v1: 9 September 2024

Research Article

Comparative Analysis of Machine and Deep Learning Techniques for Text Classification with Emphasis on Data Preprocessing

Peer-approved: 9 September 2024

Qeios, Vol. 6 (2024)
ISSN: 2632-3834

Saikat Gochhait[1]

1. Symbiosis Institute of Digital and Telecom Management, India

Physician-written discharge medical notes include vital details regarding their patients' health. Numerous deep learning algorithms have shown effectiveness in gleaning crucial insights from unstructured medical notes data, leading to potentially useful outcomes in the medical field. The goal of this research is to determine how different deep learning algorithms perform as models for text classification issues in long short-term memory (LSTM). The Titanic Disaster Dataset has been used for pre-processing, which is essential since there is a lot of unnecessary information in textual data. Next, clean up the data by eliminating duplicate rows and filling in the blanks. Besides traditional machine learning algorithms such as naive Bayes (NB), gradient boosting (GB), and support vector machine (SVM), we use deep learning algorithms to classify data, including bidirectional – LSTM using Conditional Random Fields (CRFs). BiLSTM is the most precise model compared to other models and baseline research, with a classification accuracy of 98.5%.

Corresponding author: Dr Saikat Gochhait, saikat.gochhait@sidtm.edu.in

1. Introduction

Discharge summaries and other unstructured medical notes are important health records that provide extensive clinical data regarding individuals' illnesses. It's possible that some disease-related details aren't included in the structured data fields. This method can be used to classify and categorize unstructured text. Because natural language processing (NLP) can classify and organize any text to provide insightful analysis and answers, it is often considered one of the most potent language processing techniques [1]. Thanks to natural language processing (NLP) and other machine learning approaches, computers are now able to read and comprehend text on par with human comprehension. A two-layered recurrent neural network (RNN) and hidden-layer long short-term memory (LSTM) model with significant bias may be able to predict depression from text in order to help avoid mental illness and suicide ideation [2]. Despite text's potential riches as a source of information, its disorganization makes it very difficult to mine for facts. Text classification is made easier by machine learning and natural language processing.

Text classifiers can classify, organize, and arrange a variety of textual materials, such as files, documents, medical research, and web content. Almost 80% of all data is composed of text, making it one of the most

doi.org/10.32388/XHC9J1.2

common forms of unstructured data [3]. Because text data involves a number of challenges (analysis, interpretation, organization, and sorting), businesses often struggle to use it successfully. Finding relevant information in a text requires reading through its paragraphs before classifying it. In this situation, machine learning and text classification are useful [4]. Various text types, including emails, formal documents, social media postings, consumer surveys, and others, can be effectively arranged using text classifiers. Thanks to technologies like this, businesses can now make decisions based on factual information, optimize operational procedures, and spend less time examining text data. Tools for text analysis are used by many businesses [5]. Text analysis tools allow organizations to organize large amounts of data in seconds instead of days. This covers email, chat, social media, documents, help inquiries, and more. With additional resources at their disposal, the author focused on the most important projects. Robotics deep learning applications greatly aid in solving the most critical issues that machine learning is unable to resolve [6]. Due to their low-level processing and engineering requirements, deep learning systems were highly suitable for the task of text categorization and were capable of achieving exceptionally high levels of accuracy[7]. Two of the most widely utilized deep learning architectures for text categorization are recurrent neural networks (RNNs) and convolutional neural networks (CNNs), which are used to classify data [8]. The purpose of RNNs is to capture temporal dependencies in sequential data. On the other hand, CNNs use convolutional filters to extract patterns from grid-like data, such as photographs. It's common knowledge that different algorithms are employed in different processes. Using numerous approaches simultaneously to examine large volumes of data is akin to the decision-making process of the brain [9]. Compared to standard machine learning algorithms, deep learning approaches require a significantly larger amount of training data. They are not limited by the amount of information they can learn from training data, in contrast to most ML algorithms, to address the gap present study proposes with the Titanic dataset.

The individual passenger survival statuses on the Titanic are described in the titanic and titanic2 data sets. Text classification is necessary to advance in the language descriptions of different languages. Using a combination of grouping tactics and machine learning techniques such as Hadoop MapReduce and naive Bayes classifiers could perhaps be the most efficient approach to achieve this goal.

doi.org/10.32388/XHC9J1.2

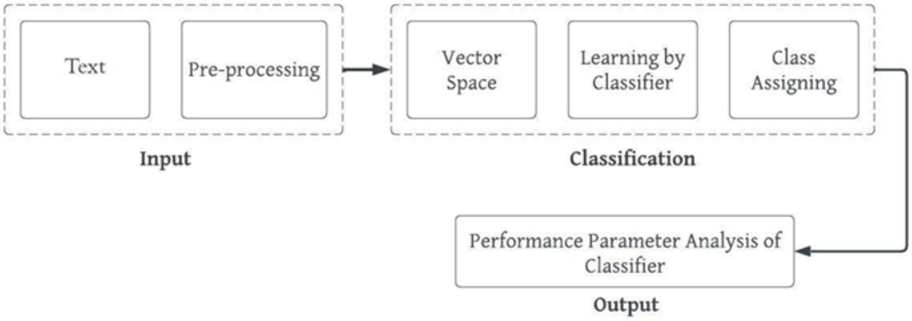


Figure 1. Foundational process for textual categorization

In performance testing, deep learning outperforms Gaussian naive Bayes by a significant margin of 72%. One problem with Naive Bayes is that it treats categorical data as though it were completely unrelated and gives no probability to any of the qualities. This study makes contributions in the following manners:

a. Provide a productive method for classifying textual material that makes use of the Long Short Term Memory algorithm.
b. We select the best classifier by evaluating the classification performance using a variety of machine learning and deep learning techniques.
c. The results show that BiLSTM works effectively since it has several hidden layers, retains important information while discarding irrelevant information, and has a substantial accuracy of 92% on our dataset.

Machine learning, deep learning, and comparative analysis are utilised to effectively filter and forecast a large amount of textual data. Figure 1 shows the architecture for text classification, which illustrates the process of creating text classification.

Initially, we handle each sequence of characters in the input text that comes after a period as a single sentence. The second step uses noise reduction and computerized comprehension to perform preliminary text processing. Is tokenization—which merges three processes into one—used alone or in conjunction with other strategies to minimize noise? A stemming approach lowers word noise by reverting words to their original forms. When a word ends in "ing," for example, it will revert to its original form instead of the stemmed form. Subsequently, the word vector is transformed into a frequency-determined space vector by the architecture. Learning a classifier, assessing its parameters, and creating a feature matrix to label the classes are the next steps. This architecture uses deep learning and machine learning techniques to assess the categorization models. We accomplish this by selecting classifiers and evaluating their performance with a variety of machine learning and deep learning techniques.

According to our results, the proposed BiLSTM is accurate and 98.5% accurate because it keeps important data while deleting irrelevant data.

