



Identifying Disease Outbreak Messages on Social Media Using Graph Neural Network Variants with TF-IDF and Word Embedding Features

Irwan Budiman¹, Mohammad Reza Faisal², Muliadi³, Muhammad Rafi⁴, Husnil Kamil⁵,
Luther Alexander Latumakulita⁶, Luu Duc Ngo⁷

^{1,2,3,4}Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia

⁵Information System, Andalas University, Padang, Indonesia

⁶Information System, Sam Ratulangi University, Manado, Indonesia

⁷Computer Science, Bac Lieu University, Bac Lieu City, Vietnam

*irwan.budiman@ulm.ac.id

Abstract

The swift dissemination of information regarding disease outbreaks via social media presents significant challenges in the automatic identification of pertinent outbreak-related messages, particularly within multilingual textual data. The objective of this research is to devise and assess classification models grounded in Graph Neural Networks for the detection of disease outbreak messages within Indonesian and English datasets. The methodology employed encompasses text normalisation, feature extraction utilizing TF-IDF and word embedding techniques, including Word2Vec, fastText, and GloVe, followed by classification employing Graph Convolutional Networks (GCN) and Graph Attention Networks (GAT). The findings of this study indicate that the optimal performance on the Indonesian dataset was attained by GCN utilizing TF-IDF features amounting to 5000, resulting in an accuracy of 93%, whereas the superior GAT model attained 92% accuracy utilizing 4000 TF-IDF features. For the English dataset, the most effective GCN model reached an accuracy of 86.77% employing fastText, while the leading GAT model achieved 86.79% with the same fastText feature set. These results surpass those of previous investigations that employed CNN, LSTM, and BERT-based models for analogous tasks of disease outbreak message classification. This study concludes that GCN and GAT represent efficacious deep learning methodologies for the classification of multilingual disease outbreak messages. GCN demonstrates superior performance on Indonesian text utilizing TF-IDF-based features, while GAT exhibits marginally better performance on English text employing fastText features. In summary, graph-based models are capable of capturing relational information among textual features and delivering competitive classification performance when juxtaposed with earlier deep learning methodologies.

Keywords: deep learning; disease outbreak surveillance; feature extraction; social media; text classification

How to Cite: I. Budiman, "Identifying Disease Outbreak Messages on Social Media Using Graph Neural Network Variants with TF-IDF and Word Embedding Features", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 10, no. 5, pp. 1464–1474, Oct. 2026, doi: [10.29207/resti.v10i5.7837](https://doi.org/10.29207/resti.v10i5.7837).
Permalink/DOI: <https://doi.org/10.29207/resti.v10i5.7837>

Received: May 12, 2026

Accepted: September 5, 2026

Available Online: October 16, 2026

This is an open-access article under the [CC BY 4.0 License](https://creativecommons.org/licenses/by/4.0/)

Published by [Ikatan Ahli Informatika Indonesia](https://www.iaii.or.id/)

1. Introduction

The advent of the internet has significantly influenced various dimensions of human existence. Among the technologies leveraging the capabilities of the internet are social media platforms. Individuals utilise social media as a means of communication, disseminating daily activities, and even sharing experiences or observations during natural disasters [1], [2] or while witnessing incidents of disease outbreaks, either personally or through the experiences of others [3], [4].

Historically, information regarding the dissemination of disease outbreaks has been primarily acquired through news disseminated via radio and television. News

sources from these media are typically composed of personnel affiliated with the respective news organisations, thus imposing limitations regarding the geographic scope and timeliness of their news updates. With the emergence of the internet, the breadth and speed of information dissemination increased; nevertheless, it remained confined to the narratives provided solely by the news agency staff. Nonetheless, this information can still serve as a foundation for the establishment of surveillance systems during the occurrence of disease outbreaks [5], [6]. Social media has facilitated a more rapid and extensive dissemination of information concerning disease outbreaks compared to conventional online news, as it is derived from a

diverse array of social media users across various geographical locations. This characteristic enables social media to function as a more effective surveillance mechanism for disease outbreaks in comparison to online news platforms [4], [7], [8]. Given the volume of social media communications, artificial intelligence can filter disease outbreak messages in real time, enabling fast searches [8], [9].

