the value of accuracy is the major state employed for fitness function design.

$$Fitness = \max(P) \quad (19)$$

$$P = \frac{TP}{TP+FP} \quad (20)$$

From the above equations, TP and FP represent the true and false positive values.

4. Experimental Validation

The investigational validation of the BDA-AODLSC method is examined by the dataset [22] comprising 64,295 files consisting of attributes that include 'Sentiments', 'App', and 'Translated_ Reviews'. There were 5158 neutral reviews, 8271 negative reviews, and 23,998 positive reviews in the database.

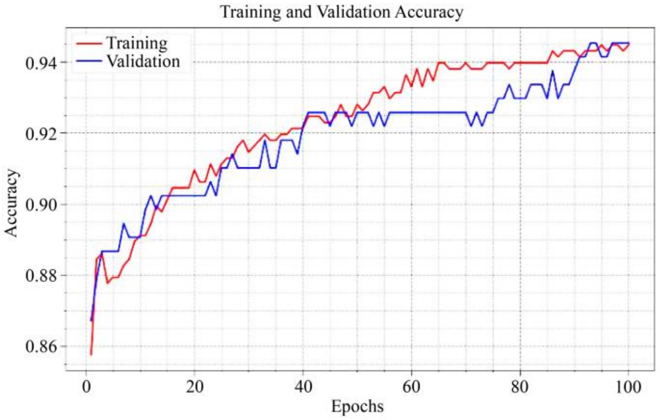

Fig. 3 TACY value and VACY value evaluation of the BDA-AODLSC model

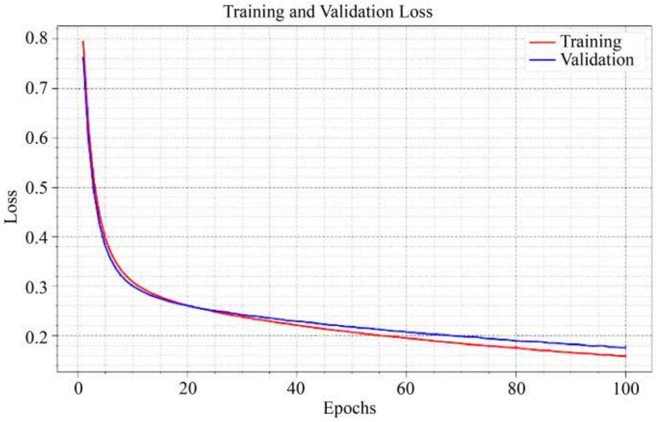

Fig. 4 TLOS value and VLOS value evaluation of the BDA-AODLSC model

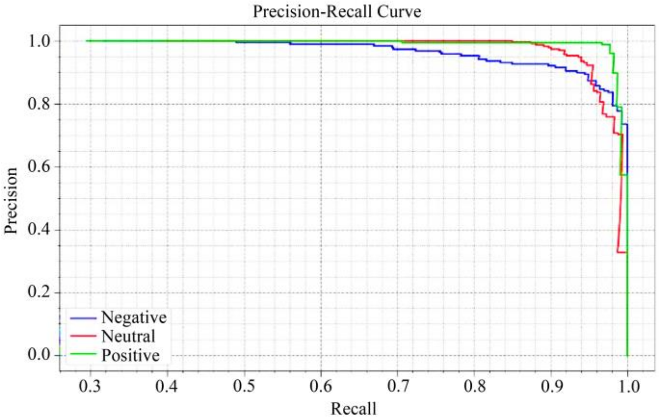

Fig. 5 Precision-Recall evaluation of the BDA-AODLSC model

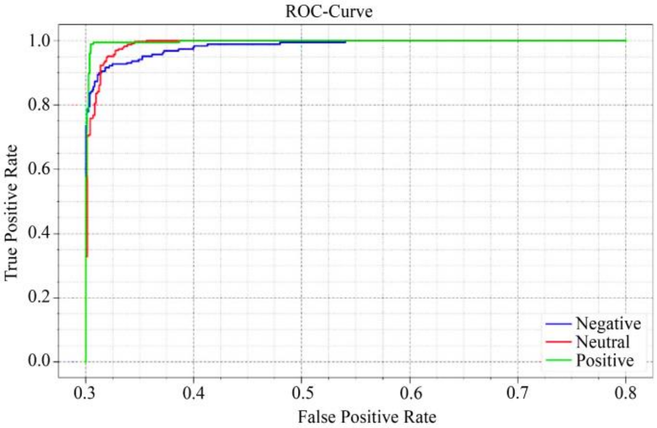

Fig. 6 ROC evaluation of the BDA-AODLSC model

The TACY and VACY values of the BDA-AODLSC technique are examined on big data accomplishment in Fig. 3. The figure exhibited that the BDA-AODLSC technique has superior accomplishment with maximum TACY and VACY values. Particularly, the BDA-AODLSC approach has obtained the highest TACY results.

