ISRG Journal of Multidisciplinary Studies (ISRGJMS)

OPEN ACCESS

ISRG PUBLISHERS

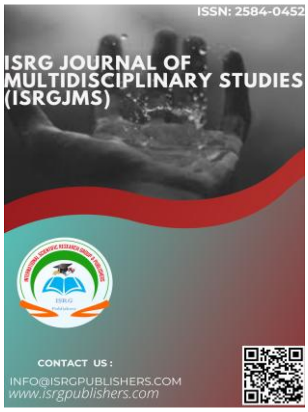
Abbreviated Key Title: isrg j. multidiscip. Stud.

ISSN: 2584-0452 (Online)

Journal homepage: https://isrgpublishers.com/isrgjms/

Volume – III, Issue -II (February) 2025

Frequency: Monthly



Deep Learning-Driven Comparative Study of Word Embedding Techniques: Word2Vec, GloVe, and FastText in Health Condition Reviews

Dian Kurniasari[1], Fiqih Aulia Pradana[2], Wamiliana[3]

[1, 2,3] Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Indonesia

| **Received:** 04.02.2025 | **Accepted:** 08.02.2025 | **Published:** 19.02.2025

*__Corresponding author:__ Dian Kurniasari

Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Indonesia

Abstract

Health products, including medications, play a crucial role in public health. Reviews from individuals who have experienced illnesses offer valuable insights for the community in selecting appropriate treatments. In statistical analysis and classification methods, these reviews are processed using Natural Language Processing (NLP), where text mining is pivotal in data processing.

This study aims to integrate Word Embedding techniques with Long Short-Term Memory (LSTM) to classify health-related reviews effectively using machine learning approaches, specifically through sentiment analysis. Word Embedding techniques, such as Word2Vec, FastText, and GloVe, are employed to analyze the structure and context of words. The dataset consisted of 215,063 reviews, separated into 161,297 training samples and 53,766 test samples, covering 13 different health conditions. Training and validation processes were conducted to assess the effectiveness of each method in combination with LSTM. The training and validation accuracy rates achieved were 95.09% for Word2Vec, 94.88% for GloVe, and 95.07% for FastText in training, with validation accuracy rates of 94.47%, 94.17%, and 95.44%, respectively. Test accuracy rates confirmed these findings, with 85.20% for Word2Vec, 84.19% for GloVe, and 86.22% for FastText. FastText outperformed the other methods in effectively categorizing health-related reviews. The results indicate that the integration of Word Embedding techniques and LSTM is effective in classifying health-related reviews, with FastText showing superior performance.

Keywords: *Classification; Drug Review; Deep Learning, LSTM, Word Embedding*

DOI: 10.5281/zenodo.14890438

I. INTRODUCTION

Health products, such as medications—particularly over-the-counter drugs generally regarded as safe—are now widely accessible through pharmacies and various e-commerce platforms. Consequently, the public can obtain these medications without a doctor's prescription. However, improper use of these drugs without adequate medical knowledge poses significant health risks. As a result, individuals often rely on reviews or recommendations from previous users. These reviews, which encapsulate users' experiences and perspectives, typically detail the ailments addressed and the medication's effectiveness, aiding others in evaluating the drug's suitability for their conditions.

This study aims to classify health-related reviews using machine learning approaches, specifically through sentiment analysis. One critical step in this process is the transformation of textual data into actionable insights, which involves converting unstructured text into structured categories or labels—a task known as text classification (Bashir et al., 2021; Rao et al., 2021; Zhu & Lei, 2022). In this context, the classification of reviews is often referred to as sentiment analysis or sentiment classification and is widely utilized in NLP applications (Bhavani & Santhosh Kumar, 2021).

NLP is a branch of Machine Learning (ML) that enables computers to comprehend spoken and written human language. With the rapid advancements in ML, Recurrent Neural Networks (RNNs) have emerged as a popular approach for handling complex textual or sequential data (Cao et al., 2021). RNNs are a Deep Learning (DL) architecture that processes inputs sequentially to capture information over time. Despite their effectiveness in various DL tasks, RNNs face challenges due to the vanishing gradient problem, which limits their ability to retain information over extended periods. Therefore, finding methods to mitigate the impact of vanishing gradients during training is crucial. Modifying network structures is a common approach to addressing this issue, as exemplified by the Long Short-Term Memory (LSTM) model proposed by Hochreiter and Schmidhuber (Hochreiter & Schmidhuber, 1997).

LSTM was developed to address the vanishing gradient problem commonly encountered in conventional RNNs. LSTM's strength lies in its ability to retain crucial information while discarding irrelevant data facilitated by its gating mechanisms. These gates include the input, forget, and output gates, which enable LSTM to manage long-term dependencies effectively. The architecture of LSTM, composed of multiple neural units, is specifically designed to enhance its capability to process and comprehend contextual information (Wang & Li, 2022).

One crucial step before classification is text representation, where textual data is transformed into vectors that serve as inputs for learning models. Generally, there are two main approaches to representing text. The first approach is non-contextual, such as the Bag of Words (BOW) method. BOW focuses on the frequency of word or phrase occurrences in the text, and therefore, it does not provide information about the structure, sequence, semantics, or context of words within a sentence. Alternatively, the second approach, contextual approaches, includes methods like word embedding and pre-trained word embedding. Word embedding is a technique that converts words into n-dimensional vectors. On the other hand, pre-trained word embeddings have been trained on specific-domain corpora, allowing them to capture word context more effectively (Dogra et al., 2022).

Pretrained word embeddings such as Word2Vec, GloVe, and FastText serve as notable examples. Word2Vec, developed by Mikolov et al. (Mikolov et al., 2013) in 2013, offers two main learning models: Continuous Bag of Words (CBOW) and Skip-gram. CBOW forecasts a word based on its surrounding context, whereas Skip-gram estimates the context from a specified word. Concurrently, GloVe was introduced by Pennington et al. (Pennington et al., 2014) at Stanford University within the same year, utilizing ratios of word co-occurrence probabilities. In 2017, Facebook released FastText, an advanced word vector model similar to Word2Vec, which includes the additional feature of incorporating subword information.

Word embeddings such as Word2Vec, GloVe, and FastText play a crucial role in ML, particularly within the realms of NLP and DL, including RNN and their advancements, such as LSTM. These embeddings are widely used and have been applied across various domains. For instance, their application can be seen in sentiment analysis of hotel reviews (Imaduddin et al., 2019; Nawangsari et al., 2019), Bengali text classification (Hossain & Timmer, 2021), hoax detection in Indonesian news (Adipradana et al., 2021), machine translation (Nath et al., 2024; Sitender et al., 2023), cyberbullying detection (Nasution & Setiawan, 2023), and sentiment analysis of Spotify users (Anjani & Nurramdhani, 2024). However, their application in healthcare, particularly for classifying health-related reviews, remains underexplored.

This research fills this gap by comparing the performance of Word2Vec, GloVe, and FastText embeddings within an LSTM model for health-related review classification. The study contributes to the broader understanding of applying pre-trained word embeddings in the healthcare sector, paving the way for further advancements and applications in this critical domain. Practically, it enhances the accuracy of medication recommendations on healthcare platforms and e-commerce sites, supporting consumers in making informed health decisions. Furthermore, the findings provide valuable insights for healthcare practitioners to optimize patient care based on user feedback analysis.

