

Enhancing text classification through novel deep learning sequential attention fusion architecture

Shilpa, Shridevi Soma

Department of Computer Science and Engineering, PDA College of Engineering, Visvesvaraya Technological University, Kalaburagi, India

Article Info

Article history:

Received Nov 24, 2023

Revised Mar 8, 2024

Accepted Mar 21, 2024

Keywords:

Deep learning

Multi-head attention mechanism

Natural language processing

Text classification

Text representation

ABSTRACT

Text classification is a pivotal task within natural language processing (NLP), aimed at assigning semantic labels to text sequences. Traditional methods of text representation often fall short in capturing intricacies in contextual information, relying heavily on manual feature extraction. To overcome these limitations, this research work presents the sequential attention fusion architecture (SAFA) to enhance the features extraction. SAFA combines deep long short-term memory (LSTM) and multi-head attention mechanism (MHAM). This model efficiently preserves data, even for longer phrases, while enhancing local attribute understanding. Additionally, we introduce a unique attention mechanism that optimizes data preservation, a crucial element in text classification. The paper also outlines a comprehensive framework, incorporating convolutional layers and pooling techniques, designed to improve feature representation and enhance classification accuracy. The model's effectiveness is demonstrated through 2-dimensional convolution processes and advanced pooling, significantly improving prediction accuracy. This research not only contributes to the development of more accurate text classification models but also underscores the growing importance of NLP techniques.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Shilpa

Department of Computer Science and Engineering, PDA College of Engineering

Visvesvaraya Technological University

Kalaburagi, India

Email: shilpa_122023@rediffmail.com

1. INTRODUCTION

According to Statistics in 2022 forecasts, it is projected that the global number of internet users will reach approximately 5.3 billion by 2023 [1]. The digital sphere facilitates the transfer of various forms of content, primarily textual material, through the processes of downloading and uploading. Each day as substantial volume of information is being generated, it is vital to engage in manual allotment of a significant quantity of textual content, however this project may incur significant costs, require a substantial time commitment, and present various challenges if undertaken [2]. The recommended approaches involve enabling text mining operations and automating information classification. Textual data is a valuable source of information, highlighting the necessity of employing efficient and cost-effective techniques for automated organization and analysis of texts in academic and commercial environments. The primary objective of text classification is to assign a given text to a predefined category. The discipline of natural language processing (NLP) encompasses a wide range of applications, thereby involving numerous responsibilities. Text categorization is commonly divided into three main categories: semi-supervised, unsupervised, and supervised [3].

Text classification is an essential subtask within the field of NLP. Its main goal is to assign one or more labels that represent the semantic meaning of a given sequence of text. Natural language inference (NLI) encompasses three types of classification tasks: binary classification, multi-classification, and multi-label classification. The aforementioned entity constitutes an integral part of the comprehensive framework. Furthermore, the system incorporates various modules to facilitate different functionalities. These modules include question answering (QA), topic classification (TC), news classification (CA), sentiment analysis (SA), and news classification [4].

The utilization of text representation is an essential intermediary step in the text classification procedure that is vital to employ. The significance of word context within sentences is often disregarded by conventional methods of text representation [5], furthermore, the aforementioned techniques are characterized by their labour-intensive nature and require a substantial allocation of resources. The main contributing factor to this issue is their reliance on human operators to perform manual feature extraction. The user's written content exhibits lack of clarity and information; text classification models are constructed using the principles of text representation learning. The textual representations that have been obtained can be effectively utilized for precise text classification, given that they possess the ability to differentiate based on classes [6].

Conventional approaches for representing text, such as the vector space model [7], require the utilization of deliberately crafted features. Conventional text representation approaches have garnered significant attention due to their user-friendly nature. The drawbacks of certain text representations include their high dimensionality and sparsity. In recent years, there has been notable advancement in the domain of NLP through the application of deep learning techniques. As a result, numerous models have been developed to extract properties from text. The models utilized in this study are the convolutional neural network (CNN) and the recurrent neural network (RNN) [8]. These models are derived from classical neural networks. These two methods address the problem of sparse, high-dimensional text representations by employing an end-to-end learning process for text feature representations. However, their capability to collect global word co-occurrences in corpora with non-continuous and long-range semantics is limited.

Attention processes have been widely employed in numerous studies to enhance the performance of text classification models. The objective of these approaches is to improve text representations by incorporating comprehensive text semantics. The majority of current neural-based text classification models fail to consider the interactions between phrases in a text when generating their text representations. In order to overcome the aforementioned limitation, text classification algorithms have resorted to utilizing graph neural networks (GNNs) as a potential remedy. Graph neural networks, commonly referred to as GNNs, have demonstrated significant advancements across various domains. In order to effectively gather the overall information of nodes, networks employ a message transmission mechanism on graphs. The primary difficulty associated with these methods pertains to the development of suitable textual graphs. The TextGCN model, as described in [9], is a GNN architecture designed specifically for text categorization tasks.

The operational procedure of this approach involves the conversion of textual information into a comprehensive heterogeneous graph. The graph in this illustration depicts words and documents as distinct nodes, providing a representation that is divided into two levels of detail. The subsequent phase of the procedure involves the utilization of a graph neural network for the purpose of classifying document nodes. In order to ensure consistent representation learning for documents within the same class and differentiated representation learning for documents across different classes, it is essential to obtain word representations and effectively distribute word information among the papers. However, it is important to note that a single word can possess multiple meanings, which can vary depending on the specific context in which it is employed. In the context of technology and food, the term "apple" can be interpreted in two distinct ways. The term "it" refers to both the fruit commonly known as apple and the renowned multinational technology company, Apple Inc. The aforementioned phrases will be assigned to distinct document categories, resulting in the dissemination of various types of information across these documents. The aforementioned factor will significantly influence the manner in which tasks such as text classification are executed in subsequent stages [10].

