A Comparative Study of Deep Learning's Performance Methods for News Article using Word Representations

[1]Iman Saladin B. Azhar, [2]Winda Kurnia Sari*, [3]Naretha Kawadha Pasemah Gumay
[1,2]Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya
[3]Information System, Faculty of Computer Science, Universitas Sriwijaya
[1,2,3]Jl. Palembang-Prabumulih, KM32 Inderalaya, Kabupaten Ogan Ilir, Sumatera Selatan
*e-mail: windakurniasari@unsri.ac.id

(***received***: 18 February 2025, ***revised***: 25 February 2025, ***accepted***: 26 February 2025)

Abstract

In natural language processing (NLP), text classification is a crucial task that involves analyzing textual data, which often has high dimensionality. A good word representation is essential to address this challenge, and the word representation using GloVe is one of the popular methods that provides pre-trained word representations in high-dimensional vectors. This research evaluates the effectiveness of three deep learning techniques Convolutional Neural Network (CNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM) for online news classification using 300-dimensional GloVe word representations. The CNN model utilizes convolutional and pooling layers to extract local features, the DNN relies on dense layers to learn abstract representations, while the LSTM excels at capturing long-term dependencies between words. The results show that the LSTM model achieved the best accuracy at 93.45%, followed by CNN at 91.24%, and DNN at 90.67%. The superiority of LSTM is attributed to its ability to effectively capture temporal relationships and context, while CNN offers efficiency with faster training times. Although DNN produced solid performance, it is less optimal in understanding word sequences. These findings indicate that LSTM outperforms the other models in online news text classification tasks.

Keywords: deep learning, CNN, DNN, LSTM, news classification

1 Introduction

Text classification serves as one of the core tasks in Natural Language Processing (NLP), with applications spanning sentiment analysis [1], [2], fake news detection [3], information retrieval [4], and recommendation systems [5]. As text data continues to grow in volume, efficiently managing and analyzing this data has become increasingly critical. Early studies on text classification often relied on traditional machine learning algorithms, such as Decision Trees [6], Naive Bayes [7], and Support Vector Machines (SVM) [8]. Although these methods achieved reasonable performance, they are generally constrained by feature representations that struggle to capture the deeper contextual and semantic relationships within textual data.

The advancement of technology has introduced deep learning (DL) as a promising approach for text classification, effectively addressing several existing challenges. Techniques like Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), and Long Short-Term Memory (LSTM) excel in automatically extracting critical features from text data through sophisticated network architectures [9],[10]. Unlike traditional approaches, DL methods are capable of identifying deeper contextual and semantic patterns, which are essential for handling more complex NLP tasks. Numerous comparative studies highlight that deep learning surpasses traditional methods in terms of accuracy and its capability to process large-scale textual data. For instance, [11] reports that LSTM outperforms SVM in multi-label text classification, especially in capturing word sequences and maintaining context over longer spans.

A key factor in deep learning's improved performance in text classification is the use of more sophisticated word representations. GloVe (Global Vectors for Word Representation), Word2Vec, and FastText are among the most popular word representation techniques. However, FastText has proven superior in several studies, primarily due to its ability to capture morphological information through subword representations, making it more adaptable to word variations and languages with complex structures [12]. Nevertheless, the choice of word representation technique is often tailored to the characteristics of the data and the research objectives, making GloVe a relevant and effective option

ISSN:2302-8149
e-ISSN:2540-9719

in certain contexts. GloVe employs a global frequency-based approach to words in a corpus, allowing the model to capture semantic relationships between words on a broader scale compared to Word2Vec and FastText, which are based on local context. This approach provides efficiency in word representation by accounting for word co-occurrence statistics across the entire corpus [13]. In addition, GloVe excels in handling data with a large and extensive vocabulary. By leveraging a global co-occurrence matrix, GloVe efficiently utilizes large-scale corpus data to produce high-quality word representations. This makes it an excellent choice for deep learning-based text classification applications, where global word relationships play a crucial role in improving model accuracy [14]

This research focuses on evaluating the performance of three deep learning models—Convolutional Neural Network (CNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM)—for news classification tasks utilizing GloVe word embeddings. While numerous studies have demonstrated the effectiveness of deep learning in text classification, this work aims to explore the specific advantages and limitations of each approach within the domain of news classification using enriched word representations. Through this comparison, the study seeks to enhance understanding of how deep learning can be applied to text classification and to identify the most suitable word representation method for such tasks.