II. LITERATURE REVIEW

A. *Word2Vec*

Word2Vec, a word embedding algorithm introduced by Mikolov et al. (Mikolov et al., 2013), transforms words into vector representations and is extensively utilized in natural language processing (NLP) for its capacity to capture the semantic meaning of words. The algorithm operates via a neural network architecture that includes hidden layers and fully connected layers, where the trained weights of the hidden layers are employed to convert words into vectors. These weights form a lookup table, with each row representing a word and each column corresponding to a word vector. The model learns linguistic patterns through linear relationships between vectors, leveraging the context of surrounding words to infer semantic meaning. As an unsupervised learning approach, Word2Vec effectively captures both the semantic and syntactic relationships between words within a text corpus (Nurdin et al., 2020).

B. *GloVe*

GloVe, or Global Vectors for Word Representation, is a technique for generating word embeddings used to analyze word similarities semantically and identify word entities. Developed by Pennington et al. (Pennington et al., 2014), GloVe employs an unsupervised

DOI: 10.5281/zenodo.14890438

learning approach. It examines word representations based on the patterns of word co-occurrence within a corpus, eliminating the need for labelled training data. By analyzing the co-occurrence relationships between words, GloVe uncovers semantic connections, thereby enhancing performance in tasks such as word analogy.

In practice, GloVe constructs a co-occurrence matrix to capture the semantic relationships between words. This matrix records the frequency of co-occurrences between pairs of words in the corpus. For instance, in the sentences "I love math" and "I love my parents," the matrix reflects how often the words appear together. Specifically, the words "I" and "love" co-occur twice within a given window. By leveraging global statistics from the corpus, GloVe ensures that the resulting word vectors accurately and effectively represent semantic relationships. This enables GloVe to produce superior embeddings for various text-based tasks.

C. FastText

Facebook developed FastText as an advanced library for word embedding, building upon the Word2Vec model. Unlike Word2Vec, which treats words as discrete units, FastText considers sub-word information, representing each word as a set of n-gram characters. This allows it to capture more nuanced representations, particularly for words with affixes, such as "describe" and "image", by leveraging semantic similarities through n-grams. FastText supports both the CBOW and Skip-gram algorithms and is particularly adept at handling out-of-vocabulary words.

In contrast to Word2Vec, which processes words individually, FastText analyzes words in a co-occurrence manner, considering the context of sequential words in the text. This enables it to generate more accurate vector representations, particularly for languages with expansive vocabularies. Furthermore, FastText can generate representations for words not present in the training dataset, although it requires more computational time than Word2Vec.

FastText also integrates linear classifiers like multinomial logistic regression, utilizing a softmax layer to generate class probability distributions. By implementing a Huffman Coding Tree-based hierarchical softmax, FastText accelerates computation when dealing with large class sets, significantly reducing computational complexity. As a result, FastText provides an efficient and scalable solution for word representation across diverse linguistic applications (Joulin et al., 2017).

III. RELATED WORKS

Mahboob and Ali (2018) extracted customer opinions on pharmaceutical products from two websites. Their study aimed to classify sentiments as positive, negative, or neutral using a lexicon-based approach applied on a limited scale. The experimental results revealed that this approach achieved the highest precision and satisfactory performance in specific drug categories, while others still have room for improvement. Mishra (2020) performed a sentiment analysis on drug reviews employing various boosting algorithms, including Light Gradient Boosting Machine (LGBM), XGBoost, and CatBoost. The sentiment labels were condensed into two categories: positive and negative. The accuracy rates achieved were 88.8% for LGBM, 76.8% for XGBoost, and 88.2% for CatBoost.

Rani and Reddy (2022) investigated sentiment analysis of drug reviews to develop a recommendation system, employing a range of machine learning classifiers, including Logistic Regression, Perceptron, Multinomial Naive Bayes, Ridge Classifier, Stochastic Gradient Descent, and LinearSVC, alongside methods like Bag of Words (BoW) and TF-IDF. They also tested classifiers such as Decision Tree, Random Forest, LightGBM (LGBM), and CatBoost using the Word2Vec technique. The performance of each method was assessed based on five metrics: precision, recall, F1-score, accuracy, and AUC score. Their findings revealed that the LinearSVC with TF-IDF achieved the highest accuracy at 93%, while the Decision Tree classifier with Word2Vec demonstrated the lowest accuracy at 78%.

Suhartono et al. (2022) proposed a DL-based approach for sentiment analysis of product reviews derived from the UCI repository, integrating GloVe word embeddings with a CNN. This research assessed the performance of both Word2Vec and GloVe embeddings within a CNN architecture, and it compared these embeddings with BERT and RoBERTa models. The findings revealed that while BERT demonstrated superior performance during the training and validation phases, the CNN model utilizing GloVe embeddings achieved the highest testing accuracy of 84.87%. This study underscores the potential of employing DL models to enhance sentiment analysis within the context of pharmaceutical product reviews.

Similarly, Tukino et al. (2024) highlighted the significance of patient feedback in enhancing the quality of services provided by community health centres (Puskesmas). This study utilized an LSTM model along with Keras word embeddings, optimized with Adadelta and Adamax, to analyze Twitter reviews. By employing a confusion matrix for evaluation, the research achieved a precision of 76%, a recall of 69%, and an accuracy of 71%, demonstrating effectiveness in classifying public opinion. These results emphasize the necessity for high accuracy to support decision-making in service improvement and recommend further research into parameter selection and dataset optimization. A summary of related work is presented in Table 1.

Despite extensive research on sentiment analysis, particularly regarding drug reviews with various methodologies and algorithms, significant gaps remain to be addressed. These include the exploration of alternative word embeddings and the refinement of sentiment classification schemes. Many existing studies rely on word embeddings that may not be ideal for analyzing sentiments in drug reviews and often categorize sentiments into overly simplistic classes, such as positive, negative, and neutral, or just positive and negative. This study investigates the application of different word embeddings and more comprehensive sentiment classification to extract richer insights from review data. The research aims to enhance our understanding and analysis of sentiment in drug reviews by focusing on these areas and providing more detailed and informative sentiment classifications.

TABLE 1

An Overview of Relevant Research

Research	Data	Research
Mahboob and Ali (2018)	The data was obtained from two online platforms focused on pharmaceutical products,	Evaluating the quality of medications through lexicon-based sentiment analysis involves assessing

DOI: 10.5281/zenodo.14890438

	specifically medications for sinus allergies, coughs, colds, fever, etc.	reviews using positive, neutral, or negative terminology to assist in purchase decisions.
Mishra (2020)	The dataset from the UCI Machine Learning Repository contains patient reviews of medications, rated on a scale from 1 to 10 based on their experiences.	Mishra employs gradient boosting algorithms such as LGBM, XGBoost, and CatBoost in sentiment analysis, simplifying sentiment labels to positive and negative.
Rani and Reddy (2022)	The dataset comprises medication reviews addressing specific conditions, including acne, hypertension, contraception, depression, and pain.	Developing a medication recommendation system based on patient reviews utilizing various vectorization techniques, including BOW, TF-IDF, and Word2Vec, to suggest optimal medications based on the highest positive sentiment scores for specific conditions.
Suhartono et al. (2022)	Medication reviews were sourced from the pharmaceutical information website drugs.com.	Developing a DL-based sentiment analysis approach for pharmaceutical product reviews by integrating GloVe and Word2Vec embeddings with CNN.
Tukino et al. (2024)	Patient reviews of community health centres (Puskesmas) were collected from Twitter.	Classifying and analyzing patient sentiment regarding public health centre services on Twitter to enhance service quality using a DL model, specifically LSTM.
Proposed Method	The research dataset, sourced from the UCI Repository in CSV format, contains 13 disease condition labels with over 2,000 reviews.	This study compares the performance of three-word embedding techniques in an LSTM model to categorize medication reviews by disease condition, aiding users and companies in understanding patient needs and developing more effective product strategies.

