

# LSTM-CNN Hybrid Model Performance Improvement with BioWordVec for Biomedical Report Big Data Classification

Dian Kurniasari<sup>1,2</sup>, Warsono<sup>2\*</sup>, Mustofa Usman<sup>2</sup>, Favorisen Rosyking Lumbaraja<sup>3</sup>, Wamiliana<sup>2</sup>

<sup>1</sup>Doctoral Student at the Faculty of Mathematics and Natural Sciences, Universitas Lampung, Bandar Lampung, 35145, Indonesia

<sup>2</sup>Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Bandar Lampung, 35145, Indonesia

<sup>3</sup>Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Bandar Lampung, 35145, Indonesia

\*Corresponding author: warsono.1963@fmipa.unila.ac.id

## Abstract

The rise in mortality rates due to leukemia has fueled the swift expansion of publications concerning the disease. The increase in publications has dramatically affected the enhancement of biomedical literature, further complicating the manual extraction of pertinent material on leukemia. Text classification is an approach used to retrieve pertinent and top-notch information from the biomedical literature. This research suggests employing an LSTM-CNN hybrid model to tackle imbalanced data classification in a dataset of PubMed abstracts centred on leukemia. Random Undersampling and Random Oversampling techniques are merged to tackle the data imbalance problem. The classification model's performance is improved by utilizing a pre-trained word embedding created explicitly for the biomedical domain, BioWordVec. Model evaluation indicates that hybrid resampling techniques with domain-specific pre-trained word embeddings can enhance model performance in classification tasks, achieving accuracy, precision, recall, and f1-score of 99.55%, 99%, 100%, and 99%, respectively. The results suggest that this research could be an alternative technique to help obtain information about leukemia.

## Keywords

Leukemia, BioWordVec, Hybrid LSTM-CNN, Hybrid Resampling

Received: 26 November 2023, Accepted: 19 February 2024

<https://doi.org/10.26554/sti.2024.9.2.273-283>

## 1. INTRODUCTION

Leukemia, often known as blood cancer, results from immune system impairment caused by elevated white blood cells in the blood or bone marrow (Ahmed et al., 2019; Du et al., 2022; Saeed et al., 2022). Leukemia is a primary topic of biomedical research because of the rising mortality rate associated with the disease. Information on leukemia can be found in scientific resources, including PubMed, ScienceDirect, Scopus, and Google Scholar. Tober (2011) and Masic and Ferhatovica (2012) highlighted that PubMed is the largest and most popular database for the biomedical field among the four databases. It is known for providing more thorough search results for biomedical documents than other databases.

The initial documentation of the leukemia study in PubMed was written by Weeks (1864). However, since February 2024, searching for "leukemia" on PubMed would yield 374,286 papers, indicating a significant increase in leukemia-related research. The rapid publication rate has dramatically expanded the volume of biomedical text data, posing challenges in manually extracting leukaemia-related information. Text classifi-

cation is a frequently utilized approach for getting pertinent, top-notch information from biomedical publications.

Text classification, or text categorization, is a core issue in Natural Language Processing (NLP) that entails labelling textual elements such as sentences, questions, or paragraphs based on pre-labeled documents (Korde and Mahender, 2012; Zhao et al., 2022). Text classification is frequently utilized in biological contexts to distinguish cancer cases in pathology reports specific phenotypes in clinical narratives, and provide medical subject titles (MeSH keywords) to scientific papers (Rios and Kavuluru, 2015). Li et al. (2022) discovered that between 1960 and 2010, text classification studies predominantly concentrated on conventional Machine Learning (ML) methods, including Naïve Bayes (NB), K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). Since the 2010s, text classification methods have transitioned from traditional machine learning models to deep learning models. Deep learning models can create meaningful representations by utilizing word embedding techniques internally, leading to the observed shift.

Word embedding represents each word in a text in a pre-

defined vector space (Li and Yang, 2018). Word embedding is categorized into custom-trained embedding and pre-trained word embedding. The primary distinction between the two methods is the benefits of pre-trained word embedding. This method is particularly efficient at gathering semantic and contextual connections between words due to its training on a vast corpus of data. Pre-trained word embeddings can improve the accuracy of learning models when there is consistency between the data domain and the corpus used for training (ALRashdi and O'Keefe, 2019; Asudani et al., 2023; Rabut et al., 2019). Conversely, custom-trained embedding is solely trained using specified datasets (Sabbah and Fasihuddin, 2023).

Regrettably, existing pre-trained word embeddings like Word2Vec, GloVe, and FastText are typically created from the news corpus or Wikipedia (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2017). Hence, its use is not well-suited for the biomedical field. Zhang et al. (2019) created a pre-trained word embedding called BioWordVec to address the issue. BioWordVec is constructed utilizing specific corpora from the biomedical fields, specifically PubMed and MeSH, to ensure its relevance to the biomedical domain.