2. Related Work

Examination of Large Data Text categorization is taking a critical turn thanks to Hadoop-based methods. Once the training data is generated, a specific label is assigned to one of the classes by the learning algorithm. According to this description, the training stage is the more scientific one, and it frequently entails gathering a modest amount of data that significantly influences the classification stage [10]. Consequently, Hadoop Map Reduce encounters certain constraints concerning the classification of text with the help of machine language. The Naive Bayes classifier is developed and applied using this technique. The NB model classifies documents that are found in text by treating the document as an event and determining the likelihood that specific words won't appear. The model might be a Bernoulli or multinomial model. In text classification, the naïve Bayes classifier operates and progresses in accordance with its naïve assumptions. Conditional independence can be measured using a semi-NB technique when determining the model's probability [11]. In contrast to more conventional approaches, Bayesian procedures provide a novel and well-structured way to combine knowledge with primary data. Numerous other techniques, like the support vector machine and the concealed Markov model, could be modified by applying the Bayesian NB classifier. It is demonstrated how to distinguish between several Distributed Denial of Service (DDOS) attacks using a multinomial classifier [12]. Application-layer denial-of-service (DDOS) attacks are explained, along with a way to counteract hostile website visitors that employs a polynomial distributed model to halt and terminate the attack. The multinomial event models have been compared in numerous studies, which has improved the Naive Bayes algorithm [13]. Smoothing strategies are employed when specific words within a document are either ignored or removed. A multinomial model was utilized for the analysis of brief texts in this investigation of improvement methods. Recent work has shown that this linear technique could be able to sidestep the curse of multidimensionality and provide near-real-time performance. Science and technology related to mechanical equipment have come a long way in the last few decades [14]. These defects cause changes in the vibration and noise patterns of the gadget. Making sure that the datasets that professionals use to develop commercial NLP systems are noise-free might be difficult [15]. Although BERT has been demonstrated to work reasonably well on noisy text, there is still room for improvement in terms of performance. This can be achieved by decreasing noise in the datasets and optimizing BERT for certain commercial use cases. Generally speaking, the kind and quantity of noise in the data might affect how well BERT and other language models function on noisy text. Practitioners can utilize methods like data cleaning, data enrichment, and data normalization to reduce noise in datasets. Eliminating or fixing typographical or grammatical problems from data is known as data cleaning. New data is created when existing data is altered, such as by changing words or adding synonyms. We call this procedure "data augmentation." Data normalization is the process of standardizing the data by converting it into a common format or by removing information that is not needed to use transformers to combine text and tabular (categorical and numerical) data for application areas [16]. Transformers' high computational cost is one of their main drawbacks, particularly for lengthy text sequences. From the Transformer's conception by Vaswani et al., it has been effectively used for numerous NLP tasks [17]. It is easy to download several pre-trained models because the toolkit integrates with Hugging Face's current API, which includes tokenization and the model hub [18]. Many NLU initiatives have found success with data augmentation, particularly those with sparse data. Data Boost is an easy and effective way to add data to text. This is accomplished through the use of conditional generation and reinforcement learning [19]. Sending and receiving private papers containing personal information securely requires attention to automatic text anonymization. An outline of the currently in-use techniques is provided, along with some basic information about text anonymization [20]. Two different but connected fields —natural language processing and privacy-aware data publishing—have given rise to the development of anonymization techniques. Due to concerns about intellectual property rights and the drive to maximize profits, the private sector is infamously reticent to share the work it produces. As a result, they rarely make their annotation systems and toolkits accessible to the broader public [21]. Despite recent reports of success with text-to-text models, it may be difficult to determine where the science stops and starts with these models. On the Clinical TempEval 2016 relation extraction test, the most logical option for output representations—where relations are grouped into fundamental predicate logic statements—did not produce satisfactory results. [22]

The best methods work on an individual event basis and yield similar results to paired temporal connection extraction methods. Text categorization algorithms have been used to investigate the Titanic dataset. Efficacy is obviously deficient. To overcome this restriction, the authors suggest classifying textual data with deep learning [23]. Food classification and nutrient profiling are challenging, expensive, and time-consuming activities due to the enormous number of items and labels in major food composition databases, as well as the constant change in the food supply. This study looked at the prediction results of models that took bag-of-words and structured nutrition information as inputs. Furthermore, it used pre-trained language models and supervised machine learning to predict food categories and nutrition quality scores using human-validated and coded data. The unstructured language from food labels was translated into lower-dimensional vector representations using a modified pretrained sentence model, Bidirectional Encoder Representations, built from the Transformers model.

Then, multiclass classification and regression tasks were done using supervised machine learning methods[24]. This study aims to convert Amazon product reviews into modern standard Arabic in order to produce a fair collection of Bahraini dialects that includes product feedback and many other dialects using deep learning LSTM. These datasets have different linguistic characteristics. The accuracy, F1 score, and AUC metrics were used in numerous tests to assess the model's performance using train-validate-test split and k-fold cross-validation. With an augmentation approach, the model's average accuracy across all datasets ranges from 96.72% to 97.04%; its F1 score ranges from 97.91% to 97.93%; and its AUC ranges from 98.46% to 98.7% [25]. In order to assign a semantic class label to every pixel, this work suggests a novel scene parsing method that considers the relationship between part-whole hierarchies and the final feature map. Scene parsing has recently been greatly impacted by deep learning-based methods. However, these techniques were unable to retain the spatial information of high-level (or mid-level) elements. Hinton, one of the pioneers of deep learning, proposed the idea of encoding pose information, such as orientation, in a capsule. A parent capsule is created by grouping all of the capsules with similar pose matrix values [26]. This work proposes a novel method for learning perceptual grouping of the retrieved features from the convolutional neural network (CNN) to display the visual structure. CNN does not take into account the spatial hierarchy between the high-level properties. The perceptual grouping of features is used to achieve this. The Modified Guided Co-occurrence Block (mGCoB) is suggested as a tool for taking intra-relationships between feature maps into consideration [27]. The suggested modality adaptation takes into account the complementary and shared information of each modality while maintaining its ability to distinguish between items in the label space. We used two different multimodal apps to test the proposed methodology. "Multi-view object detection" and "RGBD image semantic segmentation" were two of these. The results show that the suggested modality adaptation approach is a useful way to combine and transmit knowledge [28].

3. Proposed Model Framework

Data scientists and machine learners are familiar with the Titanic Disaster Dataset, which contains information about passengers of the ship. The dataset includes parameters such as gender, age, cabin number, and fare, as well as statistics about the passengers' survival rates.

The proposed framework is used to investigate patterns and relationships among the passengers and to predict their survival outcomes based on this information. Survivors' gender, cabin number, names, and ages are all potentially useful predictors of survival, as they could be related to factors such as proximity to lifeboats, social status, or physical strength. Following the data import, we purify the data, impute missing values and information, select the most valuable categorical features (i.e., "pitch"), and divide the data into training and testing sets to predict the accuracy.

Machine learning methodologies are employed to approximate the probabilities. We can see the percentage of each sex that has survived by using age categories [29]. We extract this information from the dataset to see how many people survived according to gender, and we find that the number of male survivors is substantially lower than the number of female survivors by 3 and 8%, most likely because males prioritise the protection of their families and children above everything else. Figure 2 depicts the survival report of different age and gender.