2 Research Method

In general, the research methodology used in this study requires a clear framework to guide its stages. The research framework adopted consists of several steps, as illustrated in Figure 1, starting with a literature review. The literature review involves examining relevant studies in the field of text classification using deep learning methods conducted over the past five years. This aims to provide an up-to-date understanding of advancements in deep learning methods for text classification and the word representation techniques employed.

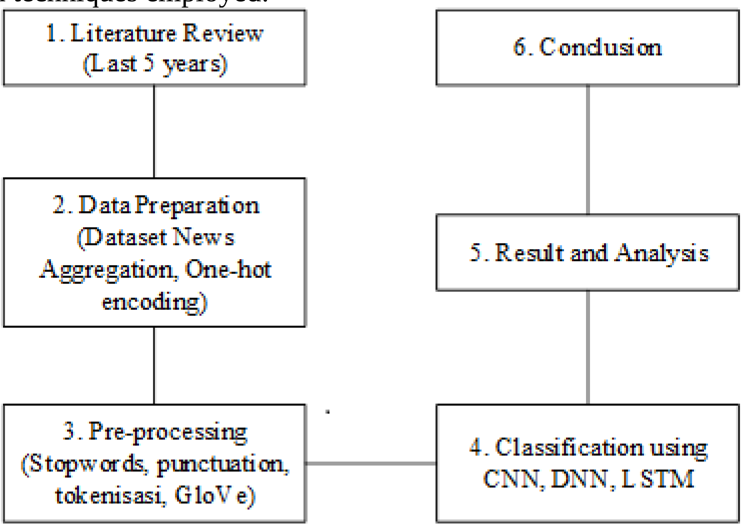


Figure 1. Research methodology

Data Preparation

Following the literature review, the next stage is data preparation. This study will utilize a news dataset obtained from the UCI Machine Learning Repository [15],[16], consisting of 422,937 text samples categorized into four labels, as shown in Table 1. This dataset will serve as the basis for comparing the performance of various deep learning methods in news classification. The data will undergo a pre-processing phase, which includes text cleaning, punctuation removal, stopword elimination, and tokenization to break the text into smaller units or tokens. Additionally, to handle categorical labels such as 'e', 'b', 't', and 'm', the one-hot encoding technique will be applied. This technique converts categorical labels into binary vector representations. Each label will be mapped into a separate column, where a value of 1 indicates the presence of the label and 0 indicates its absence. This conversion is essential for enabling deep learning algorithms to process labels in numerical format. Figure 2 illustrates the distribution of each label in the dataset, providing an overview of the balance or imbalance among the existing categories. This analysis helps ensure the model accounts for any potential biases during the classification process. The dataset exhibits a

ISSN:2302-8149
e-ISSN:2540-9719

noticeable imbalance, with category e having the highest number of samples (approximately 150,000), followed by categories b and t, which have relatively similar distributions (around 110,000 to 120,000), while category m contains the fewest samples (less than 50,000). Such an uneven distribution may lead to biased predictions, where the model becomes more inclined toward the majority class, potentially compromising its ability to accurately classify instances from the minority class.

Table 1. Dataset Description

Dataset Description	Value
Data training	380177
Data testing	42242
Classes	4 ('e', 'b', 't', and 'm')
Train/test ratio	90:10

Pre-Processing

This study employs stop words removal to eliminate words that do not significantly contribute to contextual understanding, such as conjunctions or other common words frequently appearing in text. Punctuation marks, such as periods, commas, and question marks, are also removed to eliminate irrelevant symbols that could interfere with text analysis. Next, tokenization is performed to break the text into smaller units or tokens, enabling the model to process information more effectively. Furthermore, the study utilizes GloVe 300-dimensional word embeddings to convert the words in the dataset into fixed-dimensional vector representations. GloVe captures semantic relationships between words in numerical vector form, allowing the model to comprehend the deeper context and meaning of each word in the text.