Recently, using pre-trained word embedding in different Deep Learning models like Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) has been very successful in classification tasks. The RNN model is a fundamental deep-learning architecture used by academics for text classification problems (Kowsari et al., 2019; Lai et al., 2015; Lan et al., 2020). Despite the vanishing gradient issues in RNNs, advancements like Long Short-Term Memory (LSTM) (Tai et al., 2015; Zhou et al., 2016) and Gated Recurrent Unit (GRU) (Li et al., 2019; Rathnayaka et al., 2018) have been introduced to overcome these limitations and have shown to be more efficient than traditional RNNs in various applications. The CNN model is a deep learning architecture known for its strong generalization abilities in extracting abstract features from data, particularly spatial data. Its uses extend beyond text categorization and are prevalent in diverse disciplines such as computer vision, audio processing, and face recognition.

While the accomplishments of the RNN and CNN models are well-known, research on hybrid models in deep learning is now emerging. Developing a hybrid model aims to enhance text analysis accuracy by incorporating several viewpoints and enhancing the efficiency of an individual deep learning model. One example of a diversified integration of hybrid models is the CNN-RNN hybrid model proposed by Akpatsa et al. (2021). An excellent RNN architecture for processing sequential data like text is LSTM. Thus, the prevalent hybrid model for text categorization, which mixes CNN and LSTM models, is extensively used.

Hung (2019) compared the CNN-LSTM hybrid model and the LSTM-CNN hybrid model for emotion analysis using seven distinct datasets from the general domain. They utilized Skip-gram for word representation. Their findings indicated that the CNN-LSTM hybrid model outperformed the LSTM-CNN hybrid model regarding efficiency and time consump-

tion. Wang et al. (2020) developed a model called MLCNN (Merge-LSTM-CNN) that integrates LSTM and CNN for text categorization. Text is transformed into low-dimensional vectors using a custom-word embedding technique. Combining LSTM with CNN models has been proven to enhance the accuracy of text classification, as indicated by their findings.

Tasdelen and Sen (2021) suggest a hybrid pre-miRNA classification model that combines CNN and LSTM. During data preparation, one hot encoding was utilized to transform each miRNA text base into a matrix of equal size. Their research indicates that the CNN-LSTM hybrid model outperforms other models, such as CNN, SVM, and Random Forest, in pre-miRNA classification, with an accuracy of 94.3%. Jain and Kashyap (2023) categorize sentiment about the COVID-19 virus by utilizing the CNN-LSTM hybrid model. The data used are tweets related to COVID-19. The proposed model achieved an accuracy of 92.34%.

Hasib et al. (2023) tackle multiclass text classification challenges, including imbalanced data, by utilizing a hybrid CNN and LSTM model in conjunction with the Tomek-Links algorithm, an under-sampling strategy. Each word is transformed into a compact vector representation with low dimensions during training using a pre-trained Word2Vec word embedding through an embedding layer. The study found that the CNN-LSTM hybrid model performed better when applied to balanced data, achieving a f1-score of 98.00%, compared to 95.00% when applied to imbalanced data.

Using hybrid models and word embeddings in text classification enhances model performance. Nevertheless, its current utility in the field of biomedicine is restricted. This research presents a novel hybrid model, LSTM-CNN, which incorporates BioWordVec, a pre-trained word embedding explicitly designed for the biomedical domain, to enhance the classification performance of biomedical big data reports on Genia Biomedical Event data. The dataset comprises PubMed abstracts that consist of multiclass cases or classes exceeding two.

In the context of multiclass, the distribution of classes is typically imbalanced. In order to prevent bias and inaccuracies in the classification, which can lead to errors in diagnosis, balanced class distribution in the classification of biomedical texts is an imperative biomedical necessity (Fotouhi et al., 2019; Abdulrauf Sharifai and Zainol, 2020). Therefore, we propose a new hybrid resampling technique that combines undersampling and oversampling concepts to tackle the issue. The model's performance will be assessed using accuracy, precision, recall, and f1-score metrics.

## 2. EXPERIMENTAL SECTION

### 2.1 Materials

The study utilized data from the GENIA Biomedical Event training dataset on the <https://www.kaggle.com> website. The dataset comprises 8,000 biomedical sentences relevant to leukemia and their associated TriggerWord. TriggerWord is a term used to describe a keyword in a biomedical statement. The

algorithm was trained to categorize biomedical texts using specific descriptive terms, making locating material about leukemia easier. Figure 1 displays the TriggerWord variable's frequency through a WordCloud.



Figure 1. WordCloud TriggerWord

### 2.2 Text Preprocessing

Feature selection is conducted at this point to identify independent and dependent variables. The sentence is the independent variable, while TriggerWord is the dependent variable or class. The data is then inputted into the text preprocessing stage.

Text preprocessing involves cleaning text data to address unstructured elements, including abbreviations, emoticons, symbols, and numerals. Preprocessing methods are necessary to extract data and transform unorganized text into vector forms. Text preparation has a notable effect on the accuracy of models (Alshdaifat et al., 2021).

The text preprocessing phases in this research involved lowercasing, removing punctuation, stopword removal, frequent and rare word removal, and lemmatization. Data that has been preprocessed is detailed in Table 1:

### 2.3 Resampling

Resampling is the subsequent step in addressing data imbalance concerns. The Random Undersampling (RUS) and Random Oversampling (ROS) methods have been identified as effective resampling techniques for addressing the issue of data imbalance in multiclass cases, according to the research of Hassan et al. (2020) and Pristyanto et al. (2018). This research will combine the features of RUS and ROS to boost the minority class simultaneously and reduce the number of majority classes for better results. A comparison of the amount of data prior to and after resampling utilizing the hybrid approach is depicted in Figure 2.