The escalating growth of digital content, particularly in textual form on the internet, underscores the need for robust and efficient methods of text classification to handle this vast and unstructured data. Automation of text classification has become an essential requirement due to the impracticality and high cost of manual content Allotment. The primary motivation behind this research stems from the demand for advanced techniques in the field of NLP to develop more precise and efficient mechanisms for text classification. This study is driven by the recognition of NLP's potential, coupled with existing methodologies such as graph neural networks and attention mechanisms, which significantly enhance the representation of textual data. It is envisioned that these advancements will not only prove valuable for academic research but also yield substantial advantages for businesses navigating the era of information abundance.

- Innovative multi-head attention mechanism (MHAM)-based approach: this research presents an innovative approach to text classification by combining MHAM models with deep long short-term memory (LSTM). The proposed model addresses the limitations of traditional MHAM attention mechanisms by effectively capturing local attributes and enhancing the understanding of textual data.
- Efficient attention mechanism: the study introduces a novel attention mechanism that overcomes the limitations of self-attention in MHAM models. By using a single-segment attention model in conjunction with deep, it efficiently captures dependencies between terms and optimizes data preservation, even for longer phrases.
- Advanced text classification framework: the research offers a comprehensive text classification framework that leverages advanced techniques, including attention mechanisms, convolutional layers, and pooling, to improve feature representation and classification accuracy. This methodology contributes to the development of more effective models for text classification tasks.
- Enhanced prediction accuracy: the proposed methodology incorporates 2-dimensional convolution processes and advanced pooling techniques to enhance the prediction accuracy of text classification. By efficiently extracting features and reducing the complexity of computations, the model significantly improves its ability to accurately classify text data, making it a valuable contribution to the field of NLP.

The research organization of this paper is in the first section a brief overview is given of the text recognition. In the second section a thorough literature survey is given of the existing techniques. In the third section the proposed methodology is given wherein the novel approach is designed, combining a MHAM attention model with bidirectional LSTM, and its architecture for text classification. In the last section the performance analysis is given which displays the results in form of tables and graphs.

2. RELATED WORK

For the purpose of text categorization two categories are often used in previous automated systems. Pattern matching can be accomplished using one of two different methods. Machine learning models and rule-based manual labelling are the two main approaches used for data labelling. Rule-based hand-craft labelling involves manually applying labels to data according to predefined rules [11]. Machine learning models, on the other hand, use pre-defined labels-like naïve Bayes (NB), K-nearest neighbours (KNN), and support vector machine (SVM)-to analyse patterns and correlations in the dataset and assign labels to data. Two major issues are responsible for these approaches inability to achieve high accuracy performance. Because the dataset formats were first limited to structured data, unprocessed text data presented challenges for algorithms like KNN and SVM [12]. This meant that a large quantity of data transformation was needed, which interfered with the administration of industrial data and was inconsistent with human cognitive processes. Furthermore, the amount of the dataset was bigger, necessitating the pre-training of technologies like transformer to increase accuracy in data mining and machine learning tasks. Therefore, the effectiveness and performance of existing NLP tasks were hampered by these methodologies. Compared to classical models that need the identification of functions or tasks for forecasting or classification, deep learning approaches have gained importance in the field of NLP for text categorization or generation tasks. Various datasets are supplied for the procedures [13]. Their ability to combine several tasks into a single model-which includes pre-trained language models like transformer, bidirectional encoder representations from transformers (BERT), and generative pre-trained transformers (GPT)-is the source of this capability [14].

The multi-class convolutional neural network (MCNN)-LSTM method for text classification in news data is presented in this research. To achieve the desired results, this method combines LSTM and CNN deep learning approaches. Word spatial organization in phrases, paragraphs, or pages can be captured using CNNs. When processing text-based input data, they are commonly used as feature extractors in conjunction with networks LSTMs [15]. This research introduces a new approach called the multi-hashing embedding-based differential neural architecture search technique for expressing multilingual text. The purpose of building a multi-hashing network is to effectively transfer syntactic and contextual semantic data between languages [16]. This network is also built to handle a variety of graph data formats with ease. Using parameterization-based gradient search, network topologies for multilingual text representation are explored. For each candidate operation in the search space, a continuous encoder-more precisely, a neural tensor network with multi-hashing embedding-is employed to estimate the likelihood. Maximizing search process efficiency is the aim of this methodology [17].

In this research, a unique self-supervised attention method is presented that does not require any annotation costs by using perturbations to support attention learning. The main objective is to improve accuracy by amplifying the noise level of specific words in the given sentence. Preserving the phrase's predictive power and meaning during this procedure is critical [18]. Using an attention-based gated graph neural network is the suggested approach for the automatic extraction of node feature representations.

Particularly designed for the context of connected P systems is the coupled p graph attention neural network (CPGANN) technique. To effectively reduce the dependency on non-sequential words over extended periods of time, the gating unit is being developed with attention to capture semantic links within the context. The attention technique is used by the CPGANN to extract keyword nodes. The nodes ability to distinguish for classification is improved by this extraction procedure, subgraph representations are then aggregated throughout the readout process using the retrieved keyword nodes [19].

Optimizing the effectiveness and accuracy of relative discrimination criterion (RDC) feature ranking is the main objective of the alternative relative discrimination criterion (ARDC). The primary objective of the ARDC is to seek and identify words that are commonly used in the positive class. The RDC and information relative discrimination criterion (IRDC) methodologies' outcomes were compared, and the study also looked at benchmarking functions that are often used, such as information gain (IG), Pearson correlation coefficient (PCC), and ReliefF. Boost extensible markup language (XML), an XML-based method for extreme multilabel text categorization, is presented in this work. BoostXML is a deep learning framework that has been significantly improved via the use of gradient boosting techniques. Through the use of tail labels and a primary focus on unfitted training instances, the BoostXML method enhances the residual. The suggested method involves giving tail labels more weights at every Boosting Step. It's best to include a corrective step throughout the optimization process to handle the possible issue of mismatching between the text encoder and weak learners. The model performs better when there is less chance of running across local optima [9].