The TLOS and VLOS values of the BDA-AODLSC method are tested on big data accomplishment in Fig. 4. The figure exhibited that the BDA-AODLSC method has revealed superior accomplishment with reduced TLOS and VLOS values. It is evident that the BDA-AODLSC technique has the least VLOS results.

A precise Precision-Recall investigation of the BDA-AODLSC technique under the test dataset is illustrated in Fig. 5. The figure symbolized that the BDA-AODLSC technique has improved Precision-Recall values in every class label. The elaborated ROC investigation of the BDA-AODLSC method under the test dataset is illustrated in Fig. 6. The figure symbolized the BDA-AODLSC method has exposed its capacity for classifying discrete class labels.

In Table 1 and Fig. 7, a complete sentiment classification of the BDA-AODLSC approach is examined in terms of $prec_n$ [18, 23]. The experimental outcomes represented that the BDA-AODLSC approach gains increasing values of $prec_n$ under three labels. For instance, with a negative label, the BDA-AODLSC technique attains a higher $prec_n$ of 97.64% while the BBSO-FCM, Gradient Boosted SVM (GB-SVM), Gradient Boosting Tree (GBT), Support Vector Machine (SVM), LR, and RF methods achieve less $prec_n$ of 96.61%, 91.68%, 91.41%, 88.63%, 90.49%, and 86.44% respectively.

Furthermore, with the neutral label, the BDA-AODLSC technique attains a higher $prec_n$ of 90.3%, while the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF methods accomplish lower $prec_n$ of 89.66%, 84.41%, 79.43%, 83.62%, 84.52%, and 78.49% correspondingly. Finally, with the positive label, the BDA-AODLSC method reaches a greater $prec_n$ of 97.44%, whereas the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF methods accomplish a lesser $prec_n$ of 96.67%, 93.4%, 94.48%, 92.59%, 87.41%, and 88.53% correspondingly.

Table 1. Sentiment classifier outcome of BDA-AODLSC approach concerning $prec_n$

Precision (%)				
Methods	**Negative Label**	**Neutral Label**	**Positive Label**	**Average**
BDA-AODLSC	97.64	90.3	97.44	95.13
BBSO-FCM	96.61	89.66	96.67	94.31
GB-SVM	91.68	84.41	93.4	89.83
GBT	91.41	79.43	94.48	88.44
SVM	88.63	83.62	92.59	88.28
LR	90.49	84.52	87.41	87.47
RF	86.44	78.49	88.53	84.49