Figure 2 illustrates the imbalance in the data before resampling, with a considerable difference between the majority and minority classes. The maximum number in the majority class is 205, while the minimum is 5. The first step to address the data imbalance issue involves randomly removing data from the majority class using the RUS approach. The selection process utilized by the RUS technique is determined by Equation (1):

Table 1. Text After Preprocessing

Sentence
downregulation interferon regulatory factor 4 leukemic due hypermethylation cpg motif region
therefore investigate whether irf4 methylation mutation may involve regulation irf4 leukemia
whereas mutation structural rearrangement could exclude cause alter irf4 hematopoietic irf4 methylation status find significantly influence irf4
first treatment irf4 negative lymphoid myeloid monocytic line methylationinhibitor 5 aza 2 deoxycytidine result time concentration dependent increase irf4 mrna level
second use restriction pcr assay bisulfite sequencing identify specifically methylated cpg sit irf4 negative irf4 positive

$$x_{rus} = (x_{ma} - x_i) \times \delta \tag{1}$$

with  $x_{rus}$  is a sample that has been undersampled,  $x_i$  is a majority class sample,  $x_i$  is one of the closest neighbours of the majority sample, and  $\delta$  is a randomly generated number between 0 and 1.

The majority of classes deleted have the highest count, as seen in Figure 2. The current top category totals 129. The following phase randomly replicates the minority class using the ROS approach until a balanced proportion is achieved after subtracting the majority class data. The data-generated procedure utilized by the ROS approach is:

$$x_{new} = x_{mi} + (x_i - x_{mi}) \times \delta \tag{2}$$

with  $x_{new}$  is a generated sample,  $x_i$  is a minority class sample,  $x_i$  is one of the closest neighbours of the minority sample, and  $\delta$  is a randomly generated number between 0 and 1.

After oversampling, the initial data amounted to only 5-119, balanced to 129.

### 2.4 Data Splitting

The typical data split rates are 80% for training and 20% for testing. There is no definitive rule on the optimal ratio for each dataset. Picard and Berk (1990) offer a test set size between 25% and 50%, whereas Afendras and Markatou (2019) advocate a test set size of 50%. On the other hand, extensive numerical studies have indicated that the optimal test data ratio is approximately 30% (Dobbins and Simon, 2011; Nguyen et al., 2021; Pham et al., 2018). Therefore, the dataset in this





research is split into two parts: training data and testing data, with a ratio of 70% for training and 30% for testing.

## 2.5 Feature Extraction

This research utilizes feature extraction with word embedding. Word embedding is a method that utilizes neural networks to transform words into vector representations that encapsulate interconnected meanings. This approach can decrease vector dimensions and enhance the comprehension of semantic meaning (Abdu et al., 2021; Zhu et al., 2023).

The word embedding used is BioWordVec, a pre-trained model specifically trained on biomedical domains such as PubMed and Medical Subject Heading (MeSH). Zhang et al. (2019) developed BioWordVec to address infrequent biomedical domain words that are difficult to learn using traditional word embedding techniques. That is achieved by optimizing the combined goal functions outlined in Equation (3):

$$J = J_{\text{PubMed}} + J_{\text{MeSH}} \quad (3)$$

$$J = \frac{1}{T} \sum_1^T \sum_{c \in C_t} \log p(w_c | w_t) + \frac{1}{N} \sum_1^N \sum_{c \in C_t} \log p(D_c | D_t) \quad (4)$$

$T$  is a measure of PubMed's vocabulary,  $N$  is the total number of MeSH main titles,  $C_t$  is the set of words around it, while  $\log p(w_c | w_t)$  and  $\log p(D_c | D_t)$  it is defined as the probability of an observable word.

## 2.6 Labeling Process

Computer programs can solely handle numerical inputs. All target data will undergo processing through the binarized label stage to transform previously word-form labels into binary format in a matrix that aligns with the number of labels. Table 2 displays the outcome of the binarized label procedure applied to the initial five variables of TriggerWord, which comprises 120 labels. Nevertheless, the overall trigger word count is 119 due to Python's zero-based indexing. The resampling results show that each trigger word is associated with 129 data points.

## 2.7 Hyperparameter Tuning

Identifying the correct set of parameter values is crucial for attaining optimal categorization outcomes. This procedure is called hyperparameter tuning (Yang and Shami, 2020). Early stopping is implemented on the epoch parameter to halt the training process when there is no improvement in the accuracy value. Table 3 contains information about the parameters and their estimated values.