3. PROPOSED METHODOLOGY

Considering the traditional methods of MHAM attention model being used, the mechanism of self-attention is not effective to understand and grasp the local attributes from the vectors of phrase that eventually omits essential data in phrases. Additionally, the comprehensive performance by the MHAM attention model is lacking because of the absence of language modelling capacity. Hence, a framework is proposed for the MHAM attention model and the architecture of this model is given in the Figure 1.

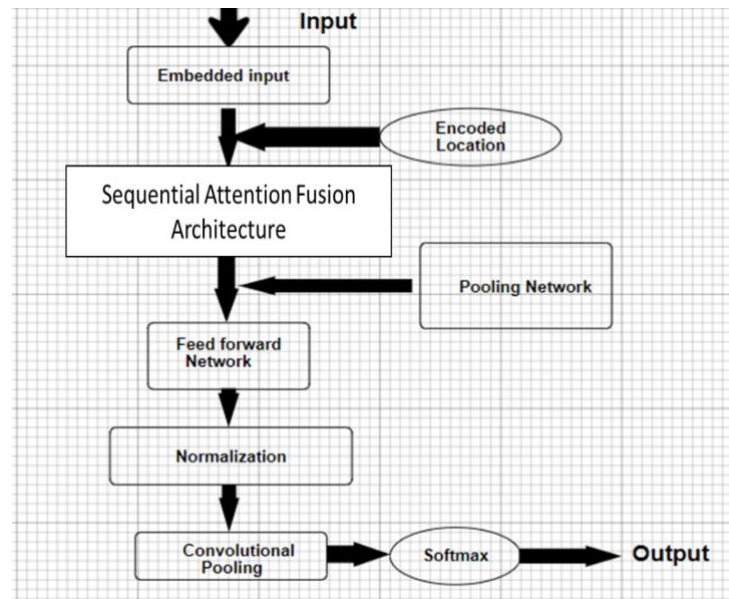


Figure 1. The architecture of the proposed MHAM attention model

This research work assumes $E = \{Y^j\}_j^o$ denotes the phrase set for every phrase that is termed as a particular class, the input data of E is transferred to the introduced method to derive attributes. Once the phrase vector is generated that is denoted by F_e to attain A which is the hidden state. Further, the attention method is used to grasp the increased distance data from A using these terms that have increased weights. However, a pool network is used for decreasing the size of A . At last, the final expression is produced by the contracted network that has a completely linked layer, which is utilized for prediction of the phrase tag. The probability of the phrase fits in the particular class l is represented as $q(l|E, \vartheta)$, in which the network attribute is represented as ϑ .

3.1. Multi-head attention mechanism model and long short-term memory

The model proposed in this study is distinct from the traditional methods used as MHAM model, as the proposed method uses a bidirectional LSTM to extract attributes that omits the dependency issue caused by RNN as well as decreases the complexity of computations that arises due to the multi-level attention model. Although, there is loss of data in bidirectional LSTM while the length of a phrase prolongs a particular length. Stating that bidirectional LSTM does not contribute to modification of data preservation. Therefore, a single segment attention model is combined with the bidirection LSTM to grasp various features. The relation between each of two terms is evaluated using vectors such as keys and queries to gain the internal links of the phrases. Hence, the attention model proposed efficiently omits any loss of data.

Consider a given phrase $Y^j = \{z_1, z_2, \dots, z_o\}$, where every term z_j is transformed as a vector a_j by selecting a matrix that is embedded X^e belongs to $S^{C \times M}$, in which the word size that is embedded is denoted as C and the dimension of vocabulary M . Consider $a_j = X^e * z_j$, in this case j is used to denote the j -th term in the phrase. After which a_j is transferred to the encoded position network to attain the position data in the input. Assume $F_e = f_j$ represents the output of a_j post the encoded position. The equations for encoded position are formulated as given in (1).

$$Q_{(position, 2j)} = \sin \sin (position/10000^{2j/e_{im}}) \quad (1)$$

$$Q_{(position, 2j)} = \cos \cos (position/10000^{2j/e_{im}}) \quad (2)$$

In the (1) and (2), the size of network is denoted as e_{im} and the vector size is given as j . This research work uses the bidirectional LSTM model that has a forward and backward method to produce the concealed representation. The forward LSTM receives $\{f_1, f_2, \dots, f_o\}$ to attain the forward concealed expression $\{i_{m0}, i_{m1}, \dots, i_{mo}\}$ by encoding, the back LSTM produces a backward concealed expression $\{i_{s0}, i_{s1}, \dots, i_{so}\}$ once the vectors $\{f_o, f_{o-1}, \dots, f_1\}$ are imported. Lastly, the forward concealed expression as well as the backward concealed expressions are split that results in $A: \{i_{m0} + i_{s0}, i_{m1} + i_{s1}, \dots, i_{mo} + i_{so}\}$. Considering the benefits and challenges of the attention model, an individual head attention model is proposed to attain essential data from the concealed expression A . This research work take into account the weight of the other terms during evaluation of the attention value of a single term through r and l , wherein r and l are the resulting vectors of the concealed expression A , the link between r_j as well as l_k is the attention value for different terms to the term, k belongs to $(1, o)$. While considering the attention method, the equations are given as mentioned in (3) and (4).

$$b_{j,k} = (r^j l^k)(e)^{-\frac{1}{2}} \quad (3)$$

$$b'_{j,k} = \exp(b_{j,k}) \left(\sum_{l=1}^o \exp(b_{j,l}) \right)^{-1} \quad (4)$$

Considering the (3) and (4), j, k belongs to $(1, o)$, e is the size of r and l to stop the dot product to be huge. The (4) is the formulation for transformation of attention value using the function softmax. The attention method output is evaluated using the equation given in (5). Here, the count of terms is denoted as o for the phrase, w is used to express the vector resulted from A . The sum of weights from various inputs is taken into consideration by fixing the value of $b'_{j,k}$.