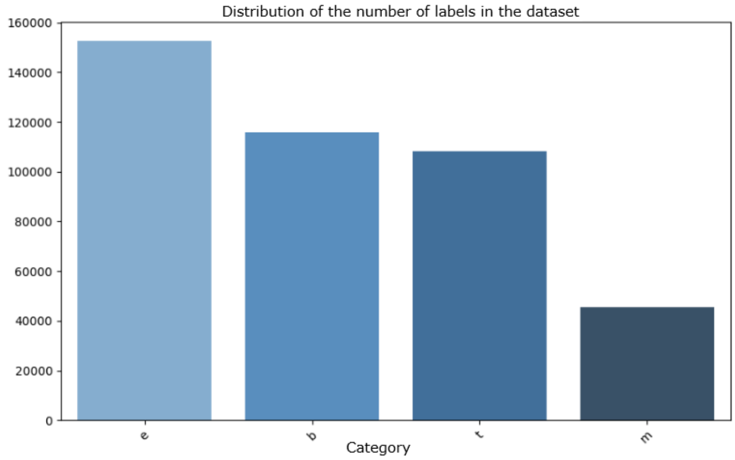


Figure 2. Distribution of the number of labels in the dataset

The embedding layer transforms text tokens into vector representations, providing a solid foundation for downstream deep learning tasks, as shown in Equation 1:

$$E_i = W_e . X_i \qquad (1)$$

Description:

E_i : Embedding representation from token to-i.
W_e: The embedding matrix (using pre-trained GloVe).
X_i : Token input (In the form of word index).

Convolutional Neural Network (CNN)

CNN is one of deep neural network specifically designed to identify key patterns in data through convolutional operations. Originally developed for image processing, CNNs have demonstrated their effectiveness across numerous natural language processing (NLP) tasks, such as text classification [17]. By leveraging multiple layers, CNNs are capable of detecting significant local features in text, such as distinctive word or phrase patterns.

ISSN:2302-8149
e-ISSN:2540-9719

The proposed process flow begins with converting input text into a numerical representation using a GloVe-based embedding layer. This representation is then processed through convolutional layers to capture important local features using kernels of progressively smaller sizes. Pooling is applied to reduce data dimensions while highlighting the most significant features, followed by a global max-pooling layer to generate a more concise final representation. This representation is passed through fully-connected layers to capture interactions between features, and finally, predictions are made using a SoftMax layer. This architecture is designed to capture both local and global information in text, enabling the model to effectively handle complex word patterns [18]. The proposed CNN architecture in this study is presented in Figure 3.

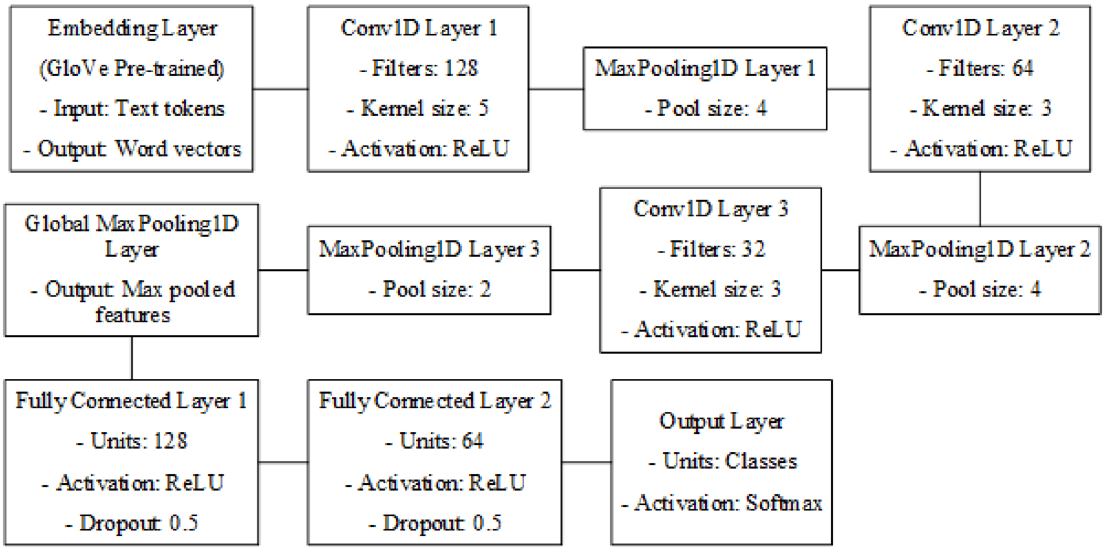


Figure 3. Proposed CNN architecture

Convolution operation forms the foundation of CNN. It is performed by applying a filter (kernel) to the input to extract features.