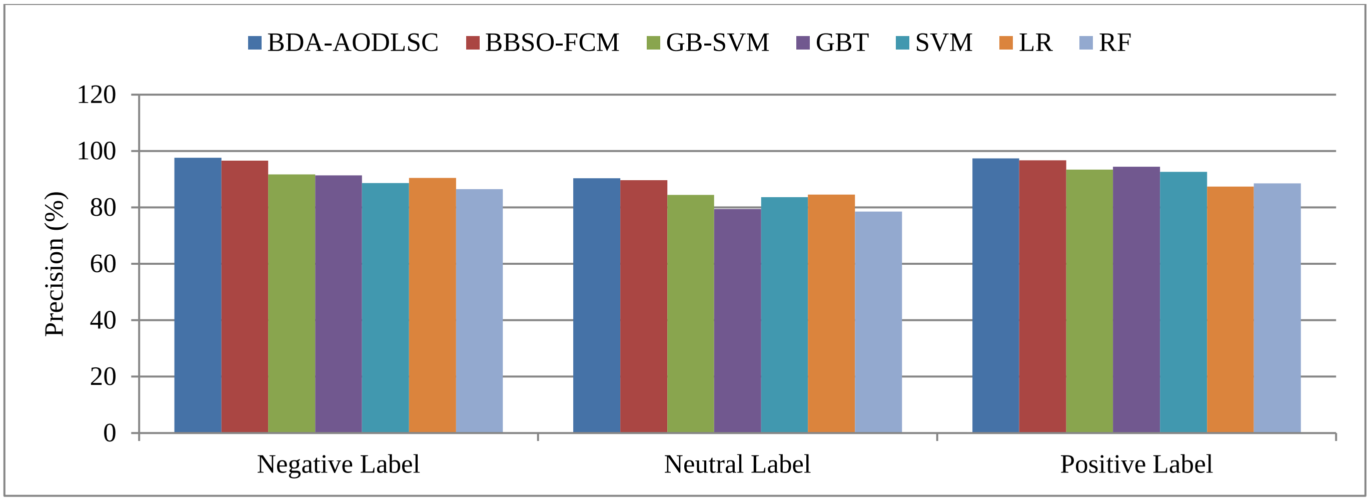

Fig. 7 $\boldsymbol{Prec_n}$ **analysis of the BDA-AODLSC approach with distinct class labels**

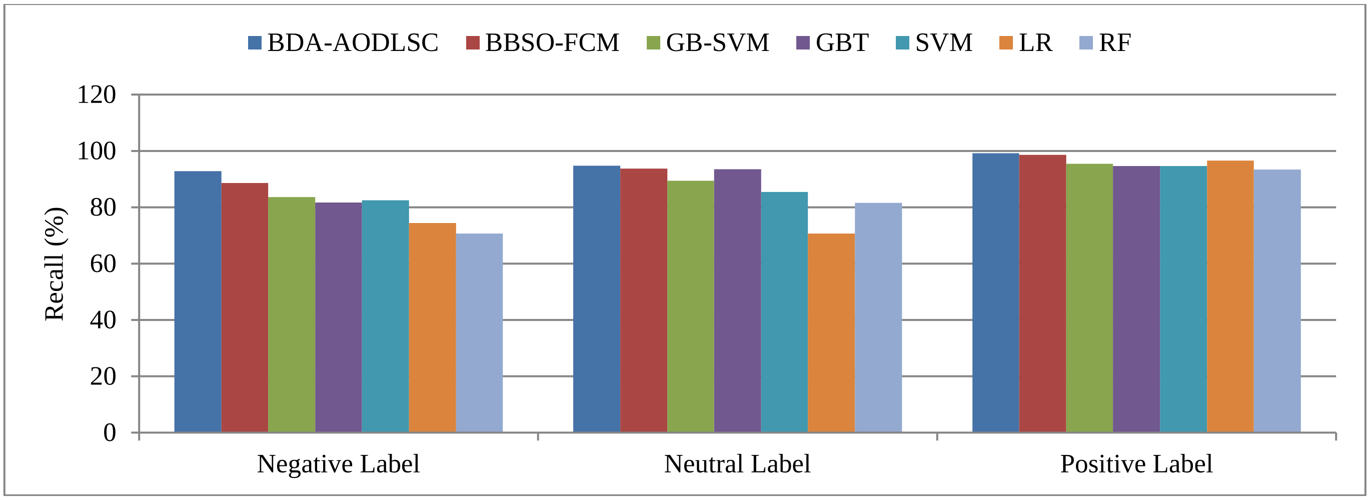

Fig. 8 $\boldsymbol{Reca_l}$ **analysis of the BDA-AODLSC approach with distinct class labels**

In Table 2 and Fig. 8, a complete sentiment classification of the BDA-AODLSC technique is examined in terms of $reca_l$. The experimental outcomes characterized that the BDA-AODLSC technique gains increasing values of $reca_l$ under three labels. For example, with a negative label, the BDA-AODLSC model attains a higher $reca_l$ of 92.79%, while the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF models achieve lower $reca_l$ of 88.60%, 83.58%, 81.68%, 82.45%, 74.44%, and 70.66% correspondingly. Additionally, with the neutral label, the BDA-AODLSC method reaches the highest $reca_l$ of 94.72%, while the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF method achieves lower $reca_l$ of 93.70%, 89.45%, 93.54%, 85.43%, 70.63%, and 81.58% correspondingly. Lastly, with the positive label, the BDA-AODLSC technique achieves the highest $reca_l$ of 99.14%, while the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF techniques accomplish lower $reca_l$ of 98.64%, 95.45%, 94.68%, 94.65%, 96.56%, and 93.41% correspondingly.