$$c' = \sum_{k=1}^o b'_{j,k} w^k \quad (5)$$

3.2. Pool network

This research work observe that the structure of the attention model internally has transformations that are linear, this unavoidably undermines the representation of every term. This is issue is resolved by mapping the expression from small to large size and further to low size by utilizing the completely linked layer. Furthermore, the rectified linear unit (ReLU) function is used, this increases the strength of the phases that have increased values as well as restrains the phases that have lesser values in the term expression to modify the expression. Additionally, the spatial data as well as the evaluation are unavoidably growing while using attention model to grasp the dependency for long distance. Hence, the pooling mechanism is used to lessen the size of the concealed expression that decreases the count of parameters. Here, we select the max-pool to result in enhanced compression rather than average, which is given as (6). Here, \square_{CB} denotes the concealed expressions from the bidirectional LSTM as well as attention method, y^j as well as the *Normalization* shows the j -th phrase as well as the normalization.

$$Y_D = \text{MaxPooling}[\text{ReLu}(\text{Normalization}([y^j]_{CB}))] \quad (6)$$

3.3. Normalization

The bidirectional LSTM, attention model as well as the forward layer have an independent structure. Hence, we utilized a residue network to omit the gradient absence, which is given as follows in (7). In the given (7), $Y_{forward}$ shows the result of Y_D after the forward network. Further, we select the expression as it has to be standardized as shown in (8).

$$Y_E = Y_D + Y_{forward} \quad (7)$$

$$Y_{concealed} = \text{Normalization Layer} (Y_E) \quad (8)$$

3.4. Deep custom feature extractor

The traditional models would aid in the prediction of classes that are resulted by the encoder by linking to the linear layer. Considering the prior used MHAM, the grasping of features does not show to be efficient without prior training as it cannot attain effective results, mainly while the terms have an increased length. Hence, the network is deepened to grasp global data. Therefore, this research work uses 2D convolution process to enhance the expression of the embedded terms. Later, the length of sequence is reduced by 2 of its initial length using MaxPooling. The process uses various dimensions for the term to produce attribute maps as given in (9).

$$h^j = \partial(v \cdot z^{j:j+u-1} + c) \quad (9)$$

Here, h^j and ∂ denote the $z^{j:j+u-1}$ feature as well as the activation function respectively. v denotes the filter and the bias is shown as c . Furthermore, the Max-Pooling process for feature maps to capture the highest value feature h' showing the major feature for the filter, in which $h' = \text{maximum}\{h^1, h^2, \dots, h^{o-u+1}\}$. Therefore, the model is shown with two times increased data as the initial phrase. Additionally, it is shown that higher count of attribute maps leads to twice the count of output channels. Using this technique would widen the evaluation, although it would not enhance the accuracy during the process of classification. Therefore, this is resolved by the feature maps having an additional padding process, after which a MaxPooling process is performed to reduce the time of evaluation by half for every layer of convolution. A residue network is also used to stop the absence of the gradient. This is expressed as given in (10).

$$Y_F = \text{MaxPooling}[\text{padding}(Y_{concealed})] \quad (10)$$

$$I = \text{convolution}[\text{ReLU}(\text{padding}(Y_F))] + Y_F \quad (11)$$

In this case, the addition sign signifies the residue link I represents the final concealed expression that is utilized in the prediction of the possible class output using SoftMax. Considering the (12), e_j shows the $j - th$ phrase. The weight is denoted using X and the bias is given using c . The loss is calculated and the training attributes are used to reduce the loss function. The training error is given as mentioned in (13). The real tag is denoted as $z_j(k)$, the tag that is predicted is expressed as $z'_j(k)$. The count of phrases is given as o , the types of classes is given as n .

$$z'_j(k) = q(l|Y^j) \quad (12)$$

$$z'_j(k) = \text{SoftMax}(X^j I + c^j) \quad (12)$$

$$M(z_j, z'_j) = -(\sum_{k=1}^n \sum_{j=1}^o z_j(k) \log(z'_j(k))) \quad (13)$$

4. PERFORMANCE EVALUATION

In this study, the benchmark datasets, namely R8, MR, and R52 were utilized for text classification tasks. These datasets encompass diverse domains, such as news articles, movie reviews, and medical documents, making them suitable for evaluating a wide range of text classification methods. The methods employed in the study encompass a spectrum of approaches, from traditional techniques like term frequency-inverse document frequency (TF-IDF) and fastText to advanced deep learning models like CNN, LSTM, and BERT. The evaluation process involves measuring the performance of these methods on the datasets. For AGNews, the evaluation was based on Macro-F1 scores and accuracy the results are shown in

the form of graph and tables, closely followed by BERT-based models with techniques like ligation independent cloning (LIC) and Cathodoluminescence (CL). This extensive evaluation framework allowed for a robust assessment of text classification methods across various datasets, showcasing the nature of BERT-based models in text classification tasks.

4.1. Dataset details

The benchmark datasets-the R8 dataset, MR dataset, and R52 dataset [2] are used in the study to conduct the investigations. Two distinct subsets-the R8 and R52 datasets-extracted from the Reuters 21,578 datasets are required to be used in order to complete the multi-class tasks. There are 7,674 total documents in the collection known as R8. The documents are divided into two separate groups: Of the papers that were assigned, 2,189 were expressly designated for testing, while the rest 5,485 were assigned for training. Eight separate categories have been applied to the papers. Two sets of the dataset R52 have been created: a training set with 6,532 samples and a testing set with 2,568 samples. 52 categories are produced as a consequence of the split. A short text dataset used for binary sentiment classification is called the MR dataset. The purpose of the software's development was to analyse movie reviews. 10,662 reviews altogether from the corpus, which is made up of 5,331 good reviews and an equal number of bad reviews. Table 1 displays the summary of the datasets. Table 1 shows the summary of the datasets.

Table 1. Summary of the datasets

Dataset	Docs	Training	Test	Words	Nodes	Classes	Average length
R8	7,674	5,485	2,189	7,688	15,362	8	65.72
MR	10,662	7,108	3,554	18,764	29,426	2	20.39
R52	9,100	6,532	2,568	8,892	18,044	52	69.82

4.2. AGNEWS dataset

The AG's News dataset [20] which contains 127,600 samples with 4 classes. AG's News dataset, is a well-known and widely used text classification dataset. This dataset is specifically designed for text categorization tasks, making it valuable for training and evaluating NLP and machine learning models. The dataset comprises news articles collected from the AG's corpus, covering a wide range of topics, including world news, business, sports, and science. Each news article is associated with a corresponding category or class label, making it suitable for supervised learning tasks. Researchers and data scientists commonly use AG's News dataset to develop and test text classification algorithms, sentiment analysis models, and various NLP applications due to its real-world news content and diverse categories.