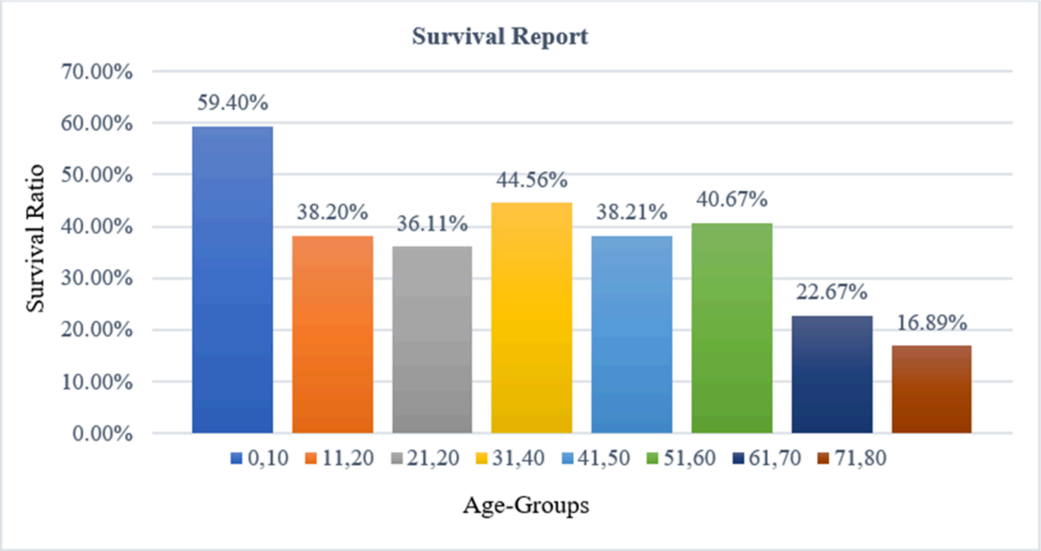


Figure 2. Dataset-survival report

Here are some instances of how text analysis can be done using standard machine learning techniques: The process of classifying a text involves placing it into one or more predetermined categories. Determining a text's sentiment—whether it be positive, negative, or neutral—is the goal of sentiment analysis. Identified Organization The process of locating and categorizing identified entities—such as individuals, groups, and places—in a text is known as recognition. Finding the underlying subjects in a collection of text documents is known as topic modeling. Following data cleansing, support vector machine imputations are applied to complete the absent values or information that remain blank. During the second phase of the pipeline design, the process involved feature engineering and the removal of superfluous features.

To make the data more computationally efficient and less susceptible to noise, the next step was normalization. At the end of the process, we chose a model that effectively generalized our data set. An array of deep and ML models was used for this task. In order to determine the likelihood of survival or mortality among members of distinct groups, the likelihood probability and class label probability associated with each feature supplied during the training phase are utilised.

Natural language processing (NLP) applications frequently employ BiLSTMs, a form of neural network architecture, to perform tasks including sentiment analysis, machine translation, and sequence labelling. It is a variant of the LSTM architecture in which two distinct hidden layers process the input sequence in both the forward and reverse orientations. Bi-LSTMs can extract contextual information from both previous and subsequent items in a sequence; they are frequently utilized in NLP applications.

This makes it particularly effective for tasks such as named entity recognition, where it is important to understand the context in which a word appears in order to correctly identify whether it is a named entity or not.

Computational Algorithm illustrates a suggested approach model that accepts as input information about survivors, including passenger class, sex, age, and gender.

Step 1: Label Encode

- Label encode the data using some encoding method to convert categorical labels into numerical values. This is common when working with machine learning algorithms that require numerical input.

Step 2: Data Normalization

- Calculate the mean (Ψ) of the data (Y^n) and normalize the data by subtracting the mean. This step aims to center the data around zero.

Step 3: Subtract Mean from Data

- Subtract the mean (Ψ) from each data point (Y).

doi.org/10.32388/XHC9J1.2

Step 4: Calculate the Mean of Squares

- Calculate the mean of the squared values (μ^2) of the normalized data.

Step 5: Assign Mean of Squares to Data

- Replace the original data (Y) with the calculated mean of squared values (μ^2).

Step 6: Matrix Convergence

- Convert a matrix (Cf) to a NumPy array $(C2)$.

Step 7-9: Randomization

- For each element $(Z(ln))$ in a range defined by the length of Ln, initialize it with a random value multiplied by the square root of $\frac{2}{y(l_n-1)}$, where y is the value of $l_n - 1$.

Step 10: Vector Conversion (Maxpooling)

- Apply Maxpooling to a vector $(vector)$ to get a converted vector (M).

Step 11: BiLSTM Layer

- Apply a BiLSTM layer to the converted vector (M).

Step 12: Hidden Layer

- Obtain a hidden layer $(fBiLSTM)$ from the BiLSTM layer.

Step 13: Dense Layer (Class Prediction)

- Predict classes using a dense layer (PrC) based on the output from the BiLSTM layer.

Step 14-18: Class Comparison and Return

- Compare the predicted classes (PrC) with a test set (x_test) and return the predicted class if it matches; otherwise, return the corresponding element from the test set.

Step 21: Output

- The process stops.

Figure 3 depicts the proposed system technique for data cleansing. Feature selection comes first, followed by missing value imputation, followed by the next step to remove unnecessary data to reduce storage costs and the risk of data leaks under data protection laws like the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA). Failure to properly destroy data may result in the revealing of sensitive information, data breaches, and legal ramifications. To successfully remove data, use secure deletion processes to prevent the information from being recovered. Feature selection is the process of picking a subset of relevant features or variables from a larger set of input characteristics for use as inputs in a machine learning model. The purpose of feature selection is to improve model performance by lowering the number of input features and eliminating those that are redundant or unnecessary.

To demonstrate the efficacy of both DL and ML algorithms for classification, we first choose the relevant variables, such as the number of survivors in groups and sex, and then divide the data and run the algorithms on separate subsets. Consequently, it requires fewer inputs. The CRFs algorithm attains an accuracy of up to 80% on the Titanic dataset by selectively incorporating only pertinent information into its prediction models.

The recurrent weighted average method we utilise also includes learning algorithms, which can evolve and improve as more data comes in. This is just one example of how machine learning techniques can improve upon the results obtained using an appropriate method.

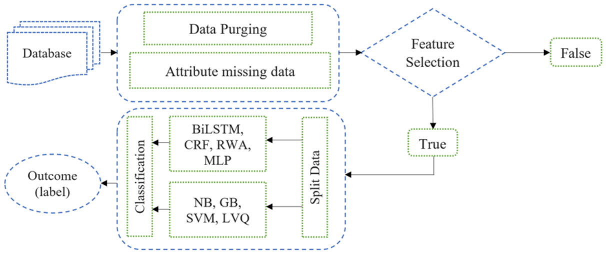


Figure 3. Proposed system classification architecture

4. Experimental Results and Discussions

It is vital to highlight that the superiority of one machine learning model over another is determined by the individual job, data type, and problem characteristics. Each algorithm has strengths and drawbacks, and the optimum model is determined by the specific use case's requirements and limits. Here are some reasons why BiLSTM models may be selected over NB, GB, and SVM [30]

Sequential Data and Temporal Dependencies: BiLSTMs are particularly well-suited for tasks involving sequential data, where the order of elements matters, and capturing temporal dependencies is crucial.