## 2.8 LSTM-CNN Hybrid Model

Hybrid LSTM – CNN combines the LSTM and CNN models. As the name suggests, the LSTM-CNN hybrid model is constructed with the CNN algorithm placed on the final layer following the LSTM network model. The LSTM model retains information in extended text sequences, while the CNN model extracts local text features (Zhang et al., 2018). Figure 3 below illustrates the design of the LSTM-CNN hybrid model:

The study turned text input into word vectors using word embedding and utilized it in a layer known as the embedding layer. The embedding layer functions as a parameterized function that assigns each word in the text to a low-dimensional vector using Equation (5):

$$f : \{x_i | x_i \in S\} \rightarrow R^n \quad (5)$$

The resulting vector is a dense vector consisting of real integers, not limited to 0 and 1. Table 4 provides a visual representation of the word embedding results in vector form:

The subsequent layer employed is the LSTM layer, which takes input from word embedding and produces fresh data via three gates: the forget, input, and output. Equations (6), (7), and (8) describe the formula for each gate:

$$f_t = \sigma(W_f * [h_{(t-1)}, x_t] + b_f) \quad (6)$$

$$i_t = \sigma(W_i * [h_{(t-1)}, x_t] + b_i) \quad (7)$$

$$o_t = \sigma(W_o * [h_{(t-1)}, x_t] + b_o) \quad (8)$$

where  $f_t$  is forget gate,  $i_t$  is the input gate,  $o_t$  is the gate output,  $\sigma$  is a sigmoid function,  $W$  and  $b$  are the weight value and bias value for each gate,  $h_{(t-1)}$  is the output value before the order to  $t$ , and  $x_t$  is the value of enter on order to  $t$ .

The gate output result from Equation (8) still utilizes the sigmoid function. To align the results more closely with the preceding information, they will be multiplied by the cell state using the tanh layer with Equation (9):

$$h_t = o_t * \tanh(C_t) \quad (9)$$

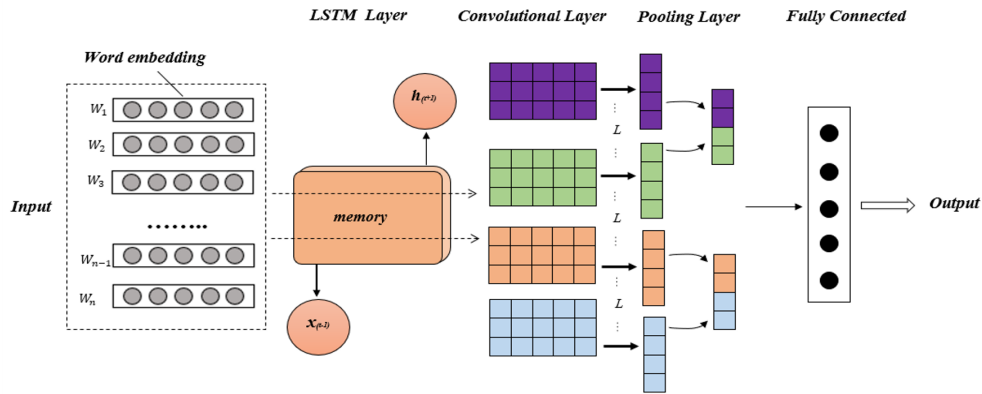
The LSTM layer's output will be inputted into the convolution layer in the CNN model to extract local features. The convolution layer is composed of 64 filters with 5×5 kernel sizes. These filters are responsible for modifying features by moving the input data both horizontally and vertically. The convolution layer functions as a linear mathematical action between matrices, commonly known as a convolution operation (Albawi et al., 2017). In a mathematical context, the convolution operation is commutative and is formulated in Equation (10) as follows:

$$h = f(x, y) * g(x, y) \quad (10)$$

Feature map extraction occurs at the pooling layer, where the feature map dimensions are reduced from the convolution layer, and non-linear operations are performed with the

**Table 2.** Binarized Label

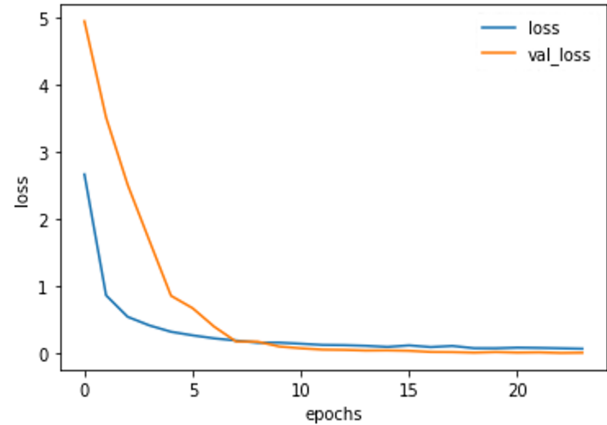
TriggerWord	Label Binarizer									
	0	1	2	3	...	115	116	117	118	119
down-regulation	1	0	0	0	...	0	0	0	0	0
regulation	1	0	0	0	...	0	0	0	0	0
alter	1	0	0	0	...	0	0	0	0	0
negative	1	0	0	0	...	0	0	0	0	0
crosslinking	1	0	0	0	...	0	0	0	0	0



**Figure 3.** LSTM-CNN Hybrid Model Architecture

**Table 3.** Hyperparameter Tuning

Hyperparameter	Definition	Estimated Value
Learning Rate	Parameters that affect the speed of the training process to achieve optimal targets	0.1, 0.01, 0.001
Epoch	Number of iterations or repetitions during the training process to update weights	50, 100
Batch Size	Training sample size per iteration	16, 32, 64