4.3. Results

Analysing the performance of various methods on the R8 dataset, it becomes evident that different approaches yield varying results. Among these methods, the paragraph vectors with a distributed bag of words model (PV-DBOW) model achieves a moderate accuracy of 85.87%, while paragraph vector-distributed memory (PV-DM) lags behind at 52.07%, indicating its lower efficiency. fastText shows a strong performance with an accuracy of 96.13%, while simple word-embedding-based models (SWEM) and low energy availability in males (LEAM) also exhibit average results of 95.32% and 93.31%, respectively. Notably, text- multimodal graph neural network (MGNN), existing system (ES) outperforms the others, achieving an accuracy of 97.39%. Traditional methods like TF-IDF + logistic regression (LR), CNN, and LSTM yield accuracies of 93.74%, 94.02%, and 93.68%, respectively. However, the post script (PS) model exhibits an accuracy of 98.13%, demonstrating its superior capabilities in text classification on the R8 dataset. Table 2 shows the comparison of various state-of-art techniques for different datasets. Figure 2 shows the comparison on R8 dataset.

Table 2. Comparison of various state-of-art techniques for different datasets

Method	R8	R52	MR
PV-DBOW [21]	0.8587	0.7829	0.6109
PV-DM [21]	0.5207	0.4492	0.5947
fastText [22]	0.9613	0.9281	0.7514
SWEM [23]	0.9532	0.9294	0.7665
LEAM [24]	0.9331	0.9184	0.7695
Text-MGNN ES [25]	0.9739	0.942	0.7746
TF-IDF + LR [26]	0.9374	0.8695	0.7459
CNN [27]	0.9402	0.8537	0.7498
LSTM	0.9368	0.8554	0.7506
PS	98.13	96.12	91.86

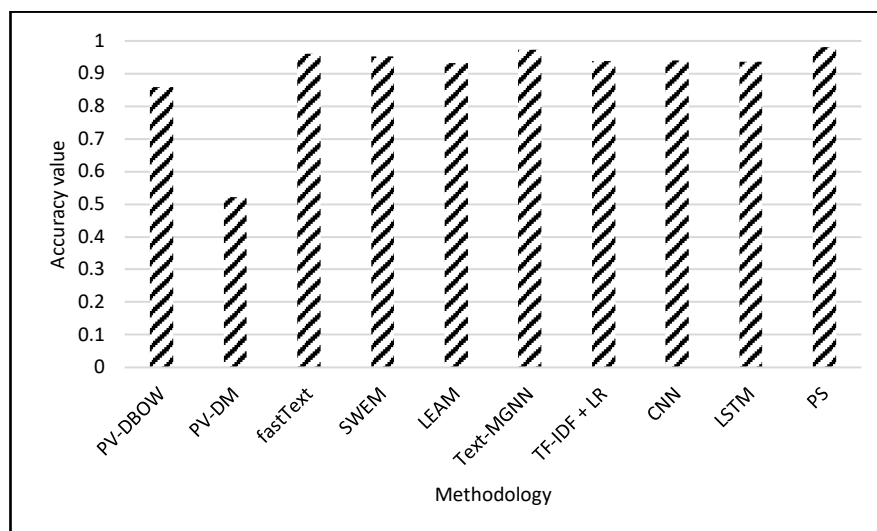


Figure 2. Comparison of different state-of-art techniques with PS on R8 Dataset

Analysing the performance of various methods on the R52 dataset. PV-DBOW exhibits an accuracy of 78.29%, showing moderate performance, whereas PV-DM trails behind at 44.92%, indicating limited effectiveness for this dataset. fastText and SWEM both perform well with accuracies of 92.81% and 92.94%, respectively. LEAM achieves an accuracy of 91.84%, demonstrating a strong classification capability. The Text-MGNN (ES) method performs remarkably well, with an accuracy of 94.20%. In contrast, traditional methods such as TF-IDF+LR, CNN, and LSTM yield accuracies of 86.95%, 85.37%, and 85.54%, respectively. Notably, the PS model, achieving an impressive accuracy of 96.12%. These results highlight the varying performance of different models, with PS emerging as the most accurate choice for text classification on the R52 dataset as shown in Figure 3.

Analysing the performance of different methods on the MR Dataset, this research work observes a range of accuracies. PV-DBOW and PV-DM both yields relatively low accuracy, with scores of 61.09% and 59.47%, respectively, indicating limited performance on this text classification task. fastText and SWEM exhibit moderate accuracy, with values of 75.14% and 76.65%, respectively. LEAM performs slightly better with an accuracy of 76.95%. The Text-MGNN (ES) method shows commendable performance, achieving an accuracy of 77.46%, making it one of the top performers in this dataset. Traditional methods like TF-IDF+LR, CNN, and LSTM achieve accuracies of 74.59%, 74.98%, and 75.06%, respectively. Notably, the PS model achieves an accuracy of 91.86%. These results illustrate the diversity in model performance, with PS emerging as the most accurate choice for text classification on the MR Dataset as shown in Figure 4.