IV. Methods

This methodology section outlines the steps taken to collect and process data for classification using the LSTM model with word embedding techniques such as Word2Vec, GloVe, and FastText. The first step involves data collection and the application of pre-processing, which includes handling missing values, duplicate data, stopwords, and lemmatization. The data is then visualized for more straightforward interpretation. Next, the data undergoes tokenization and the creation of a word corpus in the form of numerical vectors. The subsequent step is data balancing through Random Over Sampling (ROS) and the division of data into training and testing sets. An LSTM model is constructed, followed by parameter tuning using hyperparameter optimization. Model evaluation is conducted to determine the best performance based on accuracy, precision, recall, F1-Score, and the confusion matrix. These steps are summarized in the research flowchart in Figure 1.

A. Data Collecting

The data used in this study was collected from the UCI Repository through the Drug Review Dataset link. This dataset consists of six primary attributes: drugName, which refers to the name of the drug used; condition, representing the medical condition experienced by the respondent; review, which is the respondent's feedback on the drug consumed; rating, indicating the respondent's evaluation of the drug; date, the date the review was submitted; and usefulCount, which shows the number of other respondents who found the review relevant or helpful.

The dataset comprises qualitative text data with a total of 215,063 reviews, divided into 161,297 for training and 53,766 for testing. The primary focus of this research is to classify using the review and condition attributes, with the data being categorized as supervised learning due to the presence of labels. The condition attribute serves as the label for text classification using word embedding techniques such as Word2Vec, GloVe, and FastText, with the LSTM method.

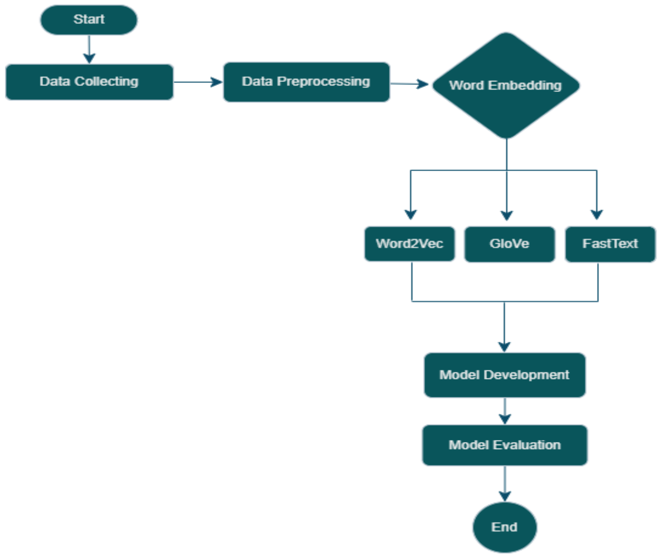


Figure 1. Research methods.

DOI: 10.5281/zenodo.14890438

The number of labels in the training data was adjusted by removing those with low frequencies to enhance model performance. A total of 13 condition labels with over 2,000 reviews were retained, while data with condition labels falling below this threshold, as well as data unrelated to the respondents' medical conditions, were excluded. The dataset was downloaded in CSV format for analysis. A sample of the research data is presented in Table 2.

B. Data Pre-processing

The initial step in the data processing pipeline involves text pre-processing, which is essential in ML as it transforms or encodes data to facilitate efficient analysis by model algorithms (Maharana et al., 2022). HaCohen-Kerner et al. (2020) highlight that employing pre-processing techniques can significantly enhance data quality, particularly for text classification tasks. This process involves removing 'noise' from the dataset, such as spelling errors, duplicate letters, and unclear acronyms. Techniques including the removal of stopwords, elimination of punctuation, stemming, and lemmatization are applied to improve the dataset's quality, making it more suitable for effective text classification.

Following the pre-processing phase, attention is given to data balancing, especially in datasets characterized by imbalances between classes. It is crucial to directly address these discrepancies within the training data to ensure robust model performance. Resampling techniques are employed, where undersampling reduces the number of examples in the majority class, and oversampling increases instances in the minority class (Sasada et al., 2020). Oversampling is often preferred as it helps mitigate data loss by generating additional samples for the minority class, effectively balancing its size with that of the majority class. Among the various oversampling methods, ROS is utilized in this study, which enhances minority class data by adding random duplicates of existing samples without introducing new variations (Hayaty et al., 2021).

Finally, data splitting is a foundational step in the machine learning data processing pipeline (Antanasijević et al., 2020). This study adopts a standard split ratio of 80% for training and 20% for testing, aligning with established practices in prior research that utilize similar proportions for training and testing datasets (Ahmed et al., 2022; Grbčić et al., 2022; Islam Khan et al., 2022; Rahman, 2019; Rezaie-Balf et al., 2020; Sajib et al., 2023; Uddin et al., 2024). This strategic division ensures that the model is adequately trained while maintaining a robust testing framework to evaluate its performance effectively.

C. *Model Development*

The development of the model begins with the selection of an appropriate word embedding technique, which is crucial for achieving optimal results in sentiment analysis (Asudani et al., 2023). Word embedding represents words as vectors within a defined number of dimensions, denoted as n, effectively capturing complex semantic meanings and transforming words into numerical vectors that reflect their contextual and semantic relationships. To identify the most effective model, various word embedding techniques are explored, including Word2Vec, GloVe, and FastText.

Word2Vec employs an Artificial Neural Network (ANN) model that captures semantic information from phrases within a corpus using unlabeled training data. It calculates the cosine similarity between word vectors to determine the degree of semantic similarity. Comprising two architectures—Skip Gram and Continuous Bag of Words (CBOW)—Word2Vec is widely utilized in various language processing applications, such as sentiment classification, Named Entity Recognition (NER), Part-of-Speech (POS) tagging, and document analysis (Sivakumar et al., 2020). GloVe shares similarities with Word2Vec, as both generate word vectors by analyzing word co-occurrence data; however, they differ fundamentally: Word2Vec operates as a predictive model, generating vectors to enhance the accuracy of error prediction, while GloVe is a counting model that produces vectors by reducing the dimensionality of the co-occurrence count matrix (Mohamed et al., 2020).

FastText, on the other hand, was developed to address the limitations of Word2Vec in handling morphological characteristics by examining subwords within individual words. This approach analyzes n-grams of characters that make up a word, allowing FastText to generate vector representations for each n-gram and calculate a comprehensive word vector by summing these vectors. One of FastText's key advantages is its ability to create new word vectors for words not present in the training data, including those previously unencountered, thus incorporating morphological features that enhance the precision of syntactic analysis (Choi & Lee, 2020).

After selecting the word embedding technique, the next step involves hyperparameter tuning, which is essential for DL approaches as it significantly impacts model performance. Hyperparameters are specified before the training process begins and remain unchanged during the training. Methods such as Grid Search are utilized to identify the optimal hyperparameter values, allowing for a systematic exploration of potential configurations (Bartz-Beielstein et al., 2023).