The Conv1D layer uses learnable filters to extract local features from input sequences, as shown in Equation 2.

$$Z[i] = \sigma\left(\sum_{j=0}^{k-1} X[i+j].W[j] + b\right) \qquad (2)$$

Description:
$Z[i]$: The output (feature map) at position i.
$X[i+j]$: Input text at position i+j.
$W[j]$: Kernel weights of a filter with size k.
b : Bias.
σ : Activation function.

The MaxPooling1D layer reduces dimensionality while preserving key features, as shown in Equation 3.

$$Z[i] = max\,\{X[i], X[i+1], \ldots, X[i+p-1]\} \qquad (3)$$

Description:
$Z[i]$: The output (feature map) at position i.
p : Pool size.
X : Feature map input.

The Global MaxPooling1D layer extracts the most prominent features from the entire sequence, as shown in Equation 4.

ISSN:2302-8149
e-ISSN:2540-9719

$$Z = max\{X[1], X[2], \dots, X[n]\} \tag{4}$$

Description:
Z : The global representation of the feature map.
$X[1], \dots, X[n]$: All values in the feature map.

Deep Neural Network (DNN)

DNN represents a widely adopted deep learning architecture applied in several Natural Language Processing (NLP) tasks. Within the NLP domain, DNN learns patterns and relationships in textual data by utilizing a series of fully-connected hidden layers. These layers progressively transform raw text inputs into more abstract and meaningful representations, allowing the model to handle tasks such as text classification, sentiment analysis, and other NLP-related applications [19]. Figure 4 illustrates the proposed DNN architecture in this study.

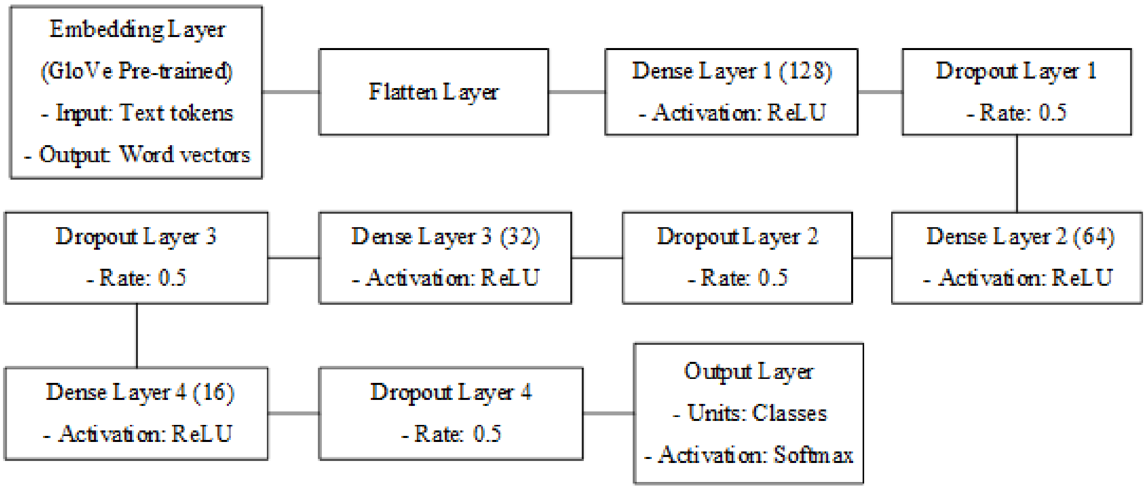


Figure 4. Proposed DNN architecture

Main Components in DNN:

A) Input Layer

This layer receives raw data as input, such as text features represented as vectors through embeddings such as GloVe.