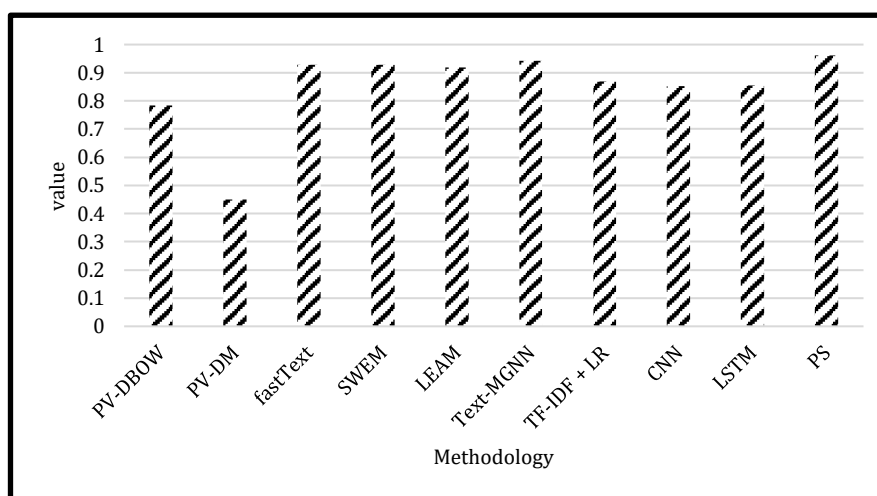


Figure 3. Comparison of different state-of-art techniques with PS on R52 Dataset

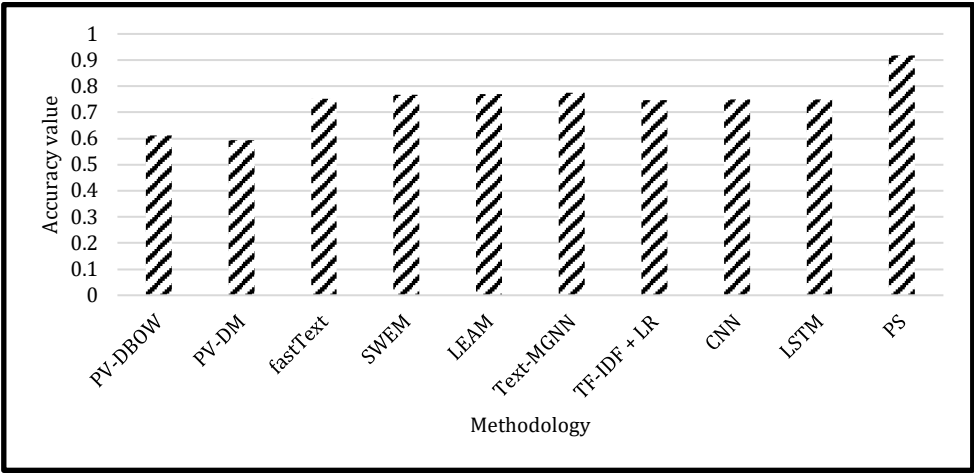


Figure 4. Comparison of different state-of-art techniques with PS on MR Dataset

Table 3 displays the results for accuracy on AGNews dataset in comparison with various models in a classification task, BERT+LIC+CL is the second most accurate model with an accuracy of 91.15, which is notably lower than PS. BERT+LIC+HNM+CL follows closely behind, with an accuracy of 91.08. The BERT model, achieves an accuracy of 89.02, showcasing the effectiveness of pre-trained language models. The CNN+LIC+CL model also performs well with an average accuracy of 88.71. Generally, models that incorporate BERT, particularly with the addition of LIC and CL, deliver strong accuracy, PS stands out as the best-performing model with an accuracy of 95.97. Figure 5 displays the accuracy comparison AGNews dataset.

Table 3. Comparison on AGNews Dataset		
Model	Accuracy	Macro-F1
CNN [26]	0.8797	0.8797
CNN+LIC	0.8822	0.882
CNN+CL	0.8763	0.8757
CNN+LIC+CL	0.8871	0.8872
CNN+LIC+HNM+CL	0.8844	0.8848
BERT [28]	0.8902	0.8895
BERT+LIC	0.8899	0.9022
BERT+CL	0.8893	0.8867
BERT+LIC+CL	0.9115	0.9123
BERT+LIC+HNM+CL	0.9108	0.9127
PS	0.9597	0.9678

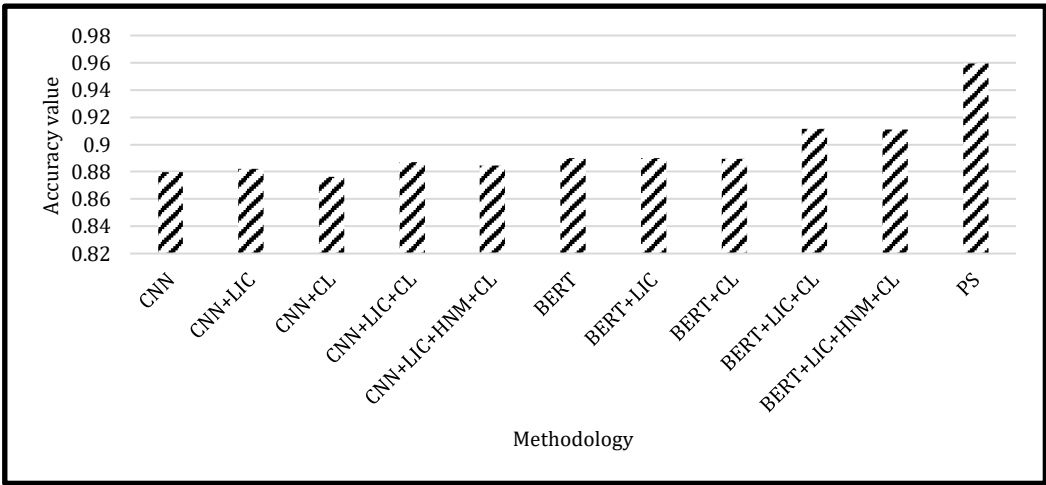


Figure 5. Comparison of accuracy for different state-of-art techniques with PS on AGNews Dataset

The Macro-F1 score graph is plotted in Figure 6. Following closely, BERT+LIC+HNM+CL and BERT+LIC+CL both deliver high Macro-F1 scores of 91.27 and 91.23, respectively, making them solid contenders. The BERT model, when combined with various techniques like LIC and CL, consistently performs well, scoring 88.95, 90.22, and 88.67, respectively. Meanwhile, the CNN models achieve average scores, with CNN+LIC+CL and CNN+LIC+HNM+CL leading the CNN-based models with scores of 88.72 and 88.48, respectively. CNN on its own achieves a Macro-F1 score of 87.97, which is lower than the top-performing models. PS achieves a remarkable Macro-F1 score of 96.78, signifying its robust capability for text classification tasks.