TABLE 2

DRUG REVIEW DATA SAMPLES

uniqueID	drugName	Condition	Review	rating	date	usefulCount
206461	Valsartan	Left Ventricular Dysfunction	"It has no side effect, I take it in combination ...."	9	20-May-12	27
95260	Guanfacine	ADHD	"My son is halfway through his fourth week of ...."	8	27-Apr-10	129
92703	Lybrel	Birth Control	"I used to take another oral contraceptive, which had ...."	5	14-Dec-09	17
138000	Ortho Evra	Birth Control	"This is my first time using any form of birth…"	8	3-Nov-15	10

DOI: 10.5281/zenodo.14890438

35696	Buprenorphine /naloxone	Opiate Dependence	"Suboxone has completely turned my life around…"	9	27-Nov-16	37
……..	……….	…………	……………	…..	……	……..
191035	Campral	Alcohol Dependence	"I wrote my first report in Mid-October of 2014.."	10	31-May-15	125
127085	Metoclopramide	Nausea/ Vomiting	"I was given this in IV before surgery. I…"	1	1-Nov-11	34
187382	Orencia	Rheumatoid Arthritis	"Limited improvement after 4 months, develop…"	2	15-Mar-14	35
47128	Thyroid desiccated	Underactive Thyroid	"Iand#039;ve been on thyroid medication 49 years…"	10	19-Sep-15	79
215220	Lubiprostone	Constipation, Chronic	"Iand#039;ve had chronic constipation all my adu…"	10	13-Dec-14	116

The final component of model development is fitting the LSTM model, which is the optimal type of RNN for tasks involving sequential data processing, such as time series and text data (Shobana & Murali, 2021). The LSTM model excels in its ability to store and modify information derived from previous data through its cell state and three gate mechanisms: input, forget, and output gates (Balci et al., 2024; Gao et al., 2020).

Three gates primarily govern the operation of LSTM: input, forget, and output, which are aided by an activation function. This strategy operates by maintaining "memory" within the cell state by the selective elimination of irrelevant information using the forget gate (f_t), incorporating new information through the input gate i_t, and generating a hidden state at each time step using the output gate (Chiam et al., 2023). The mathematical formulation of the strategy referred to as "gates" can be expressed using Equations (1), (2), and (3):

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

$$i_t = \sigma\,(W_i.[h_{t-1}, x_t] + b_i) \quad (2)$$

$$o_t = \sigma\,(W_o.[h_{t-1}, x_t] + b_o) \quad (3)$$

The activation function plays a critical role in determining the efficacy of ML models, facilitating the model's ability to extract abstract features via non-linear transformations (Dubey et al., 2022; Farzad et al., 2019). Traditional LSTM networks typically employ the sigmoid function for gating mechanisms and the tanh function for output activation (Vijayaprabakaran & Sathiyamurthy, 2022). While both functions are non-linear, they yield different output ranges: the sigmoid function outputs values between 0 and 1, whereas the tanh function outputs values between -1 and 1. The mathematical formulations for the activation functions are described in Equations (4) and) and (5).

$$f(x) = (1 - e^{-x})^{-1} \quad (4)$$

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (5)$$

D. Model Evaluation

A method for evaluating multi-class classification involves choosing the optimal model based on a single criterion, such as overall accuracy or mean accuracy. Overall accuracy is a numerical metric that calculates the percentage of correctly classified data. However, the reliability of the data may be influenced by the distribution of classes. On the other hand, the average accuracy is not significantly affected by changes in the distribution of classes (Theissler et al., 2022). Due to the uneven distribution of the data utilized in this inquiry, the average accuracy will be calculated using Equation (6):

$$acc_{avg} = \frac{1}{|C|}\sum_{i=1}^{|C|} \frac{TP_{C_i}}{TP_{C_i} + FP_{C_i}} \quad (6)$$

TP_{C_i} denotes the count of accurately classified data points belonging to the class Ci, whereas FP_{C_i} indicates the count of misclassifications inside that class.

Ting (2016) suggests that a confusion matrix can be employed to display the evaluation outcomes of a multi-class classification model. The matrix provides a succinct summary of the classifier's performance on the test data. The confusion matrix comprises four fundamental components: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). This data can determine several performance metrics, such as precision and recall. The metrics are mathematically defined according to the specifications provided in the reference (Singh et al., 2021):

1. Precision: is a measure of the model's ability to forecast positive outcomes accurately, and it is calculated using Equation (7):

$$Precision = \frac{TP}{(TP+FP)} \quad (7)$$

2. Recall: pertains to the ability of the model to detect all instances of positive situations correctly. The value is calculated using Equation (8):

$$Recall = \frac{TP}{(TP+FN)} \quad (8)$$

V. RESULTS

A. Data Pre-processing

The dataset utilized in this study consists of 215,063 entries and encompasses six variables: drug name, ailment, review, rating, date, and useful count. The primary variables requiring processing are the "review" and "condition" variables, as the classification technique involves comparing the "condition" variable with the existing reviews. The initial phase of the analysis focuses on pre-processing the data to enhance the quality and reliability of the results. Pre-processing techniques include eliminating redundant variables, detecting and addressing missing values, converting all text to lowercase (case folding), removing stopwords, and applying

DOI: 10.5281/zenodo.14890438

lemmatization. These techniques ensure that the text data is clean and uniform, as illustrated in Table 3, which displays samples of the pre-processed results for the "review" variable.

Following pre-processing, an analysis is conducted on the data to identify its characteristics, patterns, and qualities, which is essential for evaluating the effectiveness of feature engineering in enhancing the model's performance. Before proceeding, the labels to be classified undergo an initial selection process. This label selection involves the careful identification of specific labels relevant to the classification task. To ensure a sufficient sample size, only labels containing at least 2,000 reviews and directly addressing the health conditions reported by drug users are selected. This process narrows down the original 884 labels to 13, which include conditions such as depression, pain, anxiety, acne, bipolar disorder, insomnia, weight loss, obesity, ADHD, type 2 diabetes, high blood pressure, vaginal yeast infection, and abnormal uterine bleeding. The outcomes of the label selection process are graphically represented in a histogram in Figure 2 (a), which reveals a significant disparity in label frequency, particularly with the label 'depression' having a slightly higher occurrence than others. This imbalance underscores the need for employing data balancing techniques, such as resampling, to improve the study's representativeness and fairness.

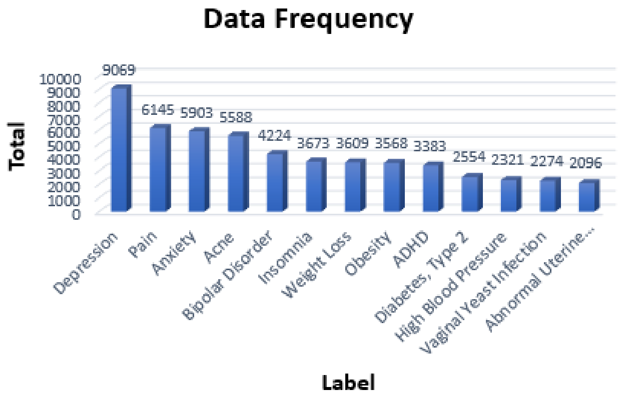
TABLE 3

Sample Data from the Pre-processing Results

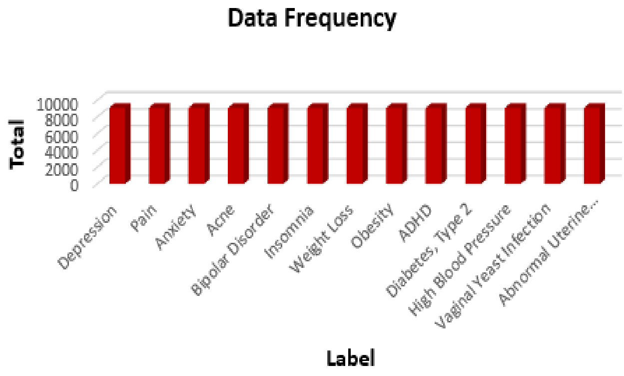
Pra	Post
It has no side effects, I take it in combination	it has no side effect take it in combination …
My son is halfway through his fourth week…	my son is halfway through his fourth week…
I used to take another oral contraceptive which…	used to take another oral contraceptive which…
Iand#039;ve been on thyroid medication 49 years...	ve been on every medicine under the sun it…
Iand#039;ve been on saxenda for 3 months now and I…	ve been on saxenda for months now…