Research into the advancement of disease outbreak surveillance grounded in social media exhibits parallels to the broader field of text classification research. Text classification investigations predominantly concentrate on enhancing the efficacy of classification models through the refinement of feature extraction methodologies and classification algorithms. Currently, widely employed feature extraction techniques include the Bag of Words (BoW) and word embedding approaches [10], [11]. The BoW-based feature extraction methodology operates by representing documents as vectors predicated on word frequency, wherein the technique assigns greater weight to words that frequently appear within a single document but are infrequently encountered in others, thereby allowing more informative terms to achieve greater prominence. The advantages of this approach lie in its simplicity, ease of implementation, and considerable effectiveness for keyword-based text classification. However, this technique exhibits drawbacks, including the generation of excessively lengthy and sparse feature sets due to its dependence on vocabulary quantity, its inability to capture word order, and its limited capacity for grasping the semantic meaning of words [11].

Conversely, the word embedding feature extraction technique translates words or documents into shorter, dense vectors, accounting for the semantic similarity between words based on their contextual occurrences. The merits of this approach include the generation of more succinct features, the capacity to capture semantic relationships between terms, and a superior representation of meanings [3], [12]. However, it is accompanied by disadvantages such as the necessity for extensive data or a pre-trained model, rendering the results more challenging to interpret [11].

Both methodologies possess distinct advantages and disadvantages, and existing research has yet to conclusively demonstrate the absolute superiority of either technique. Empirical studies have indicated that the BoW approach has the potential to enhance the performance of classification models [11], [13-15]. Conversely, alternative research has illustrated that feature extraction techniques employing vectors derived from word embeddings, such as word2vec, FastText, and Glove, outperform the BoW methodology [16-19].

Text classification research conducted thus far have employed two principal categories of classification algorithms. The most widely used algorithm today is deep learning. The deep learning algorithms that have been applied in the realm of text classification

predominantly include Convolutional Neural Network (CNN) [20-22] and Long Short-Term Memory (LSTM) [17], [23], [24]. CNN algorithms possess the capability to extract salient features that are well-suited for application in text classification research inputs characterised by numerous features. In contrast, LSTM demonstrates proficiency in processing data that entails sequence information, such as the arrangement of words within a textual document. In a separate investigation, Bidirectional Encoder Representations from Transformers (BERT) was devised for the purpose of text classification, enhancing the comprehension of natural language contexts by virtue of its capacity to interpret context from dual directions simultaneously [24-26].

Additional deep learning algorithms pertinent to text classification, beyond those previously mentioned, encompass Graph Neural Network (GNN)-based algorithms such as Graph Convolutional Network (GCN) and Graph Attention Network (GAT) [27-29]. Unlike CNNs, which predominantly concentrate on local patterns, GCN and GAT have the capacity to flexibly capture interrelationships between words or documents through the representation of text in a graphical format. Furthermore, in contrast to LSTM, which depends on a linear sequence of text, GCN and GAT are capable of modelling inter-element relationships that are distant through edges [30-32].