Complex Non-linear Relationships: BiLSTMs, being a type of RNN, can model complex non-linear relationships in data. This makes them suitable for tasks where the underlying patterns are intricate and may not be captured effectively by linear models like Naive Bayes or SVM.

Variable-Length Sequences: BiLSTMs can handle variable-length sequences, which is advantageous for tasks where input data may have varying lengths, such as text of different lengths in natural language processing.

Feature Learning: BiLSTMs autonomously acquire hierarchical representations of features from the data, hence minimising the necessity for manual feature engineering. This can be beneficial when handling data that has a large number of dimensions and is intricate.

Contextual Understanding: BiLSTMs have the ability to capture context from both past and future elements in a sequence, allowing them to understand the context and dependencies between words or data points.

Ultimately, the optimal model is determined by the individual needs, data characteristics, and job goals. This can be useful for activities when context is important. Table 1 shows how effective the suggested approaches are in contrast to other classifiers. BiLSTM includes a memory function that allows it to recollect the data sequence. However, text data contains a lot of unnecessary information that the BiLSTM can remove to save computation time and money, so there's an additional benefit. BiLSTM is ideal for text categorization because it can reject irrelevant input while maintaining the order of events.

Learning type	Techniques	Classification Accuracy
Deep Learning	BiLSTM	98.51%
	CRF	91.15%
	RWA	93.80%
Machine Learning	NB	92.23%
	GB	95.67%
	SVM	96.44%

Table 1. Classification: Accuracy of machine and deep learning techniques

As a result of only taking categorical characteristics and discarding extraneous information, RWA is one of the tools for nonlinear statistical data modeling, as 81% accuracy is achieved on this dataset. In addition to SVM, GB, and NB, various machine learning techniques were applied to this dataset. Figure 4 depicts the different approaches in DL and ML to determine the classification accuracy metrics.

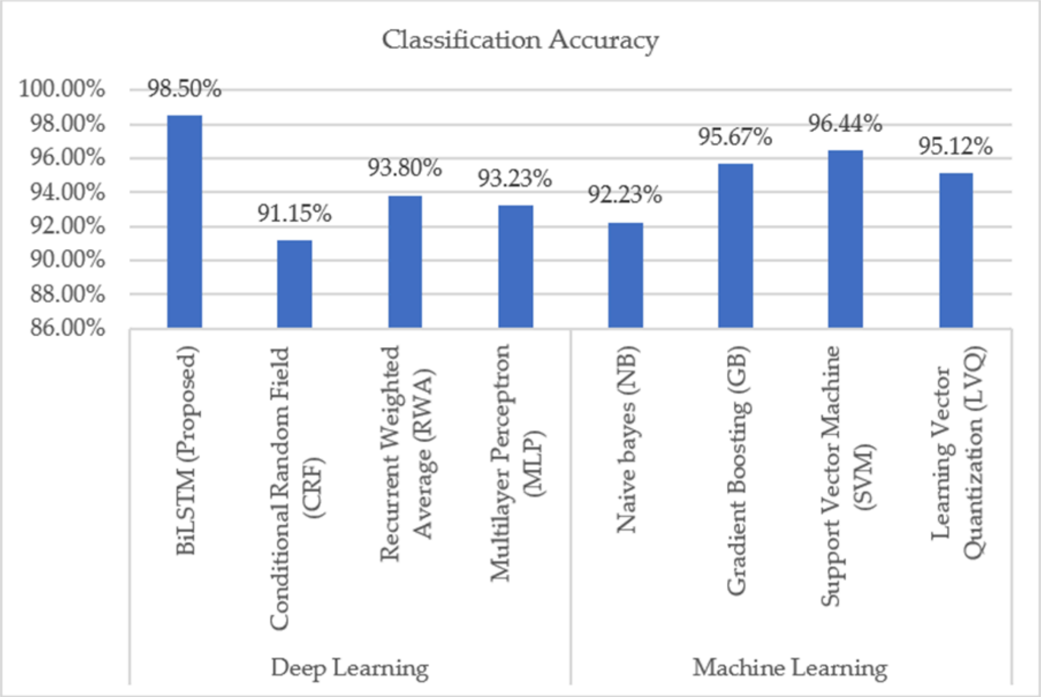


Figure 4. Various DL and ML approaches' classification accuracy metrics

Figure 5 illustrates the efficiency of machine learning models. Pre-processing data improves machine learning model performance, as demonstrated. It is clear that NB has an accuracy of 92.23%. Because NB predicts the output of the categorical dependent variable using a range of independent factors, whereas GB is particularly sensitive to noisy data and calculating the distance between each data point is quite expensive, GB achieves a high accuracy of 95.67%. LVQ obtains 95.12% accuracy, which is higher than GB; however, it is found to be biased for categorical characteristics. There is very little to chance with it. Machine learning approaches outperform base paper findings, and SVM performs well in our situation, with 96.44% accuracy, because our dataset contains independent values and we fill the missing values in distinct classes.