The resampling stage addresses this data inconsistency using the ROS method. The primary aim of ROS is to balance the distribution of data across various classes while considering the proportions of the majority class. By augmenting the quantity of data from the minority class, ROS creates a more equitable distribution of labels. This technique generates synthetic data to enhance the underrepresented class, thereby equalizing the dataset between minority and majority classes. The resulting data distribution, as shown in Figure 2 (b), demonstrates a fair and balanced frequency of occurrences for each label, setting the stage for the next phase of analysis.

Once the dataset is balanced, it is partitioned into two segments: training and validation datasets. The training dataset is employed to construct and train the model, enabling it to identify patterns and perform classifications. Meanwhile, the validation dataset assesses the model's performance post-training, ensuring its accuracy when applied to previously unseen data. The dataset is divided into two portions, with 80% allocated for training (totalling 94,317 instances) and 20% reserved for validation (comprising 23,850 instances). This structured approach to data processing, including pre-processing, label selection, resampling, and data splitting, ensures the development of a robust classification model that effectively addresses the nuances of health condition reviews.



(a)



(b)

Figure 2. Label frequency for (a) Before and (b) After resampling.

B. Model Development

In developing the neural network model, particularly the LSTM model, various strategies can be employed to enhance its performance, notably through the inclusion of an embedding layer positioned at the beginning of the network prior to the LSTM layer. The embedding layer serves a crucial role in enriching the model's comprehension and representation of the text's semantic meaning, as determined by the selected word embedding technique. By implementing this layer, textual data is transformed into numerical representations, facilitating more effective processing and analysis by the model. This improvement significantly elevates the model's categorization capabilities, particularly in the context of medical reviews.

This study investigates three-word embedding methods: Word2Vec, GloVe, and FastText, resulting in the LSTM model being executed in three distinct configurations corresponding to each word embedding technique. A systematic optimization process was conducted to identify the best values for the parameters "lstm_unit," "batch size," and "epoch" to optimize the LSTM model's performance. This optimization employed two techniques: GridSearchCV, which focused on optimizing the "lstm_unit" and "batch size", and EarlyStopping, which monitored the "epoch" parameter to terminate training if no improvement in validation accuracy was observed over ten epochs. The optimal combinations derived from this hyperparameter tuning process are outlined in Table 4.

DOI: 10.5281/zenodo.14890438

TABLE 4

Parameters Optimal Combination

Model	LSTM Unit	Batch Size	Epoch
LSTM + Word2Vec	16	128	53
LSTM + GloVe	16	128	52
LSTM + FastText	32	128	72

The best-performing parameters will be utilized to fit the model, which involves training the LSTM on pre-existing training data. This training process aims to fine-tune the model by adjusting its parameters for better alignment with the dataset, thereby enhancing its predictive accuracy. The results of this training, validation, and testing are compared in Table 5, showcasing the performance metrics achieved by each LSTM model.

Table 5 clearly illustrates that the FastText model outperforms both Word2Vec and GloVe in terms of accuracy. With training accuracy at 95.07%, validation accuracy at 95.44%, and test accuracy at 86.22%, FastText demonstrates a superior capability in processing medical reviews compared to the other embedding techniques. However, it is essential to note that while FastText exhibits the highest performance in this evaluation, further investigation is necessary to definitively establish it as the optimal choice for word embedding, highlighting the need for continued assessment of its effectiveness in diverse contexts.

TABLE 5

Comparison of Accuracy Based on Embeddings

Model	Train Acc (%)	Validation Acc (%)	Test Acc (%)
LSTM + Word2Vec	95.09	94.47	85.20
LSTM + GloVe	94.88	94.17	84.19
LSTM + FastText	95.07	95.44	86.22

C. Model Evaluation

The evaluation of model performance in this work is conducted by quantifying the average accuracy. This study examines explicitly multi-class classification and evaluates the model's efficacy using two average accuracy techniques: Macro Average (MA) and Weighted Average (WA). These methods provide a thorough assessment of the model's performance.

The MA is computed by calculating the average accuracy for each class individually without considering the size of each class. On the other hand, WA determines the mean accuracy by assigning weights based on the number of samples in each class. Integrating these two approaches accomplishes a more comprehensive evaluation of the efficacy of the multi-class classification model. The average accuracy attained is presented in Table 6.

TABLE 6

Average Accuracy of Classification Models

Model	MA (%)	WA (%)
LSTM + Word2Vec	86	85
LSTM + GloVe	85	54
LSTM + FastText	87	86

The data in Table 6 shows that the LSTM model, using FastText word embedding, achieves a superior degree of performance, with an accuracy rate of 87%. These outperform LSTM models with Word2Vec (86%) and GloVe (85%) in terms of performance. FastText has demonstrated superior efficacy in the processing and comprehension of textual context when compared to alternative embedding techniques. In addition, the LSTM+FastText model's classification results are presented as a confusion matrix, depicted in Figure 3. Figure 3 provides the necessary data for manually calculating assessment measures such as precision, recall, and f1-score using Equations (7) and (8). The evaluation metrics for the models show that the LSTM combined with FastText achieved a precision and recall of 0.87 each.

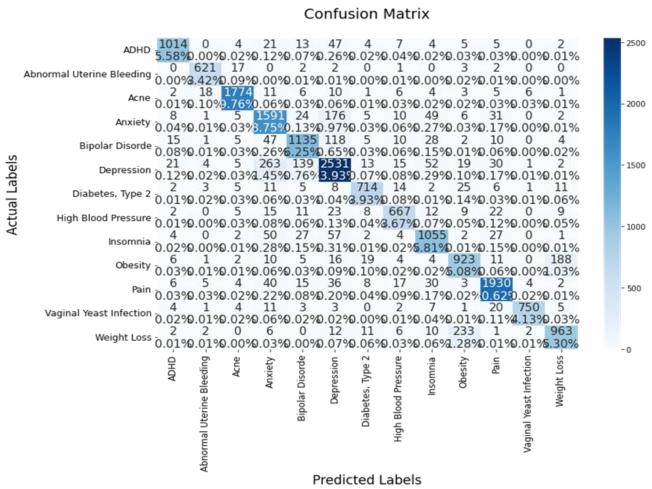


Figure 3. Confusion matrix model LSTM+FastText.