**Figure 4.** Graphics Loss Hybrid LSTM – CNN

ReLU activation function. Equation (11) defines the activation function of ReLU:

$$f(x)_{ReLU} = \begin{cases} x, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases} \quad (11)$$

During the last stage, all neurons from the preceding layer

are transformed into a feature map and linked through a completely connected layer as a vector. This layer employs the softmax activation function to determine output in multiclass situations by computing probability values for each data point across all classes using Equation (12):

$$S(y)_i = \frac{\exp(y_i)}{\sum_{j=1}^n \exp(y_j)} \quad (12)$$

$S(y)_i$  is the probability value of each class in the output layer, with  $n$  being the number of classes in the output layer, and  $(y_i)$

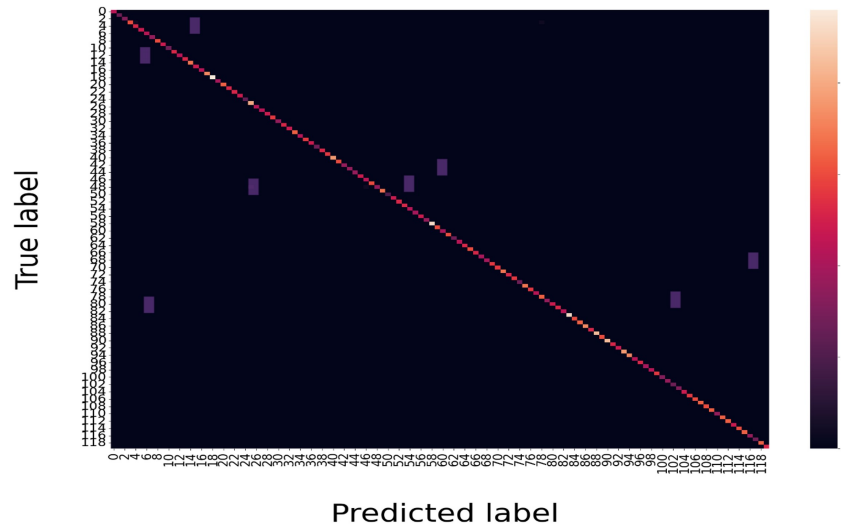


Figure 5. Confusion Matrix LSTM-CNN Hybrid Model

Table 4. Dense Vector Illustration

Substitution	Mutation	Abolish	Bind Protein	Br3	Sequence	Significantly Increase	Hil5	Activity	Activate	
452	170	189	7	61	1902	54	238 19	1616	5	14

representing the output signal in the output layer. The softmax function calculates the probability of each class by utilizing a vector of actual values from all possible classes (Chen et al., 2020).

This model was trained with a learning rate of 0.001, a batch size of 64 and an epoch of 100. In other words, the model divides the data into 64 training samples and undergoes the training process 100 times for each with a speed learning of 0.001. The parameters are optimal values derived from the hyperparameter tuning process.

The learning rate value controls the rate at which the model learns by incorporating new information  $\tau \nabla L(a^{(n-1)})$  with the old information  $a^{(n-1)}$  to calculate new parameter values  $a^{(n)}$ . It is mathematically represented by Equation (13) as follows:

$$a^{(n)} = a^{(n-1)} - \tau \nabla L(a^{(n-1)}) \tag{13}$$

A smaller learning rate value necessitates more iterations or epochs for the model to finish learning. A high learning rate number can prevent the model from converging. Berlyand and Jabin (2023) emphasize the need to find a balance in selecting learning rate values.

A dropout regularization approach is implemented on every hidden layer to prevent overfitting. Adjusting the number of neurons in the buried layer is crucial when constructing networks or models for classification to match the number of labels. With 120 labels in the study, the hidden layer should have a minimum of 120 neurons. The dropout range specified

is 0.2 to 0.3, resulting in the random deactivation of approximately 20% to 30% of neurons, equivalent to around 24 to 36, throughout the training phase.

### 2.9 Evaluation

The confusion matrix in multiclass models has dimensions determined by the number of different classes, denoted as N. Each class will be assessed using accuracy, recall, precision, and f1-score measures. The confusion matrix for multiclass scenarios is created using appropriate measurements  $N \times N$  (Markoulidakis et al., 2021). Table 5 summarises how evaluation metrics are calculated in multiclass models.

## 3. RESULTS AND DISCUSSION

An approach that can be used to determine whether the model has a good performance in classifying text is to observe the loss graph in Figure 4. The LSTM-CNN hybrid model in Figure 4 shows nearly identical loss and validation loss values once it reaches convergence, as evidenced by the proximity of the blue and orange lines. The model demonstrates strong performance in both the training and testing phases.