Table 2. Sentiment classifier outcome of BDA-AODLSC approach concerning $\boldsymbol{reca_l}$

Recall (%)				
Methods	**Negative Label**	**Neutral Label**	**Positive Label**	**Average**
BDA-AODLSC	92.79	94.72	99.14	95.55
BBSO-FCM	88.60	93.70	98.64	93.65
GB-SVM	83.58	89.45	95.45	89.49
GBT	81.68	93.54	94.68	89.97
SVM	82.45	85.43	94.68	87.52
LR	74.44	70.63	96.56	80.54
RF	70.66	81.58	93.41	81.88

Table 3. Sentiment classifier outcome of BDA-AODLSC approach concerning $F1_{score}$

F1-Score (%)				
Methods	**Negative Label**	**Neutral Label**	**Positive Label**	**Average**
BDA-AODLSC	93.23	93.77	98.36	95.12
BBSO-FCM	91.59	90.42	97.42	93.14
GB-SVM	87.58	86.43	94.5	89.50
GBT	86.44	85.5	94.52	88.82
SVM	85.64	84.62	93.56	87.94
LR	81.49	76.56	91.52	83.19
RF	77.65	80.48	91.49	83.21

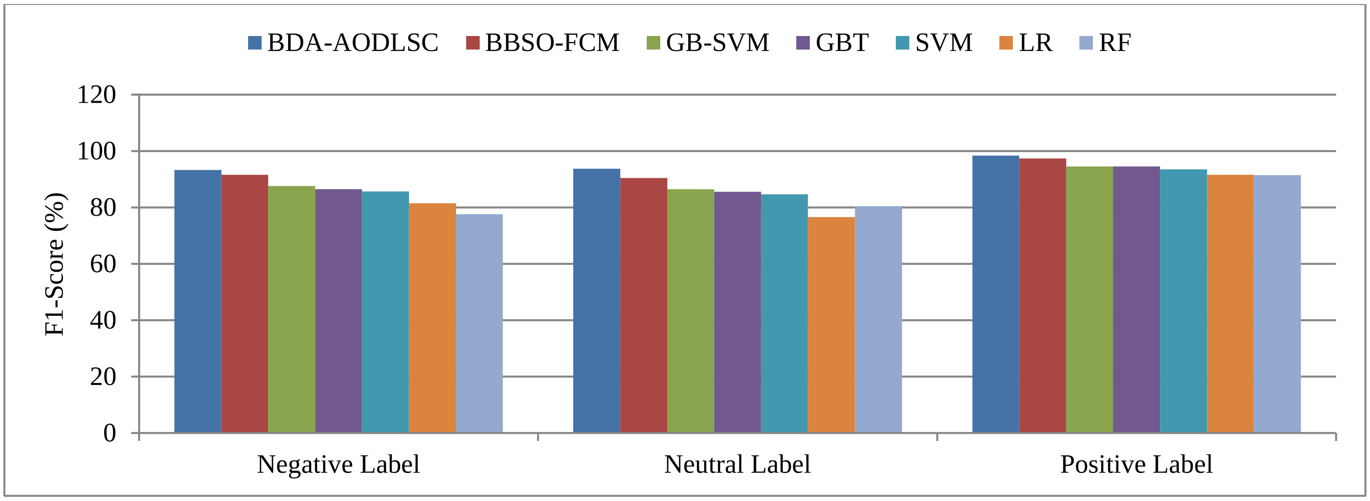

Fig. 9 $F1_{score}$ analysis of the BDA-AODLSC approach with distinct class labels

In Table 3 and Fig. 9, a complete sentiment classification of the BDA-AODLSC model is examined with regard to $F1_{score}$. The investigational outcomes symbolized that the BDA-AODLSC model gains increasing values of $F1_{score}$ under three labels. For example, with a negative label, the BDA-AODLSC method obtains a higher $F1_{score}$ of 93.23%, whereas the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF methods reach a lesser $F1_{score}$ of 91.59%, 87.58%, 86.44%, 85.64%, 81.49%, and 77.65% correspondingly. Moreover, with the neutral label, the BDA-AODLSC technique achieves a higher $F1_{score}$ of 93.77%, whereas the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF techniques accomplish a lesser $F1_{score}$ of 90.42%, 86.43%, 85.5%, 84.62%, 76.56%, and 80.48% correspondingly. Lastly, with the positive label, the BDA-AODLSC model reaches a higher $F1_{score}$ of 98.36%, whereas the BBSO-FCM, GB-SVM, GBT, SVM, LR, and RF models achieve lower $F1_{score}$ of 97.42%, 94.5%, 94.52%, 93.56%, 91.52%, and 91.49% correspondingly.