Mathematically, an average precision score of 0.87 signifies that 87% of the positive predictions for each class are accurate. For example, if the model predicts that 100 occurrences belong to the ADHD class, about 87 of these examples are accurately identified as ADHD (True Positives). Out of the total number of incidents, seven are classified as False Positives, indicating that they are wrongly labelled as ADHD even though they do not truly belong to that category. This average precision indicates that approximately 87% of the model's positive predictions are generally accurate. This outcome emphasizes that the model consistently delivers precise forecasts for most categories. However, there might be minor discrepancies among the categories. Such variability may occur due to factors such as an uneven distribution of classes, the intricacy of each class, or the quality of the features utilized in categorization.

A recall score of 0.87 indicates that the model has a high level of accuracy in correctly classifying each category. For example, let us examine the ADHD course. The model exhibits a recall rate of 0.87 for this class, demonstrating its ability to correctly recognize around 87% of all instances that genuinely fall into the ADHD group. Based on empirical evidence, when tested on a group of 100 individuals with ADHD, the model shows a capacity to accurately identify around 87 of them as having ADHD.

The classification results in Figure 3 illustrate the model's adeptness in accurately identifying and categorizing most labels. The results indicate that the model has a high level of proficiency in understanding and analyzing data, making it reliable for similar classification tasks, with low error rates on the examined data.

DOI: 10.5281/zenodo.14890438

D. Comparative Analysis with Existing Research

The proposed method demonstrates that FastText attains a training accuracy of 95.07%, a validation accuracy of 95.44%, and a testing accuracy of 86.22%. These outcomes underscore FastText's superior capacity to capture the semantic meaning of words compared to alternative techniques such as Word2Vec and GloVe. FastText utilizes an n-gram character approach, which enhances its ability to comprehend the contextual nuances of words, even those that are infrequently encountered within the dataset.

In comparison to earlier studies, such as Mahboob and Ali (Mahboob & Ali, 2018), which reported an average accuracy of 48%, and Mishra (Mishra, 2020), who achieved an accuracy of 88.89% using the LGBM model, the outcomes produced by the proposed method demonstrate a notably competitive and robust performance in sentiment classification. Furthermore, while Rani and Reddy (Rani & Reddy, 2022) achieved the highest reported accuracy of 93% employing Linear SVC with TF-IDF, and Suhartono et al. (Suhartono et al., 2022) attained an accuracy of 84.87% using CNN with GloVe, the integration of FastText within the LSTM model yields superior results in both training and validation, reaching an accuracy of 95.44%.

The effectiveness of FastText in achieving high accuracy is further exemplified by the findings of Tukino et al. (Tukino et al., 2024), who developed an LSTM model that attained an accuracy of 71%. In contrast, the proposed method demonstrates a test accuracy of 86.22%, indicating that FastText not only excels in training and validation stages but also proves reliable when utilized on test datasets, thus highlighting its applicability in real-world scenarios. Overall, the findings of this study suggest that employing FastText within an LSTM framework represents a superior strategy for sentiment classification in health reviews, establishing a robust foundation for future research in this domain.

Furthermore, unlike the approaches in prior studies, the proposed method in this research utilizes a dataset from the UCI Repository containing 13 disease condition labels, which enables a more detailed analysis of reviews based on specific categories. This approach not only increases the complexity and depth of sentiment analysis but also provides more informative insights into patient needs. The use of multiple labels contributes to greater analytical accuracy, as demonstrated by the performance of FastText within the LSTM model, and this method has the potential to significantly influence targeted product strategies and healthcare services based on user health conditions. A summary of the research findings and their comparison with previous studies is provided in Table 7.

TABLE 7

Comparative Analysis of Research Outcomes with Previous Studies

Researchers	Research Outcomes
Mahboob and Ali (2018)	The categories of eye care, skincare, and sexual health medications achieved the highest accuracy in customer sentiment analysis, reaching 100%. This was followed by allergy and sinus medications at 67%, while the remaining six medication categories reached an accuracy of 33%. Consequently, the overall average accuracy was calculated at 48%.
Mishra (2020)	The LGBM model stands out as the most effective boosting algorithm for sentiment analysis of drug reviews, achieving an accuracy of 88.89% and an F1 Score of 92.20%.
Rani and Reddy (2022)	The Linear SVC model combined with TF-IDF achieved the highest accuracy at 93%, whereas the Decision Tree model using Word2Vec attained only 78%.
Suhartono et al. (2022)	The CNN model utilizing GloVe achieved the highest test accuracy of 84.87% in the sentiment analysis of drug reviews.
Tukino et al. (2024)	This study successfully developed a public opinion classification model regarding Puskesmas utilizing the LSTM-Adamax architecture, achieving an accuracy of 71% and a peak precision of 76%, in contrast to the LSTM+Adadelta model, which attained only 57%.
Proposed Method	The word embedding technique that demonstrates the highest performance in the LSTM model for sentiment classification of drug reviews is FastText, achieving a training accuracy of 95.07%, a validation accuracy of 95.44%, and a test accuracy of 86.22%.

VI. Discussion

This study investigates the performance of LSTM models using Word2Vec, GloVe, and FastText word embedding techniques for classifying health condition reviews. The results indicate that FastText outperforms the other two methods, achieving a test accuracy of 86.22% compared to Word2Vec (85.20%) and GloVe (84.19%). FastText's superior performance is attributed to its n-gram-based character embedding, which enhances the model's ability to capture semantic meanings, even for rare or out-of-vocabulary words. This feature is especially beneficial in processing health reviews, which often contain technical or less frequently used terms.

The evaluation metrics further support FastText's effectiveness. With precision and recall scores of 0.87, the LSTM model utilizing FastText demonstrates a robust ability to classify health conditions with minimal false positives and negatives correctly. The MA and WA accuracy scores also highlight FastText's capability to perform well across both balanced and imbalanced datasets. These findings align with previous research, such as Rani and Reddy, who achieved 93% accuracy with Linear SVC and TF-IDF but exceeded the performance of studies like Suhartono et al., which reported 84.87% accuracy using CNN with GloVe.

A notable contribution of this study is the dataset's scope, which includes 13 distinct health condition labels derived from patient reviews. By employing a rigorous pre-processing pipeline, label balancing through Random Over-Sampling, and systematic hyperparameter optimization, the proposed approach ensures high reliability and applicability in sentiment classification tasks. The use of multi-label classification adds depth to the analysis, providing more actionable insights into user feedback across specific health conditions.

Compared to prior studies, this research demonstrates a more detailed and robust methodology. For example, while Mishra reported 88.89% accuracy using the LGBM model, this study's

DOI: 10.5281/zenodo.14890438

integration of FastText within an LSTM framework achieves comparable, if not superior, performance with enhanced interpretability and scalability. Additionally, the comparison with Mahboob and Ali highlights the broader applicability of the proposed method, as their study averaged only 48% accuracy across categories.

In summary, the findings emphasize FastText's potential for improving sentiment classification in health reviews, laying the groundwork for future research to explore its application in other domains. Further studies may examine the integration of hybrid models or ensemble techniques to optimize performance further. Moreover, expanding the dataset to include multilingual reviews could enhance the generalizability of the proposed method across diverse healthcare contexts.

VII. CONCLUSIONS

This study evaluates the effectiveness of LSTM models by comparing their performance using word embedding techniques, such as Word2Vec, GloVe, and FastText. The analysis reveals that the FastText embedding technique attains the best test accuracy rate of 86.22%, followed by Word2Vec, with an accuracy rate of 85.20%, and GloVe, with the lowest accuracy rate of 84.19%. These findings suggest that FastText significantly enhances the performance of LSTM models in generalizing new data. FastText integrates word morphology with local context, leading to enhanced word representations and increased efficacy of the model in text classification. These findings highlight the importance of selecting a suitable embedding methodology to get the best results in text classification issues.