The advancement of text classification models aimed at identifying disease outbreak messages has been executed utilizing CNN, LSTM, and BERT algorithms. Research focusing on the detection of self-reported messages regarding pandemic disease symptoms on social media, employing word-embedding based feature extraction alongside a 1D CNN classification algorithm, yielded a peak model accuracy of 88.5% [12]. Another investigation employed an LSTM algorithm with structured data input based on word embedding, successfully elevating the model's accuracy to 89% [3]. Conversely, the accuracy of models formulated utilizing the BERT algorithm achieved 82% accuracy [26]. The aforementioned three studies utilized an identical dataset in the Indonesian language. Research utilizing English datasets with classification models constructed via BERT and COVID-Twitter-BERT algorithms attained accuracies of 70% and 76%, respectively [33]. Additionally, other studies have employed alternative BERT variants, specifically BERT-Base-Uncased and DisT-Base-Uncased, achieving accuracies of 84% and 80%, respectively [34]. At present, no case study has been conducted on disease outbreak identification employing GCN and GAT algorithms, despite the acknowledgment that both algorithms possess several advantages which may enhance the performance of the disease outbreak message identification model beyond that of models developed using CNN, LSTM, or BERT algorithms.

This study endeavors to analyze and juxtapose models predicated upon machine learning algorithms, with a

particular emphasis on Graph Neural Network (GNN)-based algorithms, specifically the Graph Convolutional Network (GCN) and the Graph Attention Network (GAT). The key difference between this research and prior studies is the specific case and dataset: classifying social media messages about disease outbreaks using Bag of Words (BoW) feature extraction and word embedding. A key differentiating aspect of the current investigation is the employment of a variable number of features screened contingent upon the maximum frequency of word occurrences. Furthermore, this research distinguishes itself by applying three distinct word embedding methodologies—Word2Vec, FastText, and GloVe—and by proposing a novel approach to combine all three word embedding features.

The research inquiries are as follows: (1) What is the efficacy of the classification model when integrating any combination of GNN-based algorithms with structured data inputs derived from BoW and word embedding techniques? (2) In what manner does the variation in the maximum number of features within the BoW technique influence the efficacy of classification models? (3) What is the impact of the amalgamation of the three word embedding techniques on the performance of the classification model? The responses to these research inquiries were instrumental in identifying the optimal combination of feature extraction techniques and classification algorithms that

led to the most effective classification model for disease outbreak-related messages, as measured by performance metrics. The primary contribution and implications of this study are a thorough evaluation of the efficacy of GNN-based algorithms alongside BoW-based text representation and word embedding for classifying social media messages related to disease outbreaks.

Methodologically, this research enhances the existing literature by elucidating the impact of feature number variation in BoW on model performance, while simultaneously demonstrating the potential for enhanced performance through the integration of three word embedding techniques, yielding a more comprehensive feature set for analysis.

From a practical standpoint, the outcomes of this study may serve as a valuable reference for the development of more precise and efficient social media-based outbreak monitoring systems, as well as assisting researchers and practitioners in discerning the optimal amalgamation of feature extraction techniques and classification algorithms for tasks related to early detection and analysis of messages concerning disease outbreaks.

2. Methods

The flow of this research is illustrated in Figure 1.

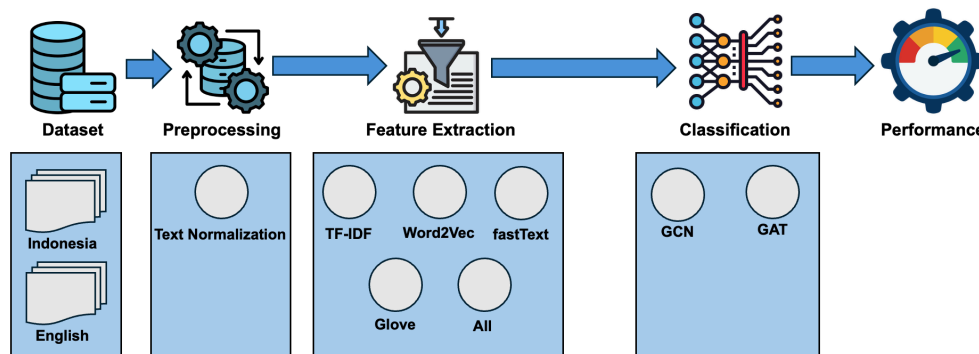


Figure 1. Research flow.