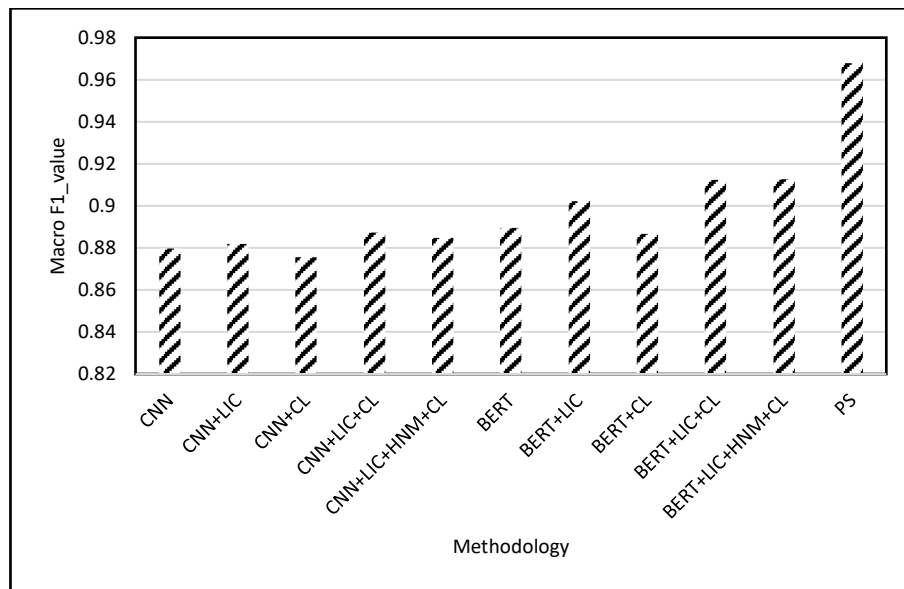


Figure 6. Comparison of Macro-F1 for different state-of-art techniques with PS on AGNews Dataset

4.4. Comparative analysis

In the evaluation of models across four datasets-AGNews, R8, R52, and MR. Table 4 presents a comparative analysis of two models, ES and PS, across three distinct datasets: R8, R52, and MR. The metrics used for evaluation indicate the performance of these models in text classification tasks. Notably, PS outperforms ES in all three datasets, showcasing higher accuracy in R8, R52, and MR by approximately 0.76%, 2.02%, and a significant 17.01%, respectively. This demonstrates the considerable effectiveness of the PS model, particularly in the MR dataset, where its performance improvement is substantial. The results suggest that PS significantly enhances the accuracy of text classification compared to ES across these diverse datasets, making it a favorable choice for text classification tasks. The improvements, especially in the MR dataset, highlight the potential of advanced text classification models to achieve higher accuracy in handling unstructured textual data.

Table 4. Comparative analysis

Dataset	ES	PS	Improvisation (%)
R8	0.9739	0.9813	0.756956
R52	0.942	0.9612	2.01765
MR	0.7746	0.9186	17.0092

Table 5 provides a comparative analysis of two models, ES and PS, on the AGNews dataset in terms of both accuracy and Macro-F1 scores. The results clearly demonstrate that the PS model outperforms ES in both metrics. When it comes to accuracy, PS achieves an impressive 95.97%, while ES lags behind with 91.08%, indicating a substantial 5.23% improvement. Similarly, in terms of Macro-F1 scores, PS leads with a strong 96.78%, significantly surpassing ES, which scores 91.27%. This remarkable improvement of 5.86% highlights the efficiency of the PS model in text classification tasks within the AGNews dataset.

Table 5. Comparative analysis for both models

Dataset	ES	PS	Improvisation (%)
AGNews (Accuracy)	0.9108	0.9597	5.22855
AGNews (Macro-F1)	0.9127	0.9678	5.86014

5. CONCLUSION

In conclusion, this research addresses the critical need for advanced text classification methods in the ever-expanding digital landscape, driven by the exponential growth of unstructured textual data on the internet. The motivation behind this study stems from the demand for precise and efficient NLP techniques to automate content curation, given the impracticality and high cost of manual methods. This work introduces an innovative MHAM-based approach that effectively combines MHAM models with bidirectional LSTM to enhance text representation and classification. The novel attention mechanism presented here overcomes the limitations of traditional self-attention in MHAM models, enabling better data preservation, even for longer text passages. The comprehensive text classification framework introduced leverages advanced techniques, including attention mechanisms, convolutional layers, and pooling, to improve feature representation and classification accuracy. This research makes a valuable contribution to the field of NLP by enhancing prediction accuracy, particularly in multi-class text classification tasks. The proposed methodology optimizes the effectiveness and accuracy of text classification, paving the way for more efficient handling of vast amounts of textual data on the internet.