B) Hidden Layers

Hidden layers are essential for processing data and detecting patterns within it. Within these layers, each neuron forms connections with every neuron in both the previous and subsequent layers. The mathematical computations performed include Equation 5:

$$Z^{(l)} = \sigma\left(W^{(l)}.Z^{(l-1)} + b^{(l)}\right) \tag{5}$$

Description:
$Z^{(l)}$: Output of layer l.
$W^{(l)}$: Weight matrix in layer l.
$Z^{(l-1)}$: Output of the previous layer.
$b^{(l)}$: Bias.
σ : Activation function

C) Activation Function

The activation function contains of non-linearity, allowing the network to capture and learn intricate relationships within the data. Common examples of activation functions include:

The ReLU activation function is simple and effective in mitigating the vanishing gradient problem, as shown in Equation 6.

$$f(x) = \max(0, x) \tag{6}$$

 ISSN:2302-8149
e-ISSN:2540-9719

The Softmax function converts logits into probability distributions for multi-class classification, as shown in Equation 7.

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \quad (7)$$

Long Short-Term Memory (LSTM)

LSTM is a distinct neural network architecture built upon Recurrent Neural Networks (RNNs) and designed to overcome the vanishing gradient problem often associated with standard RNNs [20]. This issue can hinder a network's ability to effectively learn long-term dependencies in sequential data, such as text, speech, or other signal-based inputs [21]. This study proposes an LSTM model architecture to improve performance in text classification, as illustrated in Figure 5.

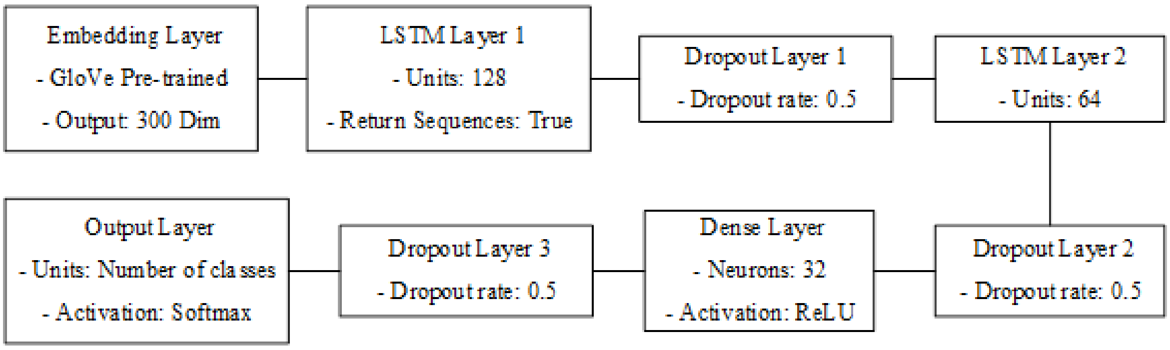


Figure 5. Proposed LSTM architecture

LSTM was first introduced by Hochreiter and Schmidhuber in 1997, incorporating a memory mechanism that allows the network to retain important information over long periods of time. This makes LSTM particularly well-suited for tasks such as text analysis, language translation, time series prediction, and more [22]–[24]. Each unit in an LSTM consists of three main gates that enable the model to regulate the flow of information:

A) Forget Gate

The forget gate determines which information from the previous cell state should be discarded, as defined in Equation 8:

$$f_t = \sigma\left(W_f . [h_{t-1}, x_t] + b_f\right) \quad (8)$$

Description:

h_{t-1} : The output from the previous time step
x_t : The input at time *t*.
σ : Sigmoid function.

B) Input Gate

The input gate decides which new information should be incorporated into the cell state. This process involves two steps:

- First, determine the important part of the new information as given in Equation 9:

$$i_t = \sigma(W_i . [h_{t-1}, x_t] + b_i) \quad (9)$$

- Second, create candidate new values for the cell state as given in Equation 10:

$$\tilde{C}_t = tanh(W_C . [h_{t-1}, x_t] + b_C) \quad (10)$$

C) Output Gate

The output gate determines the current output, as defined in Equation 11:

$$o_t = \sigma(W_o . [h_{t-1}, x_t] + b_o) \quad (11)$$

The output at time step *t* is calculated as defined in Equation 12:

$$[h_t = o_t . tanh(C_t)] \quad (12)$$

Three deep learning models CNN, DNN, and LSTM were implemented to analyze performance in news classification. CNN is known for its ability to capture spatial features through convolution operations, making it efficient for processing short-sequence data but less optimal in handling long-