2.1. Dataset

In this research, a dataset comprising social media communications in both Indonesian and English was employed. Specifics and origins of the dataset are delineated in Table 1.

Table 1. Dataset summary by language, class label, and number of samples

Ref	Language	Class	# Sample	Notation
[12]	Indonesia	Positive	500	d_{id}
		Negative	500	
[34]	English	0	527	d_{en}
		1	681	

Additionally, the research utilized extensive textual data to develop the word embedding model represented in Table 2. Samples of textual data from Indonesian

datasets are presented in Table 3, whereas samples from English datasets are exhibited in Table 4.

Table 2. Wikipedia corpora used for training word embedding models.

Language	Name	GB	Articles	Notation
Indonesia	idwiki	2	700.000	$Wiki_{id}$
English	edwiki	50	7.000.000	$Wiki_{en}$

Table 3. Sample texts from the Indonesian disease outbreak dataset.

No	Text	Label
1	positif covid tuh ga enak banget	positive
...	...	positive
500	Enggak nii. Tiba2 ngerasain gejala aja terus kemarin test ehh positif	positive
501	hasil swab gue negatif juga	negative
...	...	negative
1000	Aku flu.. karna idungku kemasukan debu item.. sampe sekarang masih. Untungnya masih bisa nyium bau2an dan negatif	negative

Table 4. Sample texts from the English disease outbreak dataset.

No	Text	Label
1	dear serious question about vaccine in the army made me take flu vaccine spent two weeks in hospital sicker than a dog amp high fever never had to take it again amp have never gotten flu since makes me nervous amp hesitant abt taking coronavirus vaccine	0
...	...	0
527	the hospital where he was treated will use the death certificate when billing as cause of death	0
528	covid conspiracy i haven t got sick or felt sick since the start of covid i just found it weird i haven t caught a cold in over a year makes me think about a lot of things	1
...	...	1
1208	ok now im convinced the government is giving ppl covid cause i was fine days ago then i got tested now i feel like shit not to mention i came up positive something was in that swab no one is gonna tell me otherwise	1

2.2. Preprocessing

The text preprocessing phase of this investigation concerning X/Twitter data employs text normalisation methodologies that encompass text cleaning procedures aimed at eliminating extraneous characters, including punctuation, numerical digits, and special symbols, case folding to standardize all characters to a uniform format (lowercase), and tokenisation to decompose the text into more manageable units of analysis, such as words or phrases [35], [36].

These strategies are designed to mitigate data noise and enhance the quality of input for the analytical model, thereby ensuring that the resultant outcomes are more precise and effective [35].

2.3. Feature Extraction

Feature extraction pertaining to textual data constitutes the methodology of transforming unprocessed textual information into a quantitative representation meticulously organized to facilitate analysis by machine learning or deep learning algorithms [37]. This methodology holds considerable significance owing to the inherent unstructured nature of text; in the absence of feature extraction, classification algorithms are rendered ineffectual in discerning patterns or relationships among lexical items. This process offers improved model accuracy, reduced data dimensionality, and the filtering of relevant information while also reducing extraneous noise [37]. Its impact on classification algorithms is profound: an effective feature representation can expedite the training process, augment generalisation, and optimize the model's capacity for contextual comprehension [38]. Two prevalent methodologies for feature extraction include the Bag of Words (BoW) approach and word embeddings, exemplified by Word2Vec, FastText, and GloVe [39], [40].

In the context of BoW-based feature extraction employed in this study, the TF-IDF (Term Frequency-Inverse Document Frequency) approach is utilized. The TF-IDF methodology allocates weights to each lexical

item based on its frequency within the document and its infrequency across the broader corpus [37], [41]. The operational mechanism involves: (1) calculating the term frequency (TF) for each word within the document, (2) determining the inverse document frequency (IDF) to assess the rarity of the word's occurrence throughout the corpus, and (3) multiplying the TF by the IDF to derive the conclusive weight [41]. The resultant output manifests as a numerical vector whose length corresponds to the quantity of unique words present in the corpus, with each element representing the TF-IDF weight assigned to the respective word [42].