Table 6 details the accuracy achieved in both the training and testing phases. The table outlines the optimal parameter combination obtained from the hyperparameter tuning process. The data from Table 6 indicates that the suggested model demonstrated outstanding performance, with an accuracy of 97.83% during training and 99.55% during testing. Nevertheless, accuracy is not the sole criterion for assessing classification scenarios involving highly imbalanced data. Overfitting can

**Table 5.** Performance Metrics for Multiclass Confusion Matrix

Metric	Formula
Accuracy	$Acc = \frac{\sum_{i=1}^N TP(L_i)}{\sum_{i=1}^N \sum_{j=1}^N L_{i,j}}$
Recall of Class $L_i$	$R(L_i) = \frac{TP(L_i)}{TP(L_i) + FN(L_i)}$
Precision of Class $L_i$	$P(L_i) = \frac{TP(L_i)}{TP(L_i) + FP(L_i)}$
F1-Score of Class $L_i$	$F1(L_i) = 2 \times \frac{R(L_i) \cdot P(L_i)}{R(L_i) + P(L_i)}$
Recall (Macro Average)	$R_{macro} = \frac{1}{N} \sum_{i=1}^N R(L_i)$
Precision (Macro Average)	$P_{macro} = \frac{1}{N} \sum_{i=1}^N P(L_i)$
F1-Score (Macro Average)	$F1_{macro} = 2 \times \frac{R_{(macro)} \cdot P_{(macro)}}{R_{(macro)} + P_{(macro)}}$

**Table 6.** LSTM-CNN Hybrid Model Accuracy

Model	Learning Rate	Epoch	Train - Test		Accuracy	Test Accuracy
			Batch Size	Time		
LSTM – CNN	0.01	24	16	6min 45s	97.83	99.55

lead to high accuracy, especially when there is data imbalance. Hence, it is essential to consider more thorough assessments like precision, recall, and f1-score.

The most suitable evaluation metric for analyzing a model’s performance in identifying imbalanced data is the f1-score. The f1-score evaluates the precision and recall of a model by taking into account the distribution of data, unlike accuracy,

which does not incorporate data distribution. A higher f1-score indicates better model quality in categorizing imbalanced data. Table 7 summarises evaluation measure values, including precision, recall, and f1-score.

Table 7 displays the average macro values for metrics like precision and recall used to assess the hybrid LSTM-CNN model, which are 99% and 100%, respectively. The high pre-



**Table 7.** Precision, Recall, and F1-Score Model Hybrid LSTM-CNN

Macro Average	Precision	Recall	F1-Score
	0.99	1.00	0.99

recision and recall scores suggest the model can accurately categorize data within each class. These results suggest that the suggested model performs optimally by accurately grouping 99% of the data in each class. The hybrid LSTM-CNN model achieved a f1-score of 99%, indicating exceptional and precise performance in classifying imbalanced data. The hybrid LSTM-CNN model achieved great accuracy without overfitting.

Figure 5 displays the evaluation results of the LSTM-CNN hybrid model in a confusion matrix. The confusion matrix displays correctly classified classes within the diagonal lines and incorrectly classified classes outside the diagonal lines.

Figure 5 demonstrates that the LSTM-CNN hybrid model effectively classified most classes in biomedical texts. The model's classification accuracy is demonstrated in the following example:

- Sentence: "IL-10 has been shown to block the antigen-specific T-cell cytokine response by inhibiting the CD28 signalling pathway."
  - TriggerWord Prediction: Block
  - Interpretation: This sentence states that IL-10 can inhibit the CD28 signalling pathway to halt the cytokine response of T cells activated by antigens. The prediction of the trigger word "block" aligns with the context, suggesting the action of halting or blocking the cytokine response of the T cell.
- Sentence: "IL-10 is a key cytokine in regulating inflammatory responses, mainly by inhibiting the production and function of proinflammatory cytokines."
  - TriggerWord Prediction: Regulate
  - Interpretation: This sentence states that IL-10 plays a crucial role in controlling inflammation by suppressing the synthesis and activity of proinflammatory cytokines. The word "regulate" in this context suggests that IL-10 manages or controls inflammatory processes in the body.

#### 4. CONCLUSIONS

The LSTM-CNN hybrid model effectively classified imbalanced data with exceptional performance, achieving a 99% f1-score. This model demonstrated superior accuracy to the classification model suggested by earlier studies due to its utilization of BioWordVec's pre-trained word embedding as a word representation method. Thus, pre-trained word embeddings specific to the study topic can enhance the model's accuracy in classification tasks. This research can potentially be an alternate way of discovering information about leukemia, as determined

by the accuracy and f1-score achieved by the model.

#### 5. ACKNOWLEDGMENT

The author would like to thank the Research Center, Universitas Lampung, through a PDD Grant Scheme for the academic year 2023. Also, the author would like to thank the Doctor of MIPA Program Study at the Faculty of Mathematics and Natural Sciences, Universitas Lampung.