ISSN:2302-8149
e-ISSN:2540-9719

term dependencies in text. DNN, consisting of multiple perceptron layers, can learn complex nonlinear patterns but lacks a specialized mechanism for capturing word order or temporal relationships in text. Meanwhile, LSTM is designed to handle long-term dependencies with a memory mechanism that retains information from previous sequences, making it superior in understanding context in sequential data such as news articles. Experimental results show that LSTM achieved the best performance in news classification due to its ability to capture semantic relationships between words in a sentence, albeit at a higher computational cost compared to CNN and DNN.

3 Results and Analysis

This study focused on evaluating the effectiveness of deep learning models, specifically Convolutional Neural Network (CNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM) in classifying online news articles using GloVe word representation. The models' performance was evaluated by measuring accuracy and their capacity to capture contextual information, particularly in understanding semantic relationships between words. Each model was fine-tuned with optimized parameters, and the news dataset underwent preprocessing steps to ensure high-quality data.

The evaluation results revealed that the LSTM model outperformed both CNN and DNN in terms of performance. The strength of LSTM lies in its capability to effectively capture temporal dependencies or word sequences in text, which is vital for text-based data. Thanks to its long-term memory feature, LSTM can retain important information from previous words to understand the overall context, leading to more precise predictions in online news text classification. As detailed in Table 2, the LSTM model achieved the highest accuracy of 93.45%, surpassing the CNN and DNN models, which recorded accuracies of 91.24% and 90.67%, respectively. This highlights the superior capability of LSTM in managing the complexities of textual data, outperforming other deep learning models.

Table 2. Model evaluation results

Model	Accuracy (%)
Proposed CNN	91.24
Proposed DNN	90.67
Proposed LSTM	93.45

Furthermore, Table 3 presents the classification report for the CNN model, Table 4 shows the classification report for the DNN model, and Table 5 displays the classification report for the LSTM model, highlighting the performance metrics such as precision, recall, and F1-score for each model.

Table 3. Classification report using CNN

Label	Precision	Recall	F1-Score	Support
b (Business)	88	89	89	11736
e (Entertainment)	95	96	95	15198
m (Health)	89	88	89	4439
t (Technology)	90	88	89	10869

Table 4. Classification report using DNN

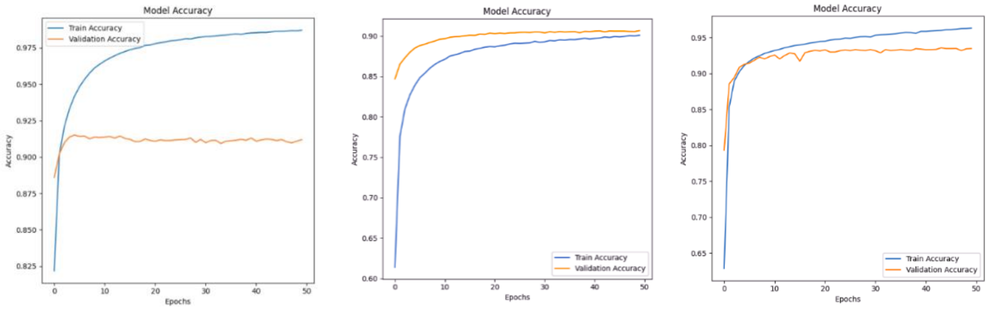
Label	Precision	Recall	F1-Score	Support
b (Business)	87	89	88	11736
e (Entertainment)	93	96	95	15198
m (Health)	90	86	88	4439
t (Technology)	91	86	89	10869

Table 5. Classification report using LSTM

Label	Precision	Recall	F1-Score	Support
b (Business)	90	93	92	11736
e (Entertainment)	96	97	97	15198
m (Health)	93	91	92	4439
t (Technology)	93	91	92	10869