The process of feature extraction predicated upon word embedding is accomplished by developing an embedding model utilizing a corpus derived from the amalgamation of the research dataset and the WIKI corpus. This integration of the corpus is intended to enhance the representation of words, thereby enabling the model to identify a broader spectrum of vocabulary variations along with the contextual usage of words. The resultant word embedding model generates word vectors, each containing a dimensionality of 100 features for every individual word [39]. Once the embedding model is established, each document in the dataset is converted into structured data by mapping each word in the document to a numerical vector, using the embedding representation produced. Moreover, the word vectors within a singular document are amalgamated through the concatenation technique, ensuring that each document is depicted as a sequential arrangement of word vectors [39]. To ensure the structured length of all documents is uniform, we apply padding based on the maximum word count in the dataset. In instances where a document contains fewer words than the document with the maximum word count, the vacant sections will be supplemented with a value of 0 [25], [31]. Consequently, the entire document maintains a consistent representation size, rendering it suitable for input into deep learning classification algorithms.

2.4. Classification

In this research, two distinct algorithms predicated on Graph Neural Networks (GNNs) were employed, specifically the Graph Convolutional Network (GCN) and the Graph Attention Network (GAT).

Graph Neural Networks (GNN) have become valuable tools in text classification research, especially when text data can be represented as a graph. In this context, words or documents can be considered as nodes, while the relationships between words (such as co-occurrence or semantic relationships) can be viewed as edges. By leveraging GNNs' ability to capture complex relationships among graph elements, these models can enhance text classification accuracy by considering a broader context than single-word or phrase-based analysis. GNNs enable the modelling of interactions between words and documents, which is crucial for understanding meaning and context in text [43], [44].

Graph Convolutional Networks (GCN) have emerged as an effective approach in text classification by leveraging graph structures. In text classification research, GCNs can be used to aggregate information from related words within documents. By applying convolution operations to a graph representing relationships between words, GCNs can produce richer, more informative representations for each word or document. This enables the model to better comprehend the context and relationships among words, ultimately enhancing its performance in text classification tasks. Although GCNs have limitations in modelling dynamic graphs, they remain a popular choice in text classification applications that require deeper contextual understanding [29].

Graph Attention Networks (GAT) offer a more sophisticated approach to text classification by

introducing an attention mechanism that allows the model to assess the importance of neighbouring words in the classification context. In text classification research, GATs can assign different weights to related words, enabling the model to focus more on the most relevant words when determining text categories. GAT's advantages in capturing both local and global information make it highly effective for complex classification tasks, where word context is crucial. While GATs have higher computational complexity compared to GCNs, their flexibility and interpretability make them an attractive choice for text classification research, especially in situations where relationships among words are diverse and dynamic [43], [45].

GCN and GAT generally follow similar core computational steps, as outlined in Table 5 [29], [43].

Table 5. Core computation steps in GCN and GAT.

No	Step	GCN	GAT
1	Input Data	Node feature matrix and adjacency matrix A	Same as GCN: node features and adjacency matrix (or edge list)
2	Graph Normalisation	Adjacency matrix A is added with self-loops and normalized using node degrees: $\hat{A} = \tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2}$	Does not explicitly normalize the adjacency matrix
3	Feature Aggregation	Aggregates neighbor features using a linear operation based on the adjacency matrix: $H^{(l+1)} = \sigma(\hat{A} H^{(l)} W^{(l)})$	Aggregates using an attention-weighted sum between nodes: $h_i' = \sigma(\sum_{j \in N(i)} \alpha_{ij} W h_j)$
4	Inter-Node Weighting	Connection weights between nodes are determined by the graph structure (static)	Connection weights are computed dynamically using the attention mechanism: $\alpha_{ij} = \text{softmax}(\text{LeakyReLU}(a^T [W h_i W h_j]))$
5	Training & Optimisation	Trained like a standard neural network using cross-entropy or other loss functions	Same, but includes additional parameters from the attention vector.
6	Output Layer	Node representations from graph convolution, usually followed by a softmax	Node representations after the attention mechanism are also typically passed through a softmax