#### REFERENCES

- Abdu, S. A., A. H. Yousef, and A. Salem (2021). Multimodal Video Sentiment Analysis Using Deep Learning Approaches, a Survey. *Information Fusion*, **76**; 204–226
- Abdulrauf Sharifai, G. and Z. Zainol (2020). Feature Selection for High-Dimensional and Imbalanced Biomedical Data Based on Robust Correlation Based Redundancy and Binary Grasshopper Optimization Algorithm. *Genes*, **11**(7); 717
- Afendras, G. and M. Markatou (2019). Optimality of Training/test Size and Resampling Effectiveness in Cross-Validation. *Journal of Statistical Planning and Inference*, **199**; 286–301
- Ahmed, N., A. Yigit, Z. Isik, and A. Alpkocak (2019). Identification of Leukemia Subtypes from Microscopic Images Using Convolutional Neural Network. *Diagnostics*, **9**(3); 104
- Akpatsa, S. K., X. Li, and H. Lei (2021). A Survey and Future Perspectives of Hybrid Deep Learning Models for Text Classification. In *Artificial Intelligence and Security: 7th International Conference, ICAIS 2021, Dublin, Ireland, July 19–23, 2021, Proceedings, Part I 7*. Springer, pages 358–369
- Albawi, S., T. A. Mohammed, and S. Al-Zawi (2017). Understanding of a Convolutional Neural Network. In *2017 International Conference on Engineering and Technology (ICET)*. Ieee, pages 1–6
- ALRashdi, R. and S. O'Keefe (2019). Deep Learning and Word Embeddings for Tweet Classification for Crisis Response. *ArXiv Preprint ArXiv:1903.11024*
- Alshdaifat, E., D. Alshdaifat, A. Alsarhan, F. Hussein, and S. M. F. S. El-Salhi (2021). The Effect of Preprocessing Techniques, Applied to Numeric Features, on Classification Algorithms' Performance. *Data*, **6**(2); 11
- Asudani, D. S., N. K. Nagwani, and P. Singh (2023). Impact of Word Embedding Models on Text Analytics in Deep Learning Environment: A Review. *Artificial Intelligence Review*, **56**(9); 10345–10425
- Berlyand, L. and P.-E. Jabin (2023). *Mathematics of Deep Learning: An Introduction*. de Gruyter
- Bojanowski, P., E. Grave, A. Joulin, and T. Mikolov (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**; 135–146
- Chen, C. C., Z. Liu, G. Yang, C. C. Wu, and Q. Ye (2020). An Improved Fault Diagnosis Using 1D-convolutional Neural Network Model. *Electronics*, **10**(1); 59
- Dobbin, K. K. and R. M. Simon (2011). Optimally Splitting

- Cases for Training and Testing High Dimensional Classifiers. *BMC Medical Genomics*, **4**(1); 1–8
- Du, M., W. Chen, K. Liu, L. Wang, Y. Hu, Y. Mao, X. Sun, Y. Luo, J. Shi, and K. Shao (2022). The Global Burden of Leukemia and Its Attributable Factors in 204 Countries and Territories: Findings from the Global Burden of Disease 2019 Study and Projections to 2030. *Journal of Oncology*, **2022**; 1–14
- Fotouhi, S., S. Asadi, and M. W. Kattan (2019). A Comprehensive Data Level Analysis for Cancer Diagnosis on Imbalanced Data. *Journal of Biomedical Informatics*, **90**; 103089
- Hasib, K. M., S. Azam, A. Karim, A. Al Marouf, F. J. M. Shamrat, S. Montaha, K. C. Yeo, M. Jonkman, R. Alhajib, and J. G. Rokne (2023). MCNN-LSTM: Combining CNN and LSTM to Classify Multi-Class Text in Imbalanced News Data. *IEEE Access*, **11**; 93048–93063
- Hassan, H., N. B. Ahmad, and S. Anuar (2020). Improved Students' Performance Prediction for Multi-Class Imbalanced Problems Using Hybrid and Ensemble Approach in Educational Data Mining. In *Journal of Physics: Conference Series*, volume 1529. IOP Publishing, page 052041
- Hung, B. T. (2019). Document Classification by Using Hybrid Deep Learning Approach. In *Context-Aware Systems and Applications, and Nature of Computation and Communication: 8th EAI International Conference, ICCASA 2019, and 5th EAI International Conference, ICTCC 2019, My Tho City, Vietnam, November 28-29, 2019, Proceedings*. Springer, pages 167–177
- Jain, V. and K. L. Kashyap (2023). Ensemble Hybrid Model for Hindi COVID-19 Text Classification with Metaheuristic Optimization Algorithm. *Multimedia Tools and Applications*, **82**(11); 16839–16859
- Korde, V. and C. N. Mahender (2012). Text Classification and Classifiers: A Survey. *International Journal of Artificial Intelligence & Applications*, **3**(2); 85
- Kowsari, K., K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown (2019). Text Classification Algorithms: A Survey. *Information*, **10**(4); 150
- Lai, S., L. Xu, K. Liu, and J. Zhao (2015). Recurrent Convolutional Neural Networks for Text Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29
- Lan, Y., Y. Hao, K. Xia, B. Qian, and C. Li (2020). Stacked Residual Recurrent Neural Networks with Cross-Layer Attention for Text Classification. *IEEE Access*, **8**; 70401–70410
- Li, Q., H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He (2022). A Survey on Text Classification: From Traditional to Deep Learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, **13**(2); 1–41
- Li, W., P. Liu, Q. Zhang, and W. Liu (2019). An Improved Approach for Text Sentiment Classification Based on a Deep Neural Network Via a Sentiment Attention Mechanism. *Future Internet*, **11**(4); 96
- Li, Y. and T. Yang (2018). Word Embedding for Understanding Natural Language: A Survey. *Guide to Big Data Applications*, **26**; 83–104
- Markoulidakis, I., G. Kopsiaftis, I. Rallis, and I. Georgoulas (2021). Multi-Class Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem. In *The 14th Pervasive Technologies Related to Assistive Environments Conference*. pages 412–419
- Masic, I. and A. Ferhatovica (2012). Review of Most Important Biomedical Databases for Searching of Biomedical Scientific Literature. *Donald School Journal of Ultrasound in Obstetrics and Gynecology*, **6**; 343–61
- Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013). Distributed Representations of Words and Phrases and Their Compositionality. *Advances in Neural Information Processing Systems*, **26**; 1–9
- Nguyen, Q. H., H.-B. Ly, L. S. Ho, N. Al-Ansari, H. V. Le, V. Q. Tran, I. Prakash, and B. T. Pham (2021). Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil. *Mathematical Problems in Engineering*, **2021**; 1–15
- Pennington, J., R. Socher, and C. D. Manning (2014). Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543
- Pham, B. T., I. Prakash, A. Jaafari, and D. T. Bui (2018). Spatial Prediction of Rainfall-Induced Landslides Using Aggregating One-Dependence Estimators Classifier. *Journal of the Indian Society of Remote Sensing*, **46**; 1457–1470
- Picard, R. R. and K. N. Berk (1990). Data Splitting. *The American Statistician*, **44**(2); 140–147
- Pristyanto, Y., I. Pratama, and A. F. Nugraha (2018). Data Level Approach for Imbalanced Class Handling on Educational Data Mining Multiclass Classification. In *2018 International Conference on Information and Communications Technology (ICOLACT)*. IEEE, pages 310–314
- Rabut, B. A., A. C. Fajardo, and R. P. Medina (2019). Multi-Class Document Classification Using Improved Word Embeddings. In *Proceedings of the 2nd International Conference on Computing and Big Data*. pages 42–46
- Rathnayaka, P., S. Abeysinghe, C. Samarajeewa, I. Manchanayake, and M. Walpola (2018). Sentylic at IEST 2018: Gated Recurrent Neural Network and Capsule Network Based Approach for Implicit Emotion Detection. *arXiv preprint arXiv:1809.01452*; 254–259
- Rios, A. and R. Kavuluru (2015). Convolutional Neural Networks for Biomedical Text Classification: Application in Indexing Biomedical Articles. In *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics*. pages 258–267
- Sabbeh, S. F. and H. A. Fasihuddin (2023). A Comparative Analysis of Word Embedding and Deep Learning for Arabic Sentiment Classification. *Electronics*, **12**(6); 1425
- Saeed, A., S. Shoukat, K. Shehzad, I. Ahmad, A. Eshmawi, A. H. Amin, and E. Tag-Eldin (2022). A Deep Learning-Based Approach for the Diagnosis of Acute Lymphoblastic Leukemia. *Electronics*, **11**(19); 3168