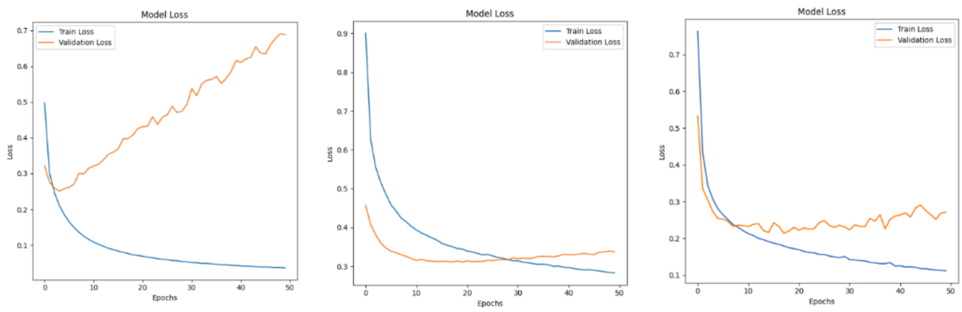
ISSN:2302-8149
e-ISSN:2540-9719

The comparison of accuracy and loss metrics for CNN, DNN, and LSTM models is illustrated in Figure 6 and Figure 7, respectively. Figure 6 presents the accuracy trends during the training and validation processes, showing the improvement and consistency of each model in classifying the text data. Meanwhile, Figure 7 highlights the loss values, which indicate the models' optimization performance over the training epochs. Although overfitting is still observed in all models, particularly during the later stages of training, the LSTM model demonstrates a superior ability to generalize compared to CNN and DNN. This is evident from its higher validation accuracy and lower validation loss, further confirming its effectiveness in handling the complexity of textual data for online news classification tasks.



(a) CNN (b) DNN (c) LSTM

Figure 6. Model accuracy of proposed method



(a) CNN (b) DNN (c) LSTM

Figure 7. Model loss of proposed method

4 Conclusion

This research evaluates the effectiveness of deep learning models, such as Convolutional Neural Network (CNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM) in classifying news articles with the use of GloVe word representation. The results show that the LSTM model consistently outperforms both CNN and DNN in terms of accuracy, achieving the highest score of 93.45%, while CNN and DNN attained 91.24% and 90.67%, respectively. The superior performance of LSTM is due to its ability to capture sequential dependencies and maintain contextual information throughout the text, allowing it to identify long-range word relationships—an important characteristic for tasks involving text data.

While all models show some level of overfitting, the LSTM model demonstrates superior generalization compared to CNN and DNN, as indicated by its lower validation loss and higher validation accuracy. These findings highlight the promise of LSTM as the most efficient deep learning approach for text classification when paired with GloVe embeddings. Future studies will focus on comparing other word representation techniques, such as FastText and Word2Vec, to assess their impact on LSTM performance. These investigations could provide deeper insights into how different feature representations affect the effectiveness of deep learning models in text classification tasks.

ISSN:2302-8149
e-ISSN:2540-9719

Acknowledgement

This dataset comes from the UCI Machine Learning Repository. Any publications that use this data should cite the repository as follows:

Lichman, M. (2013). UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science.

Reference

[1] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A *Survey on Sentiment Analysis Methods, Applications, and Challenges*," *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5731–5780, 2022.

[2] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "*A Survey of Sentiment Analysis un Social Media*," *Knowl. Inf. Syst.*, vol. 60, no. 2, pp. 617–663, 2019.

[3] W. K. Sari, I. S. B. Azhar, Z. Yamani, and Y. Florensia, "F*ake News Detection using Optimized Convolutional Neural Network and Bidirectional Long Short-Term Memory*." Computer Engineering and Applications, 2024.

[4] J. Guo *et al.*, "*A Deep Look Into Neural Ranking Models for Information Retrieval*," *Inf. Process. Manag.*, vol. 57, no. 6, p. 102067, 2020.

[5] A. Kumar Sharma, B. Bajpai, R. Adhvaryu, S. Dhruvi Pankajkumar, P. Parthkumar Gordhanbhai, and A. Kumar, "*An Efficient Approach of Product Recommendation System using NLP Technique,"* Mater. *Today Proc.*, vol. 80, pp. 3730–3743, 2023.

[6] B. T. Jijo and A. M. Abdulazeez, "*Classification based on Decision Tree Algorithm for Machine Learning*." Journal Of Applied Science And Technology Trends, 2021.