Finally, a comprehensive comparative $accu_y$ investigation of the BDA-AODLSC approach with current models are given in Table 4 and Fig. 10. The outcomes inferred that the R-NB-KNN approach results in a lower $accu_y$ of 72.64%. Meanwhile, the Stochastic GDC-LR (SGDC-LR), Graph Convolution Network (GCN), Simple GCN (SGCN), and Neural Attention BoE (NA-BoE) models obtain certainly improved $accu_y$ of 88.47%, 88.45%, 89.01%, and 86.53% respectively.

Concurrently, the BBSO-FCM and GB-SVM model accomplishes closer $accu_y$ of 95.85% and 92.99%. But the BDA-AODLSC approach gains a maximum $accu_y$ of 97.32%. Hence, this BDA-AODLSC approach can be implemented for precise sentiment classification.

Table 4. Comparative evaluation of the BDA-AODLSC model with recent approaches

Methods	Accuracy (%)
BDA-AODLSC	97.32
BBSO-FCM	95.85
GB-SVM	92.99
SGDC-LR	88.47
R-NB-KNN	72.64
GCN	88.45
SGCN	89.01
NA-BoE	86.53

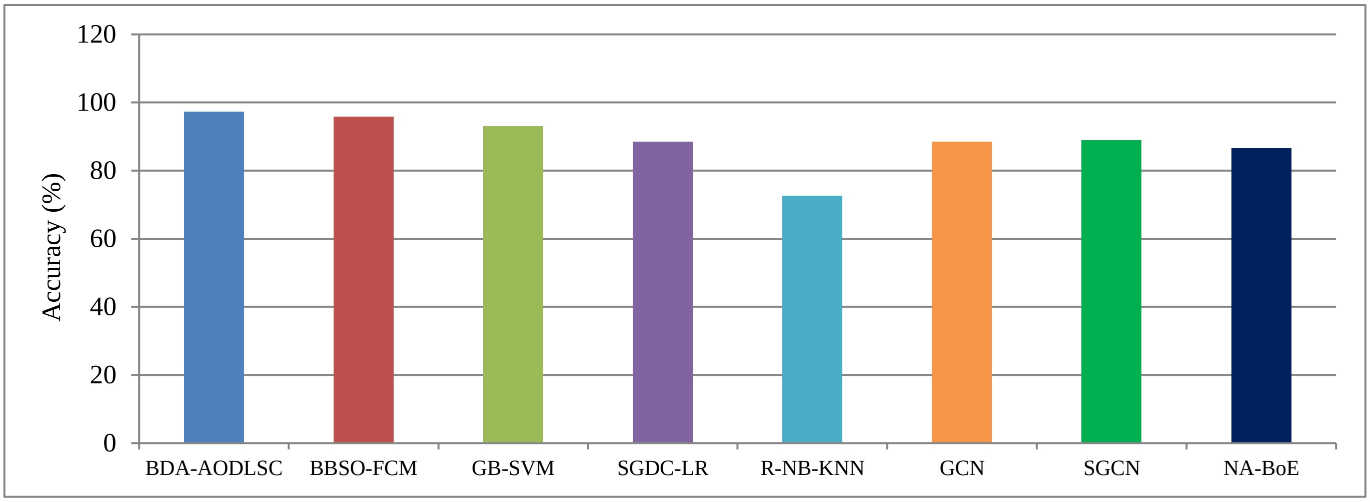

Fig. 10 $Accu_y$ analysis of BDA-AODLSC technique with recent approaches

5. Conclusion

This article presents a new BDA-AODLSC methodology for accurate sentiment classification. This BDA-AODLSC methodology exploits BDA tools for sentiment classification. Initially, the BDA-AODLSC technique performs data preprocessing to transform it into a compatible format, and the TF-IDF method is utilized for the word embedding process. For sentiment classification, the ALSTM method is utilized, and its hyperparameters can be selected by the AOA.

For managing big data, the Hadoop MapReduce tool is employed. A far-reaching analysis has been accomplished to reveal the superior accomplishment of the BDA-AODLSC technique. The extensive results exhibited the significant accomplishment of the BDA-AODLSC method over other existing methodologies. In the future, the accomplishment of the BDA-AODLSC method can be enhanced by data clustering models.