- Tai, K. S., R. Socher, and C. D. Manning (2015). Improved Semantic Representations from Tree-Structured Long Short-Term Memory Networks. *arXiv preprint arXiv:1503.00075*, **1**; 1556–1566
- Tasdelen, A. and B. Sen (2021). A Hybrid CNN-LSTM Model for Pre-miRNA Classification. *Scientific Reports*, **11**(1); 14125
- Tober, M. (2011). Pubmed, Scencedirect, Scopus or Google Scholar—Which Is the Best Search Engine for an Effective Literature Research in Laser Medicine? *Medical Laser Application*, **26**(3); 139–144
- Wang, K., P. Zhang, and J. Su (2020). A Text Classification Method Based on the Merge-LSTM-CNN Model. In *Journal of Physics: Conference Series*, volume 1646. IOP Publishing, page 012110
- Weeks, G. R. (1864). Pyæmia, or Leukæmia. *The Boston Medical and Surgical Journal*, **69**(23); 459–463
- Yang, L. and A. Shami (2020). On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing*, **415**; 295–316
- Zhang, J., Y. Li, J. Tian, and T. Li (2018). LSTM-CNN Hybrid Model for Text Classification. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. IEEE, pages 1675–1680
- Zhang, Y., Q. Chen, Z. Yang, H. Lin, and Z. Lu (2019). BioWordVec, Improving Biomedical Word Embeddings with Subword Information and MeSH. *Scientific Data*, **6**(1); 52
- Zhao, H., J. Xie, and H. Wang (2022). Graph Convolutional Network Based on Multi-Head Pooling for Short Text Classification. *IEEE Access*, **10**; 11947–11956
- Zhou, P., Z. Qi, S. Zheng, J. Xu, H. Bao, and B. Xu (2016). Text Classification Improved by Integrating Bidirectional LSTM with Two-Dimensional Max Pooling. *arXiv preprint arXiv:1611.06639*; 3485–3495
- Zhu, L., Z. Zhu, C. Zhang, Y. Xu, and X. Kong (2023). Multimodal Sentiment Analysis Based on Fusion Methods: A Survey. *Information Fusion*, **95**; 306–325