doi.org/10.32388/XHC9J1.2

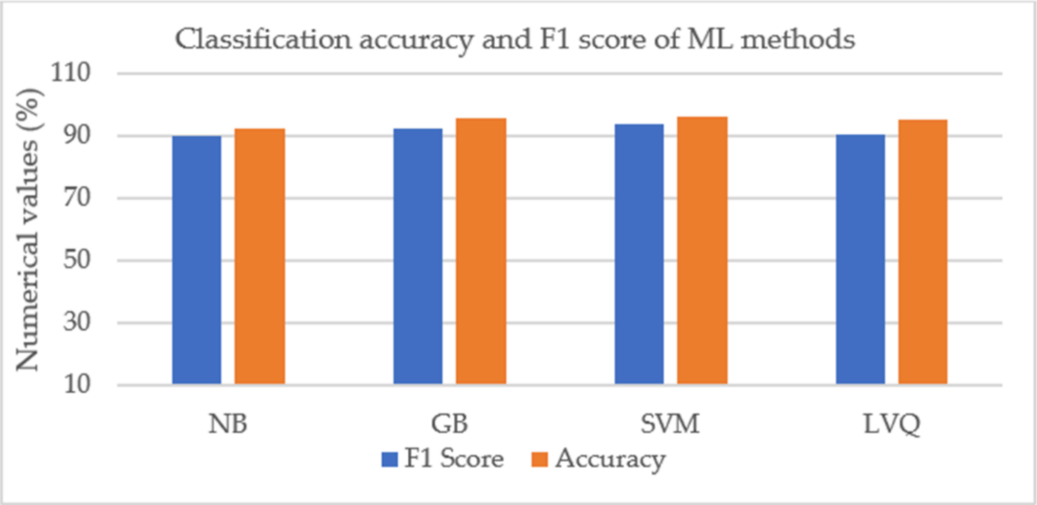


Figure 5. F1 score and accuracy value of ML methods

doi.org/10.32388/XHC9J1.2

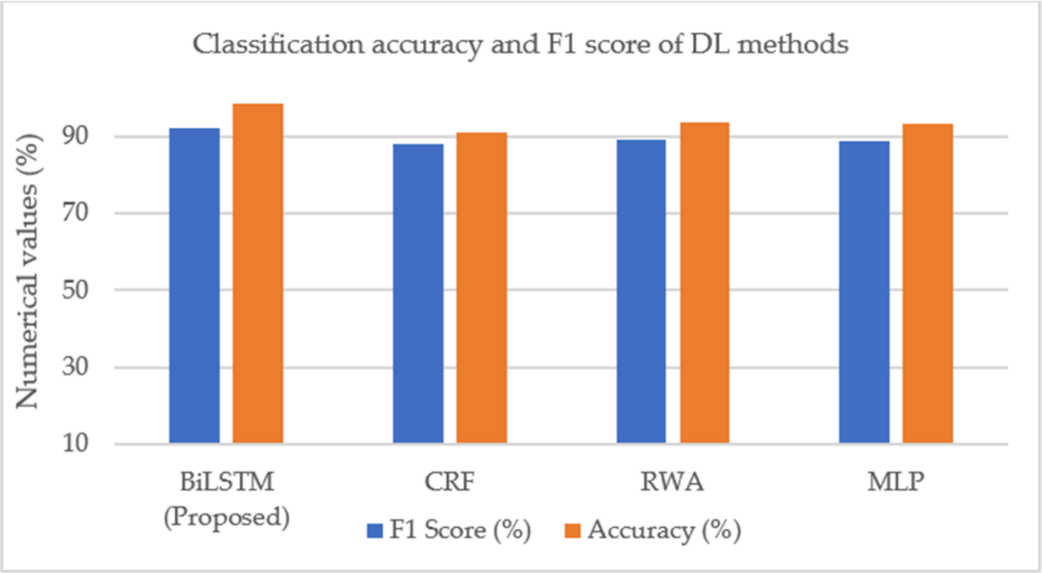


Figure 6. F1 score and accuracy value of DL methods

The effectiveness of deep learning models is depicted in Figure 6. RWA has an accuracy rating of 93.81%. The CRF achieves 91.15% accuracy, which is lower than that of the BiLSTM. This is because the BiLSTM treats the output at each time step and cell memory as distinct notions. At each time step, the output and hidden state of CRF are identical. BiLSTM works well on this data, with an accuracy of 98.51%, because of its discriminative nature and the inclusion of a feature that helps it to recall the data sequence.

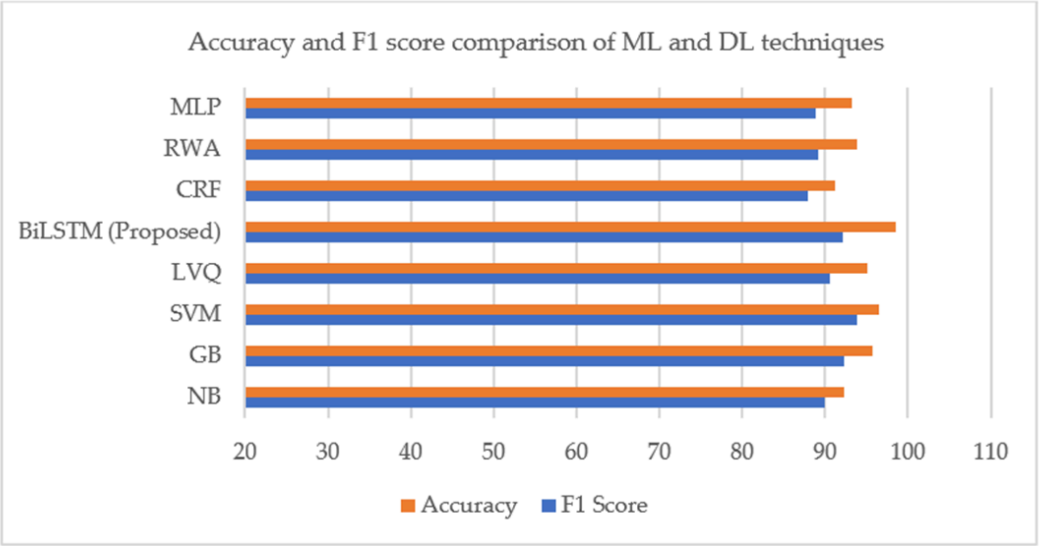
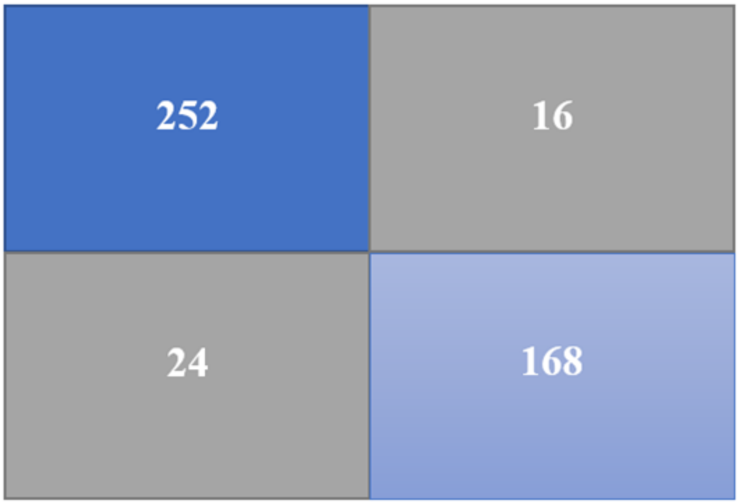


Figure 7. Comparing the F1 score and classification accuracy of ML and DL approaches

This could enable the BiLSTM to learn some latent sequence attributes that are not directly related to items in the sequence.

Text data has a great deal of superfluous information, which the BiLSTM may be able to eliminate in order to save time and resources on calculations. This is another benefit that it possesses, and it is the ability to work toward the removal of useless information. With an F1 score of 92.12%, which represents the predictive performance of a model by combining precision and recall, BiLSTM outperforms all other machine and deep learning models on this data (see Figure 7). The F1 score is the harmonic mean of these two metrics.