[7] S. Ruan, B. Chen, K. Song, and H. Li, "*Weighted Naïve Bayes Text Classification Algorithm based on Improved Distance Correlation Coefficient*," *Neural Comput. Appl.*, vol. 34, no. 4, pp. 2729–2738, 2022.

[8] D. E. Cahyani and I. Patasik, "*Performance Comparison of TF-IDF and Word2Vec Models for Emotion Text Classification*," *Bull. Electr. Eng. Informatics; Vol 10, No 5 Oct. 2021DO - 10.11591/eei.v10i5.3157* , Oct. 2021.

[9] Z. Ren, Q. Shen, X. Diao, and H. Xu, "*A Sentiment-Aware Deep Learning Approach for Personality Detection from Text*," *Inf. Process. Manag.*, vol. 58, no. 3, p. 102532, 2021.

[10] S. Hakak, M. Alazab, S. Khan, T. R. Gadekallu, P. K. R. Maddikunta, and W. Z. Khan, "*An Ensemble Machine Learning Approach Through Effective Feature Extraction to Classify Fake News*," *Futur. Gener. Comput. Syst.*, vol. 117, pp. 47–58, 2021.

[11] W. K. Sari, D. P. Rini, R. F. Malik, and I. S. B. Azhar, "*Multilabel Text Classification in News Articles using Long-Term Memory with Word2Vec*." Jurnal Resti, pp. 276–285, 2020.

[12] E. M. Dharma, F. L. Gaol, H. L. H. S. Warnars, and B. Soewito, "*The Accuracy Comparison among Word2vec, Glove, and Fasttext Towards Convolution Neural Network (CNN) Text Classification*." Journal of Theoretical and Applied Information Technology, 2022.

[13] R. A. Stein, P. A. Jaques, and J. F. Valiati, "*An Analysis of Hierarchical Text Classification using Word Embeddings*," *Inf. SCI. (Ny).*, vol. 471, pp. 216–232, 2019.

[14] J. Pennington and R. Socher, "*Glove: Global Vectors for Word Representation*," in *AES: Journal of the Audio Engineering Society*, 1971, vol. 19, no. 5, pp. 417–425.

[15] F. Gasparetti, "*News Aggregator*." UCI Machine Learning Repository, 2017.

[16] M. Lichman, "*UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]*." Irvine, CA: University of California, School of Information and Computer Science, 2013.

[17] A. K. Sharma, S. Chaurasia, and D. K. Srivastava, "*Sentimental Short Sentences Classification by using CNN Deep Learning Model with Fine Tuned Word2Vec*," *Procedia Comput. SCI.*, vol. 167, pp. 1139–1147, 2020.

[18] A. Kulkarni and A. Shivananda, "*Deep Learning for NLP BT - Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python*," A. Kulkarni and A. Shivananda, Eds. Berkeley, CA: Apress, 2021, pp. 213–262.

[19] B. Alshemali and J. Kalita, "*Improving the Reliability of Deep Neural Networks in NLP: A Review*," *Knowledge-Based Syst.*, vol. 191, p. 105210, 2020.

[20] S.-H. Noh, "*Analysis of Gradient Vanishing of RNNs and Performance Comparison*," *Information*, vol. 12, no. 11. 2021.

ISSN:2302-8149
e-ISSN:2540-9719

[21] M. Roodschild, J. Gotay Sardiñas, and A. Will, "*A New Approach for the Vanishing Gradient Problem on Sigmoid Activation*," *Prog. Artif. Intell.*, vol. 9, no. 4, pp. 351–360, 2020.
[22] A. Yadav, C. K. Jha, and A. Sharan, "*Optimizing LSTM for Time Series Prediction in Indian Stock Market*," *Procedia Comput. Sci.*, vol. 167, pp. 2091–2100, 2020.
[23] B. Ren, "*The use of Machine Translation Algorithm based on Residual and LSTM Neural Network in Translation Teaching*." Plos one, 15(11), e0240663, 2020.
[24] D. G. S. N. Murthy, S. R. Allu, B. Andhavarapu, M. Bagadi, and M. Belusonti, "*Text based Sentiment Analysis using LSTM*." International Journal of Engineering Research & Technology (IJERT), 2020.