2.5. Performance

The confusion matrix serves as a pivotal evaluative instrument utilized to gauge the efficacy of classification models through a comparative analysis of actual labels against the predictive outcomes generated by the model, culminating in four primary components: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). In the context of binary classification [11], [13]. Conversely, the accuracy metric can be derived from the confusion matrix utilizing Equation 1.

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

2.6. Experiment Setup

This feature extraction methodology uses Python with Scikit-Learn, Gensim, Stanford NLP Group, and FastText libraries. The TF-IDF technique is applied according to the specifications in Table 6.

Table 6. TF-IDF feature extraction parameter settings.

Parameter	Value
min_df	1
max_df	0.95
ngram_range	(1,2)
stop_word	none
max_features	1000, 2000, 3000, 4000, 5000

The maximum number of TF-IDF features was varied from 1000 to 5000 to evaluate the influence of vocabulary size on classification performance. The upper limit of 5000 was chosen as a practical trade-off between retaining informative terms and controlling feature dimensionality. A larger number of TF-IDF features may capture more vocabulary information. However, it can also produce sparse and high-dimensional representations, increase computational cost, and introduce noisy or redundant terms. These conditions may negatively affect the stability and generalization of GCN and GAT models, particularly when the number of training samples is limited. Therefore, 5000 features were considered sufficient to represent the main textual patterns while keeping the structured input computationally feasible for graph-based classification.

Conversely, the parameters pertaining to word-based embedding methodologies are articulated in Table 7. This study also investigates the integration of feature extraction results from the three word embedding techniques. The process of combining the three structured data representations is illustrated in Equation 2.

$$d_{all}^{we} = \{d_{word2vec}^{we}, d_{fastText}^{we}, d_{Glove}^{we}\} \quad (2)$$

Table 7. Word embedding model parameter settings.

Model	Parameter	Value
Word2Vec	vector_size	100
	window	10
	min_count	5
	workers	8
FastText	vector_size	100
	window	10
	min_count	5
	workers	8
Glove	sg	1
	-vector-size	100
	-window-size	10
	-verbose	2
	-memory	4.0
	-threads	8
	-x-max	10
	-iter	10
	-binary	0

Text that has been transformed into feature vectors, such as TF-IDF and word embedding representations, is not readily amenable to direct processing in a graph format. To facilitate the use of GCN and GAT, it is imperative that each text datum be instantiated as a node, with subsequent inter-node relationships represented as edges. The methodology employed is a hybrid of K-Nearest Neighbors (KNN) and Cosine Similarity Thresholding. The initial phase involves identifying candidate nodes via KNN, in which each document is associated with the K most similar documents. The value of K designated for TF-IDF-based structured data is K=5, while for word embedding-based structured data, it is K=8. Subsequently, the Cosine Similarity Threshold is applied to select nodes. The threshold values implemented for word embedding and TF-IDF structured data are 0.3 and 0.1, respectively. Moreover, this information can be utilized to construct an adjacency matrix. This adjacency matrix subsequently serves as the input for graph-based classification algorithms.

The details regarding the configuration of classification algorithms are encapsulated in Table 8.

Table 8. Hyperparameter settings for GCN and GAT classifiers.