Looking ahead, future work could explore the hybrid approaches that combine multiple embedding techniques, which may yield further improvements in model performance and robustness. Further research could also assess the integration of FastText with other advanced DL architectures, such as transformer models, to enhance contextual understanding and improve predictive accuracy in various applications.

Funding: This research received no specific grant from any funding agency.

CONFLICTS OF INTEREST: THE AUTHORS DECLARE NO CONFLICT OF INTEREST.

REFERENCES

1. Adipradana, R., Nayoga, B. P., Suryadi, R., & Suhartono, D. (2021). Hoax analyzer for indonesian news using rnns with fasttext and glove embeddings. *Bulletin of Electrical Engineering and Informatics*, *10*(4), 2130–2136. https://doi.org/10.11591/eei.v10i4.2956
2. Ahmed, A. N., Yafouz, A., Birima, A. H., Kisi, O., Huang, Y. F., Sherif, M., Sefelnasr, A., & El-Shafie, A. (2022). Water level prediction using various machine learning algorithms: a case study of Durian Tunggal river, Malaysia. *Engineering Applications of Computational Fluid Mechanics*, *16*(1), 422–440. https://doi.org/10.1080/19942060.2021.2019128
3. Anjani, M., & Nurramdhani, H. (2024). Comparison Performance of Word2Vec, Glove, Fasttext Using Support Vector Machine Method for Sentiment Analysis. *Jurnal Teknik Informatika (JUTIF)*, *5*(3), 669–674. https://doi.org/10.52436/1.jutif.2024.5.3.1366
4. Antanasijević, D., Pocajt, V., Perić-Grujić, A., & Ristić, M. (2020). Multilevel split of high-dimensional water quality data using artificial neural networks for the prediction of dissolved oxygen in the Danube River. *Neural Computing and Applications*, *32*(8), 3957–3966. https://doi.org/10.1007/s00521-019-04079-y
5. Asudani, D. S., Nagwani, N. K., & Singh, P. (2023). Impact of word embedding models on text analytics in deep learning environment: a review. *Artificial Intelligence Review*, *56*(9), 10345–10425. https://doi.org/10.1007/s10462-023-10419-1
6. Balci, M., Dokur, E., Yuzgec, U., & Erdogan, N. (2024). Multiple decomposition-aided long short-term memory network for enhanced short-term wind power forecasting. *IET Renewable Power Generation*, *18*(3), 331–347. https://doi.org/10.1049/rpg2.12919
7. Bartz-Beielstein, T., Zaefferer, M., & Mersmann, O. (2023). Tuning: Methodology. In *Hyperparameter Tuning for Machine and Deep Learning with R: a Practical Guide* (pp. 7–26). https://doi.org/10.1007/978-981-19-5170-2
8. Bashir, M. F., Arshad, H., Javed, A. R., Kryvinska, N., & Band, S. S. (2021). Subjective Answers Evaluation Using Machine Learning and Natural Language Processing. *IEEE Access*, *9*, 158972–158983. https://doi.org/10.1109/ACCESS.2021.3130902
9. Bhavani, A., & Santhosh Kumar, B. (2021). A Review of State Art of Text Classification Algorithms. *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*, 1484–1490. https://doi.org/10.1109/ICCMC51019.2021.9418262
10. Cao, W., Shi, Y. Z., Qiu, H., & Zhang, B. (2021). SS-RNN: A Strengthened Skip Algorithm for Data Classification Based on Recurrent Neural Networks. *Frontiers in Genetics*, *12*. https://doi.org/10.3389/fgene.2021.746181
11. Chiam, D. H., Lim, K. H., & Law, K. H. (2023). LSTM power quality disturbance classification with wavelets and attention mechanism. *Electrical Engineering*, *105*(1), 259–266. https://doi.org/10.1007/s00202-022-01667-5
12. Choi, J., & Lee, S. W. (2020). Improving FastText with inverse document frequency of subwords. *Pattern Recognition Letters*, *133*, 165–172. https://doi.org/10.1016/j.patrec.2020.03.003
13. Dogra, V., Verma, S., Kavita, Chatterjee, P., Shafi, J., Choi, J., & Ijaz, M. F. (2022). A Complete Process of Text Classification System Using State-of-the-Art NLP Models. In *Computational Intelligence and Neuroscience* (Vol. 2022). https://doi.org/10.1155/2022/1883698
14. Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. In *Neurocomputing* (Vol. 503, pp. 92–108). https://doi.org/10.1016/j.neucom.2022.06.111
15. Farzad, A., Mashayekhi, H., & Hassanpour, H. (2019). A comparative performance analysis of different activation functions in LSTM networks for classification. *Neural Computing and Applications*, *31*(7), 2507–2521. https://doi.org/10.1007/s00521-017-3210-6
16. Gao, R., Tang, Y., Xu, K., Huo, Y., Bao, S., Antic, S. L.,