252	16
24	168

Figure 8. BiLSTM confusion matrix

doi.org/10.32388/XHC9J1.2

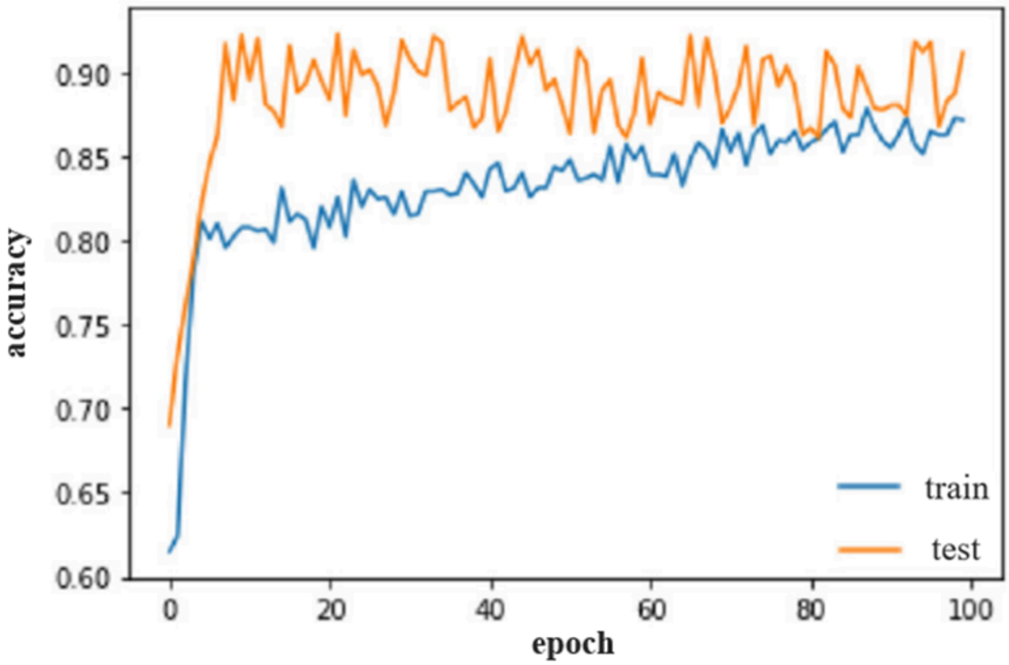


Figure 9. BiLSTM classification accuracy curve

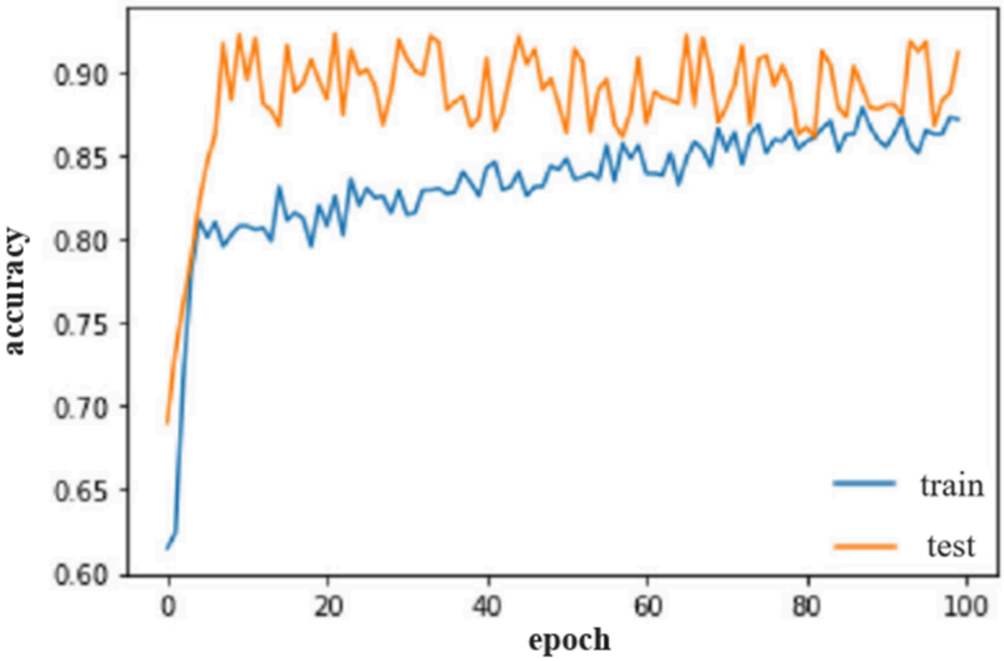


Figure 10. Classification: Accuracy of BiLSTM with early stopping

Therefore, the BiLSTM is a great tool for text classification because of its capacity to eliminate unnecessary information while simultaneously retaining the order in which events occurred. Figure 8 depicts the BiLSTM confusion matrix, showing that 252 persons have been appropriately identified as survivors. Then, FP and FN reflect the number of unexpected values, while TN, the 168th element of the diagonal, indicates human casualties. Figure 9 depicts the BiLSTM accuracy curve, revealing that the proposed model reaches 98.51% accuracy at epoch 10, and thereafter it oscillates up to epoch 100; hence, we use early stopping on our model, as shown in Figure 10, which clearly demonstrates how our model performs on textual data.