REFERENCES

- [1] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, "Document-level text classification using single-layer multisize filters convolutional neural network," *IEEE Access*, vol. 8, pp. 42689–42707, 2020, doi: 10.1109/ACCESS.2020.2976744.
- [2] Y. Gu, Y. Wang, H. R. Zhang, J. Wu, and X. Gu, "Enhancing text classification by graph neural networks with multi-granular topic-aware graph," *IEEE Access*, vol. 11, pp. 20169–20183, 2023, doi: 10.1109/ACCESS.2023.3250109.
- [3] Z. Xie, W. Lv, S. Huang, Z. Lu, B. Du, and R. Huang, "Sequential graph neural network for urban road traffic speed prediction," *IEEE Access*, vol. 8, pp. 63349–63358, 2020, doi: 10.1109/ACCESS.2019.2915364.
- [4] Z. Ye, Y. J. Kumar, G. O. Sing, F. Song, and J. Wang, "A comprehensive survey of graph neural networks for knowledge graphs," *IEEE Access*, vol. 10, pp. 75729–75741, 2022, doi: 10.1109/ACCESS.2022.3191784.
- [5] Z. Xing and S. Tu, "A graph neural network assisted monte carlo tree search approach to traveling salesman problem," *IEEE Access*, vol. 8, pp. 108418–108428, 2020, doi: 10.1109/ACCESS.2020.3000236.
- [6] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 2019, pp. 7370–7377.
- [7] M. Kutbi, "Named entity recognition utilized to enhance text classification while preserving privacy," *IEEE Access*, vol. 11, pp. 117576–117581, 2023, doi: 10.1109/ACCESS.2023.3325895.
- [8] K. Zeng *et al.*, "ITSMATCH: improved safe semi-supervised text classification under class distribution mismatch," *IEEE Transactions on Consumer Electronics*, pp. 1–1, 2024, doi: 10.1109/TCE.2023.3323982.
- [9] F. Li, Y. Zuo, H. Lin, and J. Wu, "BoostXML: Gradient boosting for extreme multilabel text classification with tail labels," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2024, doi: 10.1109/TNNLS.2023.3285294.
- [10] X. Zhao, Y. An, N. Xu, and X. Geng, "Variational continuous label distribution learning for multi-label text classification," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–15, 2024, doi: 10.1109/TKDE.2023.3323401.
- [11] X. Chen, P. Cong, and S. Lv, "A long-text classification method of Chinese news based on BERT and CNN," *IEEE Access*, vol. 10, pp. 34046–34057, 2022, doi: 10.1109/ACCESS.2022.3162614.
- [12] I. Fursov *et al.*, "A differentiable language model adversarial attack on text classifiers," *IEEE Access*, vol. 10, pp. 17966–17976, 2022, doi: 10.1109/ACCESS.2022.3148413.
- [13] G. Althari and M. Alsulmi, "Exploring transformer-based learning for negation detection in biomedical texts," *IEEE Access*, vol. 10, pp. 83813–83825, 2022, doi: 10.1109/ACCESS.2022.3197772.
- [14] Q. Qi, L. Lin, R. Zhang, and C. Xue, "MEDT: Using multimodal encoding-decoding network as in transformer for multimodal sentiment analysis," *IEEE Access*, vol. 10, pp. 28750–28759, 2022, doi: 10.1109/ACCESS.2022.3157712.
- [15] K. M. Hasib *et al.*, "MCNN-LSTM: Combining CNN and LSTM to classify multi-class text in imbalanced news data," *IEEE Access*, vol. 11, pp. 93048–93063, 2023, doi: 10.1109/ACCESS.2023.3309697.
- [16] X. Yan, H. Huang, Y. Jin, L. Chen, Z. Liang, and Z. Hao, "Neural architecture search via multi-hashing embedding and graph tensor networks for multilingual text classification," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 8, no. 1, pp. 350–363, Feb. 2024, doi: 10.1109/TETCI.2023.3301774.
- [17] H. Feng, Z. Lin, and Q. Ma, "Perturbation-based self-supervised attention for attention bias in text classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 3139–3151, 2023, doi: 10.1109/TASLP.2023.3302230.
- [18] J. Zhang and X. Liu, "A gated graph neural network with attention for text classification based on coupled P systems," *IEEE Access*, vol. 11, pp. 72448–72461, 2023, doi: 10.1109/ACCESS.2023.3295572.
- [19] S. A. Alshalif *et al.*, "Alternative relative discrimination criterion feature ranking technique for text classification," *IEEE Access*, vol. 11, pp. 71739–71755, 2023, doi: 10.1109/ACCESS.2023.3294563.
- [20] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, pp. 1–9, 2015.
- [21] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," *31st International Conference on Machine Learning, ICML 2014*, vol. 4, pp. 2931–2939, 2014.
- [22] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in *Proceedings of the 15th*




- Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2017, pp. 427–431, doi: 10.18653/v1/E17-2068.
- [23] D. Shen *et al.*, “Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2018, pp. 440–450, doi: 10.18653/v1/P18-1041.
- [24] G. Wang *et al.*, “Joint embedding of words and labels for text classification,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2018, pp. 2321–2331, doi: 10.18653/v1/P18-1216.
- [25] Z. Rezaei, B. Eslami, M.-A. Amini, and M. Eslami, “Hierarchical three-module method of text classification in web big data,” in *2020 6th International Conference on Web Research (ICWR)*, IEEE, Apr. 2020, pp. 58–65, doi: 10.1109/ICWR49608.2020.9122326.
- [26] Y. Kim, “Convolutional neural networks for sentence classification,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2014, pp. 1746–1751, doi: 10.3115/v1/D14-1181.
- [27] P. Liu, X. Qiu, and H. Xuanjing, “Recurrent neural network for text classification with multi-task learning,” *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, vol. 2016, pp. 2873–2879, 2016.
- [28] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 1, pp. 4171–4186, 2019.

BIOGRAPHIES OF AUTHOR



Mrs. Shilpa    received her Bachelors Degree in Computer Science and Engineering from the Visvesvaraya Technological University, BELGAUM - India in 2010 and Master Degree in Computer Science and Engineering from same University in 2012. She is currently pursuing her Ph.D. degree from the same university. She is presently working as Assistant Professor in Department of Computer Science and Engineering, Sharnbasva University Kalaburagi, Karnataka, India. Her primary area of interest is image processing, machine learning, and pattern recognition. She can be contacted at this email: shilpa_122023@rediffmail.com.



Dr. Shridevi Soma    working presently as Professor & HOD in Department of Computer Science and Engineering, Poojya Doddappa Appa College of Engineering, Kalaburagi. She has 18 years of teaching and 10 years of research experience, and completed her B.E., M.Tech., and Ph.D. in Computer Science and Engineering. Her research area includes digital image processing, pattern recognition, cloud computing, internet of things, and big data analytics. She published more than 30 research papers in above mentioned areas, also guiding research students. She has also received grant for establishment of “Cloud Computing Lab” from VGST. She can be contacted at email: shridevisoma@gmail.com