References

[1] N.P. Jayasri, and R. Aruna, “Big Data Analytics in Health Care by Data Mining and Classification Techniques,” *ICT Express*, vol. 8, no. 2, pp. 250-257, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Shan-Ju Yeh,Tsun-Yung Yeh, and Bor-Sen Chen, “Systems Drug Discovery for Diffuse Large B Cell Lymphoma Based on Pathogenic Molecular Mechanism via Big Data Mining and Deep Learning Method,” *International Journal of Molecular Sciences*, vol. 23, no. 12, pp. 1-22, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] S. Ejaz Ahmed, “Statistical and Machine-Learning Data Mining: Techniques for Better Predictive Modelling and Analysis of Big Data,” *Technometrics*, vol. 63, no. 2, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[4] Ania Cravero et al., “Challenges to Use Machine Learning in Agricultural Big Data: A Systematic Literature Review,” *Agronomy*, vol. 12, no. 3, pp. 1-34, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Hajra Waheed et al., “Predicting Academic Performance of Students from VLE Big Data using Deep Learning Models,” *Computers in Human Behavior*, vol. 104, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] David Soriano-Valdez et al., “The Basics of Data, Big Data, and Machine Learning in Clinical Practice,” *Clinical Rheumatology*, vol. 40, no. 1, pp. 11-23, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Usha Moorthy, and Usha Devi Gandhi, “A Survey of Big Data Analytics using Machine Learning Algorithms,” *Research Anthology on Big Data Analytics, Architectures, and Applications*, IGI Global, pp. 655-677, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Shalini Ramanathan, and Mohan Ramasundaram, “Accurate Computation: COVID-19 rRT-PCR Positive Test Dataset Using Stages Classification through Textual Big Data Mining with Machine Learning,” *The Journal of Supercomputing*, vol. 77, no. 7, pp. 7074-7088, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] Snigdha Sen et al., “Astronomical Big Data Processing Using Machine Learning: A Comprehensive Review,” *Experimental Astronomy*, vol. 53, pp. 1-43, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Justin Zuopeng Zhang et al., “Big Data Analytics and Machine Learning: A Retrospective Overview and Bibliometric Analysis,” *Expert Systems with Applications*, vol. 184, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] Rui Hou et al., “Unstructured Big Data Analysis Algorithm and Simulation of Internet of Things based on Machine Learning,” *Neural Computing and Applications*, vol. 32, no. 10, pp. 5399-5407, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Swetha Chittam et al., “Big Data Mining and Classification of Intelligent Material Science Data using Machine Learning,” *Applied*

Sciences, vol. 11, no. 18, pp. 1-17, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jui-Chan Huang et al., "Statistical Modeling and Simulation of Online Shopping Customer Loyalty based on Machine Learning and Big Data Analysis," *Security and Communication Networks*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] M.C. Pegalajar et al., "Analysis and Enhanced Prediction of the Spanish Electricity Network through Big Data and Machine Learning Techniques," *International Journal of Approximate Reasoning*, vol. 133, pp. 48-59, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Moyang Cui, "Big Data Medical Behavior Analysis based on Machine Learning and Wireless Sensors," *Neural Computing and Applications*, vol. 34, no. 12, pp. 9413-9427, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Nicolay Rudnichenko et al., "Decision Support System for the Machine Learning Methods Selection in Big Data Mining," *CMIS*, 2020. [Google Scholar] [Publisher Link]

[17] Quan Zou, Guoqing Li, and Wenyang Yu, "Mapreduce Functions to Remote Sensing Distributed Data Processing—Global Vegetation Drought Monitoring as Example," *Software: Practice and Experience*, vol. 48, no. 7, pp. 1352-1367, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] Deepak Kumar Jain et al., "An Intelligent Cognitive-Inspired Computing with Big Data Analytics Framework for Sentiment Analysis and Classification," *Information Processing & Management*, vol. 59, no. 1, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Monika Sethi et al., "Classification of Alzheimer's Disease using Gaussian-Based Bayesian Parameter Optimization for Deep Convolutional LSTM Network," *Computational and Mathematical Methods in Medicine*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Wanli Luo, and Lei Zhang, "Question Text Classification Method of Tourism Based on Deep Learning Model," *Wireless Communications and Mobile Computing*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] N. Deepa, and S.P. Chokkalingam, "Optimization of VGG16 Utilizing the Arithmetic Optimization Algorithm for Early Detection of Alzheimer's Disease," *Biomedical Signal Processing and Control*, vol. 74, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22] [Online]. Available: https://www.kaggle.com/lava18/google-playstore-apps

[23] Madiha Khalid et al., "GBSVM: Sentiment Classification from Unstructured Reviews using Ensemble Classifier," *Applied Sciences*, vol. 10, no. 8, pp. 1-20, 2020. [CrossRef] [Google Scholar] [Publisher Link]