doi.org/10.32388/XHC9J1.2

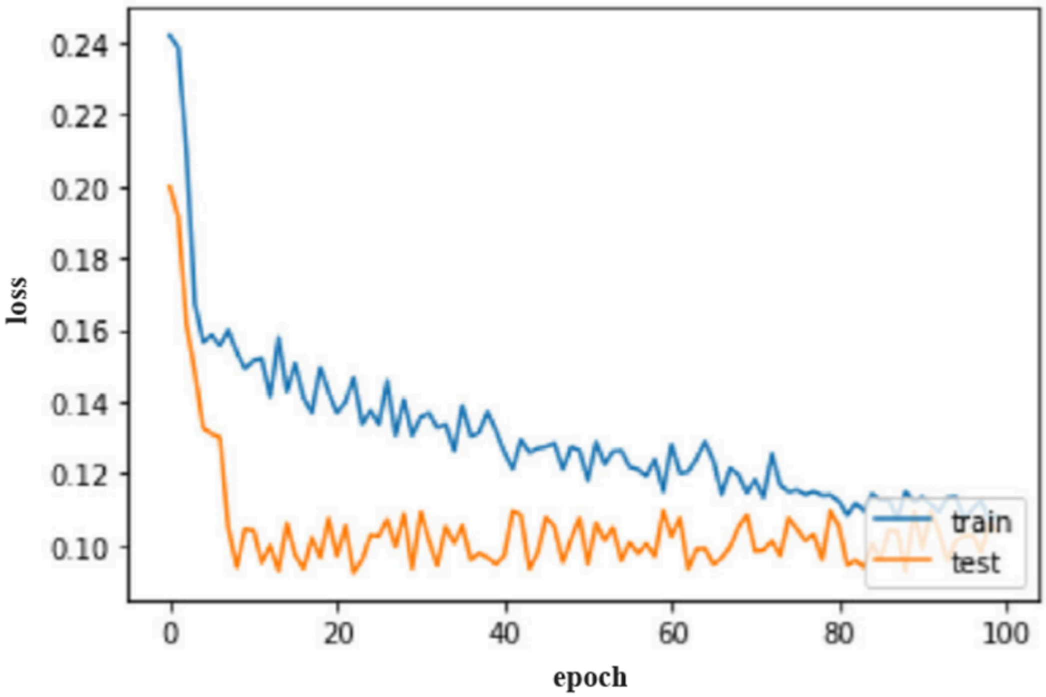


Figure 11. BiLSTM loss curve

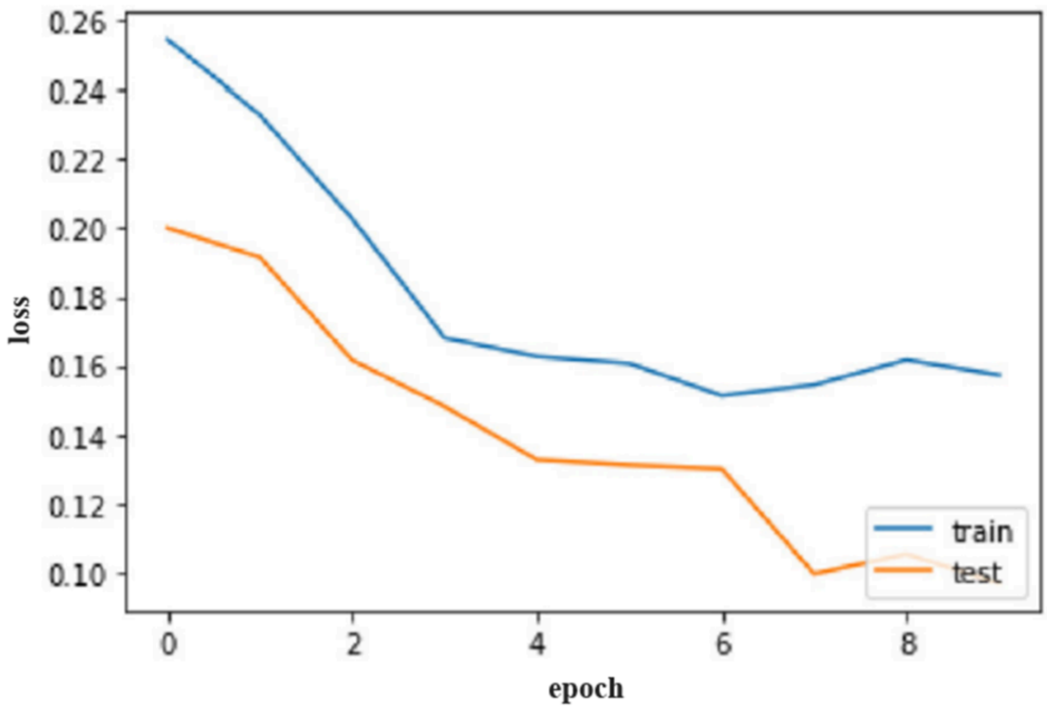


Figure 12. BiLSTM loss curve - early stopping

Curve of loss severity: We use early stopping on our model since Figure 12 clearly shows the loss of accuracy on our data, and Figure 11 of LSTM shows that our model loses 16% at epoch 10 and then oscillates up to 100.

5. Conclusion

In recent years, the quantity of textual data that is accessible has increased substantially. With such a large volume of data, it becomes challenging to effectively sort through it and extract meaningful information. Therefore, it is crucial to find efficient methods for quickly sorting through vast amounts of data. This organized data, often referred to as information, is then used for strategic planning in various industries, such as commerce and manufacturing. In this particular study, we proposed a method based on BiLSTM for categorizing textual data. When classifying texts, they employed a range of machine learning and deep learning techniques. The BiLSTM model, specifically designed for textual information, has the ability to remember the order of presentations. This characteristic is particularly useful when dealing with sequential data, such as sentences or paragraphs. One of the advantages of using BiLSTM is its capability to discard irrelevant or unnecessary information while still retaining the order of the remaining information. By doing so, it helps filter out noise or redundant data, which can be beneficial for classification tasks and other text-based applications. The present study applied the Titanic Disaster Dataset, a well-known dataset used to forecast the survival rates of Titanic passengers, to test our suggested approach. Leveraging the BiLSTM model's capability to eliminate superfluous data while maintaining the hierarchy, we attain an improved reliability of 98.51% on this dataset. This high accuracy demonstrates the effectiveness of the BiLSTM-based approach for classification tasks and highlights its potential as a powerful tool for working with text-based data. Overall, the study indicates that by leveraging the model's capacity to maintain order and weed out extraneous information, using BiLSTM-based techniques can help in effectively classifying textual data. This could lead to increased precision and dependability in jobs involving categorization as well as other text-based studies across a range of fields.

6. Future Work

In the future, advancements in algorithms for deep learning, with the help of BERT (Bidirectional Encoder

Representations from Transformers), are expected to enable the automatic classification of data. BERT is considered a cutting-edge system that utilizes a multilayer transformer and attention techniques, making it highly parallelizable. One of the key features of BERT is its bidirectionality, which means it considers both the preceding and following words when processing a given word in a sentence. This bidirectional approach allows BERT to capture contextual information effectively and understand the relationships between words. In addition to bidirectionality, BERT incorporates a multilayer transformer architecture. Transformers are neural network models that excel in detecting long-term dependencies and context in sequential data. BERT uses transformers to analyse and handle textual data more efficiently and effectively than typical recurrent neural networks (RNNs) or convolutional neural networks (CNNs). Furthermore, BERT incorporates attention methods that allow the model to focus on the most important elements of the input sequence. Attention processes give various weights to different sections of the input, emphasising crucial information while minimising the influence of irrelevant or less significant pieces.

This attention mechanism greatly contributes to the model's ability to extract meaningful features from the data. By combining these techniques, BERT enhances the performance of recurrent neural networks and convolutional neural networks, which were previously popular deep learning approaches for NLP tasks. BERT's ability to capture bidirectional context, leverage the power of transformers, and employ attention mechanisms results in improved accuracy and efficiency in tasks such as text classification, named entity recognition, and question-answering. In summary, in the future, cutting-edge deep learning algorithms like BERT will enable automatic data classification. BERT's unique combination of bidirectionality, multilayer transformers, and attention techniques allows for more accurate and efficient analysis of textual data compared to traditional approaches like RNNs and CNNs. This advancement promises to revolutionize various natural language processing tasks and improve overall performance in text-based applications. Although Bidirectional Long Short-Term Memory (BiLSTM) networks are effective and adaptable for some tasks, they have some drawbacks that should be taken into account for further study, including overfitting, computational intensity, imbalances in the quality of the data, hyperparameter tuning, lengthy training times, and sequential dependency.

Other References

- Ankush Chatterjeea, Umang Guptaa, Manoj Kumar Chinnakotlab, Radhakrishnan Srikanthb. (2019). Understanding Emotions in Text Using Deep Learning and Big Data. Computers in Human Behavior. 93(1):309-317.
- Botao Zhonga, Xuejiao Xinga, Peter Lovec, XuWangab. (2019). Convolutional neural network: Deep learning-based classification of building quality problems. Advanced Engineering Informatics. 40(1):46-57.
- Gochhait Saikat, Sharma K. Deepak, Singh Rathore Rajkumar and Jhaveri H. Rutvij*, Load Forecasting with Hybrid Deep Learning Model for Efficient Power System Management, Recent Advances in Computer Science and Communications 2024; 17 (1): e061023221828.