Epstein, E. S., Deppen, S., Paulson, A. B., Sandler, K. L., Massion, P. P., & Landman, B. A. (2020). Time-distanced gates in long short-term memory networks. *Medical Image Analysis*, *65*. https://doi.org/10.1016/j.media.2020.101785
17. Grbčić, L., Družeta, S., Mauša, G., Lipić, T., Lušić, D. V., Alvir, M., Lučin, I., Sikirica, A., Davidović, D., Travaš, V., Kalafatovic, D., Pikelj, K., Fajković, H., Holjević, T., & Kranjčević, L. (2022). Coastal water quality prediction based on machine learning with feature interpretation and spatio-temporal analysis. *Environmental Modelling and Software*, *155*. https://doi.org/10.1016/j.envsoft.2022.105458
18. HaCohen-Kerner, Y., Miller, D., & Yigal, Y. (2020). The influence of preprocessing on text classification using a bag-of-words representation. *PLoS ONE*, *15*(5). https://doi.org/10.1371/journal.pone.0232525
19. Hayaty, M., Muthmainah, S., & Ghufran, S. M. (2021). Random and Synthetic Over-Sampling Approach to Resolve Data Imbalance in Classification. *International Journal of Artificial Intelligence Research*, *4*(2), 86. https://doi.org/10.29099/ijair.v4i2.152
20. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
21. Hossain, R., & Timmer, D. D. (2021). Machine learning model optimization with hyper parameter tuning approach. *Global Journal of Computer Science and Technology*, *21*(2).
22. Imaduddin, H., Widyawan, & Fauziati, S. (2019). Word embedding comparison for Indonesian language sentiment analysis. *Proceeding - 2019 International Conference of Artificial Intelligence and Information Technology, ICAIIT 2019*, 426–430. https://doi.org/10.1109/ICAIIT.2019.8834536
23. Islam Khan, M. S., Islam, N., Uddin, J., Islam, S., & Nasir, M. K. (2022). Water quality prediction and classification based on principal component regression and gradient boosting classifier approach. *Journal of King Saud University - Computer and Information Sciences*, *34*(8), 4773–4781. https://doi.org/10.1016/j.jksuci.2021.06.003
24. Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2017). Bag of tricks for efficient text classification. *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference*, *2*. https://doi.org/10.18653/v1/e17-2068
25. Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, *3*(1), 91–99. https://doi.org/10.1016/j.gltp.2022.04.020
26. Mahboob, K., & Ali, F. (2018). Sentiment Analysis of Pharmaceutical Products Evaluation Based on Customer Review Mining. *Journal of Computer Science & Systems Biology*, *11*(3). https://doi.org/10.4172/jcsb.1000271
27. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In: Conference on Advances in Neural Information Processing Systems. *Distributed Representations of Words and Phrases and Their Compositionality*.
28. Mishra, S. (2020). Drug Review Sentiment Analysis and EDA. *International Journal of Trend in Scientific Research and Development*, *Volume 5*(Issue 4, May-June 2021).
29. Mohamed, E. H., Moussa, M. E. S., & Haggag, M. H. (2020). An Enhanced Sentiment Analysis Framework Based on Pre-Trained Word Embedding. *International Journal of Computational Intelligence and Applications*, *19*(4). https://doi.org/10.1142/S1469026820500315
30. Nasution, M. A. S., & Setiawan, E. B. (2023). Enhancing Cyberbullying Detection on Indonesian Twitter: Leveraging FastText for Feature Expansion and Hybrid Approach Applying CNN and BiLSTM. *Revue d'Intelligence Artificielle*, *37*(4), 929–936. https://doi.org/10.18280/ria.370413
31. Nath, B., Sarkar, S., & C. Debnath, N. (2024). A Study of Word Embedding Models for Machine Translation of North Eastern Languages. *Communications in Computer and Information Science*, *1956 CCIS*, 343–359. https://doi.org/10.1007/978-3-031-48879-5_26
32. Nawangsari, R. P., Kusumaningrum, R., & Wibowo, A. (2019). Word2vec for Indonesian sentiment analysis towards hotel reviews: An evaluation study. *Procedia Computer Science*, *157*, 360–366. https://doi.org/10.1016/j.procs.2019.08.178
33. Nurdin, A., Anggo Seno Aji, B., Bustamin, A., & Abidin, Z. (2020). Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks. *Jurnal Tekno Kompak*, *14*(2), 74. https://doi.org/10.33365/jtk.v14i2.732
34. Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1532–1543. https://doi.org/10.3115/v1/d14-1162
35. Rahman, A. (2019). Statistics-based data preprocessing methods and machine learning algorithms for big data analysis. *International Journal of Artificial Intelligence*, *17*(2), 44–65.
36. Rani, M. R. J., & Reddy, K. P. K. R. (2022). Drug recommendation system based on analysis of drug reviews using machine learning. *International Journal of Computer Science Trends and Technology*, *10*(5), 94–100. https://doi.org/10.1063/5.0168205
37. Rao, M. S., Reddy, B. E., Ramana, K., Prasanna, K., & Singh, S. (2021). Texture classification using Minkowski distance measure-based clustering for feature selection. *Journal of Electronic Imaging*, *31*(04). https://doi.org/10.1117/1.jei.31.4.041204
38. Rezaie-Balf, M., Attar, N. F., Mohammadzadeh, A., Murti, M. A., Ahmed, A. N., Fai, C. M., Nabipour, N., Alaghmand, S., & El-Shafie, A. (2020). Physicochemical parameters data assimilation for efficient improvement of water quality index prediction: Comparative assessment of a noise suppression hybridization approach. *Journal of Cleaner Production*, *271*. https://doi.org/10.1016/j.jclepro.2020.122576
39. Sajib, A. M., Diganta, M. T. M., Rahman, A., Dabrowski, T., Olbert, A. I., & Uddin, M. G. (2023). Developing a novel tool for assessing the groundwater incorporating water quality index and machine learning

DOI: 10.5281/zenodo.14890438

approach. *Groundwater for Sustainable Development*, *23*. https://doi.org/10.1016/j.gsd.2023.101049

40. Sasada, T., Liu, Z., Baba, T., Hatano, K., & Kimura, Y. (2020). A resampling method for imbalanced datasets considering noise and overlap. *Procedia Computer Science*, *176*, 420–429. https://doi.org/10.1016/j.procs.2020.08.043
41. Shobana, J., & Murali, M. (2021). An efficient sentiment analysis methodology based on long short-term memory networks. *Complex and Intelligent Systems*, *7*(5), 2485–2501. https://doi.org/10.1007/s40747-021-00436-4
42. Singh, P., Singh, N., Singh, K. K., & Singh, A. (2021). Diagnosing of disease using machine learning. In *Machine Learning and the Internet of Medical Things in Healthcare* (pp. 89–111). https://doi.org/10.1016/B978-0-12-821229-5.00003-3
43. Sitender, Sangeeta, Sushma, N. S., & Sharma, S. K. (2023). Effect of GloVe, Word2Vec and FastText Embedding on English and Hindi Neural Machine Translation Systems. *Lecture Notes in Networks and Systems*, *572*, 433–447. https://doi.org/10.1007/978-981-19-7615-5_37
44. Sivakumar, S., Videla, L. S., Rajesh Kumar, T., Nagaraj, J., Itnal, S., & Haritha, D. (2020). Review on Word2Vec Word Embedding Neural Net. *Proceedings - International Conference on Smart Electronics and Communication, ICOSEC 2020*, 282–290. https://doi.org/10.1109/ICOSEC49089.2020.9215319
45. Suhartono, D., Purwandari, K., Jeremy, N. H., Philip, S., Arisaputra, P., & Parmonangan, I. H. (2022). Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews. *Procedia Computer Science*, *216*, 664–671. https://doi.org/10.1016/j.procs.2022.12.182
46. Theissler, A., Thomas, M., Burch, M., & Gerschner, F. (2022). ConfusionVis: Comparative evaluation and selection of multi-class classifiers based on confusion matrices. *Knowledge-Based Systems*, *247*. https://doi.org/10.1016/j.knosys.2022.108651
47. Ting, K. M. (2016). Confusion Matrix. In: Sammut, C., Webb, G.I. (eds) Encyclopedia of Machine Learning. In *Encyclopedia of Machine Learning and Data Science*.
48. Tukino, Hananto, A., Nanda, R. A., Novalia, E., Sediyono, E., & Sanjaya, J. (2024). LSTM and Word Embedding: Classification and Prediction of Puskesmas Reviews Via Twitter. *E3S Web of Conferences*, *500*. https://doi.org/10.1051/e3sconf/202450001018
49. Uddin, M. G., Nash, S., Rahman, A., Dabrowski, T., & Olbert, A. I. (2024). Data-driven modelling for assessing trophic status in marine ecosystems using machine learning approaches. *Environmental Research*, *242*. https://doi.org/10.1016/j.envres.2023.117755
50. Vijayaprabakaran, K., & Sathiyamurthy, K. (2022). Towards activation function search for long short-term model network: A differential evolution based approach. *Journal of King Saud University - Computer and Information Sciences*, *34*(6), 2637–2650. https://doi.org/10.1016/j.jksuci.2020.04.015
51. Wang, H., & Li, F. (2022). A text classification method based on LSTM and graph attention network. *Connection Science*, *34*(1), 2466–2480. https://doi.org/10.1080/09540091.2022.2128047
52. Zhu, H., & Lei, L. (2022). The Research Trends of Text Classification Studies (2000–2020): A Bibliometric Analysis. *SAGE Open*, *12*(2). https://doi.org/10.1177/21582440221089963

DOI: 10.5281/zenodo.14890438