References

1. ^*Iqbal., Jalil, Gadekallu, Kryvinska. (2022). Authorship identification using ensemble learning. Scientific Reports. 12(1):1–16. doi: 10.1038/s41598-022-13690-4.*
2. ^*Meysam Chenaghlu, Shervin Minaee, Nal Kalchbrenner. (2021). Deep Learning--based Text Classification: A Comprehensive Review. ACM Computing Surveys. 54(3):1–40.*
3. ^*Haihua Chena, Jiangping Chena, WeiLuc, JunhuaDinga. (2022). A comparative study of automated legal text classification using random forests and deep learning. Information Processing & Management. 59(2):1–12.*
4. ^*Christophera Schröder, Andreas Nieklera. (2020). A Survey of Active Learning for Text Classification using Deep Neural Networks. Computation and Language. 4(2):1–16.*
5. ^*Ryo Akit, Akira Yoshihar, Takashi Matsubar, Kuniaki Uehar. (2016). Deep learning for stock prediction using numerical and textual information. International Conference on Computer and Information Science. 16251900:1–6. doi:10.1109/ICIS.2016.7550882.*
6. ^*Shakeel Ahmada, Muhammad Zubair, Fahad Mazaed, Sherafzal Khah. (2020). Classification of Poetry Text in to the Emotional States Using Deep Learning Technique. IEEE Access. 8(2):1–14.*
7. ^*Thirumalaisamy, Selvakumar, Kamaleshwar Thangavilou, Hariharan Rajadurai, Oumaima Saidani, Nazik Alturki, Sandeep kumar Mathivanan, Prabhu Jayagopal, and Saikat Gochhait. 2023. "Breast Cancer Classification Using Synthesized Deep Learning Model with*

Metaheuristic Optimization Algorithm" Diagnostics 13, no. 18: 2925.

8. *^Joseph, Prusa Khoshgoftar. (2017). Improving deep neural network design with new text data representations. Journal of Big Data. 4(7):1-16.*
9. *^Venkatesh, Ranjitha. (2018). Classification and Optimization Scheme for Text Data using Machine Learning Naïve Bayes Classifier. IEEE World Symposium on Communication Engineering. 18602207:1-10. doi.10.1109/WSCE.2018.8690536.*
10. *^MinYuan Cheng, Denny Kusoema, Richard AntoniGosna. (2020). Text mining-based construction site accident classification using hybrid supervised machine learning. Automation in Construction. 118(1):1-12.*
11. *^Saravanan S, Vinoth Kumar, Velliangiri Sarveshwaran, Alagiri Indirajithu, Elangovan, Shaikh Muhammad Allayear. (2022). Computational and Mathematical Methods in Medicine Glioma Brain Tumor Detection and Classification Using Convolutional Neural Network. Computational and Mathematical Methods in Medicine. 4380901:1-12.*
12. *^Pilar Lópeza, Manuel Carlos, Díaz-Galianoa, TeodoroMartín. (2021). Automatic medical protocol classification using machine learning approaches. Computer Methods and Programs in Biomedicine. 200(2):1.10.*
13. *^Saravanan S, Thirumurugan P. 2020. Performance Analysis of Glioma Brain Tumor Segmentation Using Ridgelet Transform and Co-Active Adaptive Neuro Fuzzy Expert System Methodology. Journal of Medical Imaging and Health Informatics. 10(11):2642-2648.*
14. *^Mojtaba Heidarysafa, Donald E. Brown, Kiana Jafari Meimanda. (2018). An Improvement of Data Classification Using Random Multimodel Deep Learning. International Journal of Machine Learning and Computing. 8(4):298-310.*
15. *^Hannah Kima, Young Seob. 2019. Sentiment Classification Using Convolutional Neural Networks. Applied Sciences. 9(11):1-14.*
16. *^Mai-Fenga, Tian Shaonanb, Lee Chihoon. (2019). Deep learning models for bankruptcy prediction using textual disclosures. European Journal of Operational Research. 274(2):743-758.*
17. *^Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Advances in neural information processing systems. 2017. p. 5998–6008.*
18. *^R Ravi Kumar, Babu Reddy, Praveen. 2019. Text Classification Performance Analysis on Machine Learning. International Journal of Advanced Science and Technology. 28(20):691-697.*
19. *^Weili-Fang, Hanbin Luo, ShuangjieXu, Peter Lovee, ZhenchuanLu. (2020). Automated text classification of near-misses from safety reports: An improved deep learning approach. Advanced Engineering Informatics. 44(2):1-12.*
20. *^Abdullah Y.Muaadab, HemanthaKumara, J.Hanumanthappa, J.V. BibalBenifacM. (2022). An effective approach for Arabic document classification using machine learning. Global Transitions Proceedings. 3(1):267-271.*
21. *^Menghan-Zhang. (2021). Applications of Deep Learning in News Text Classification. Scientific Programming for Smart Internet of Things. 6095354:1-9.*
22. *^Ammar Mohammeda, RaniaKoraa. (2022). An effective ensemble deep learning framework for text classification. Journal of King Saud University - Computer and Information Sciences. 34(10):8825-8837.*
23. *^Jamil Ahmada, Haleem Farmana, Zahoor Jan. (2022). Deep Learning Methods and Applications. Deep Learning: Convergence to Big Data Analytics. 12(2):31-42.*
24. *^Guanlan Hu, Mavra Ahmed, Mary R. L'Abbé. (2023). Natural language processing and machine learning approaches for food categorization and nutrition quality prediction compared with traditional methods. The American Journal of Clinical Nutrition. 117(3):553-563.*
25. *^Thuraya M. Omran a, Baraa T. Sharef b, Crina Grosan c, Yongmin Li. (2023). Transfer learning and sentiment analysis of Bahraini dialects sequential text data using multilingual deep learning approach. Data & Knowledge Engineering. vol.143, pp.1-14.*
26. *^Karim Abbasi, Parvin Razzaghi. (2020). Incorporating part-whole hierarchies into fully convolutional network for scene parsing', Expert Systems with Applications, 160:1-120.*
27. *^Parvin Razzaghi, Karim Abbasi, Pegah Bayat. (2020). Learning spatial hierarchies of high-level features in deep neural network', Journal of Visual Communication and Image Representation, 70:1-25.*
28. *^Parvin Razzaghi, Karim Abbasi, Mahmoud Shirazi, Niloofar Shabani. (2021). 'Modality adaptation in multimodal data', Expert Systems with Applications, 179:1-8.*
29. *^Yanshan-Wang, Feichen Shen, Atkinson, Shreyasee Amina, Hongfang Liu. (2019). A clinical text classification paradigm using weak supervision and deep representation. BMC Medical Informatics and Decision Making. 19(1):1-13.*
30. *^Xi X, Chen M, Wang Y, Yang H. Band-Optimized Bidirectional LSTM Deep Learning Model for Bathymetry Inversion. Remote Sensing. 2023; 15(14):3472. https://doi.org/10.3390/rs15143472*

Declarations

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.