Method	Parameter	Value
GCN	Hidden dim	64
	Number class	2
	Dropout rate	0.5
	Activation	ReLU
	Output Activation	Log Softmax
	Epoch	200
	Learning Rate	0.01
	Optimizer	Adam
	Random State	42
GAT	Hidden dim	128
	num_classes	2
	dropout	0.5
	heads	8
	Layer 1 Activation	ELU (Exponential Linear Unit)
		Log Softmax,
	Layer 2 Activation	dim=1
	Epoch	200
	Learning Rate	0.01
	Optimizer	Adam
	Random State	42

3. Results and Discussions

3.1 Results

Following the execution of the text normalisation procedure, feature extraction was subsequently conducted on both the Indonesian (d_{id}) and English text datasets (d_{en}). Upon completion of this procedure, sanitized textual data was acquired, accompanied by quantitative insights regarding the word count in each dataset, as illustrated in Table 9.

Table 9. Descriptive statistics of document length in Indonesian and English datasets after text preprocessing.

Dataset	Average	Median	Max	Min
d_{id}	17	15	49	2
d_{en}	30	30	78	5

The TF-IDF-based feature extraction methodology was implemented with variations in the maximum number of features, specifically at 1000, 2000, 3000, 4000, and 5000. This process yielded ten structured datasets. Conversely, the outcomes of the word-embedding based feature extraction methodology resulted in eight structured datasets. The comprehensive dataset comprising the results of the feature extraction is presented in Table 10.

Table 10. Structured datasets generated from feature extraction methods.

Feature Extraction	Dataset	
TF-IDF	d_{id}	d_{en}
	d_{id}^{tf-idf}	d_{en}^{tf-idf}
	$d_{id,1000}^{tf-idf}$	$d_{en,1000}^{tf-idf}$
	$d_{id,2000}^{tf-idf}$	$d_{en,2000}^{tf-idf}$
	$d_{id,3000}^{tf-idf}$	$d_{en,3000}^{tf-idf}$
	$d_{id,4000}^{tf-idf}$	$d_{en,4000}^{tf-idf}$
	$d_{id,5000}^{tf-idf}$	$d_{en,5000}^{tf-idf}$
	$d_{id,word2vec}^{we}$	$d_{en,word2vec}^{we}$
	$d_{id,fasttext}^{we}$	$d_{en,fasttext}^{we}$
	$d_{id,glove}^{we}$	$d_{en,glove}^{we}$
Word Embedding	$d_{id,all}^{we}$	$d_{en,all}^{we}$

Each instance of such structured data is partitioned into training data and testing data for the purposes of two graph neural network-based algorithms. The training data is employed to develop classification models. Subsequently, the classification model is evaluated on the test data to assess its effectiveness. The performance outcomes of each classification model, evaluated with the respective structured data produced, are presented in Table 11.

A comparative analysis of the findings from this investigation with those of prior research is illustrated in Table 12. This comparison elucidates that both Graph Convolutional Networks (GCN) and Graph Attention Networks (GAT) are capable of producing robust classification models in the context of identifying disease outbreak messages.

3.2 Discussion

Table 11 illustrates that the Graph Convolutional Network (GCN) and Graph Attention Network (GAT) models achieved the highest performance metrics of

93% and 92%, respectively, when applied to Indonesian datasets. The optimal GCN model was derived utilizing inputs from the Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction method, incorporating 5000 features. Conversely, the most effective GAT models were attained through the same TF-IDF feature extraction approach, albeit with 4000

features. In contrast, the superior model for English datasets yielded an accuracy of 86.77% when employing a combination of FastText-based feature extraction and the GCN algorithm, while an accuracy of 86.79% was recorded for the combination of FastText and GAT.

Table 11. Classification results of GCN and GAT using TF-IDF and word embedding features

Classification	Features	Accuracy	d_{id}			d_{en}			
			F1-score	Precision	Recall	Accuracy	F1-score	Precision	Recall
GCN	TF-IDF (1000)	87	86.97	87.37	87	76.86	77.43	78.03	76.86
	TF-IDF (2000)	85	84.88	86.17	85	80.99	80.14	79.38	80.99
	TF-IDF (3000)	86	85.99	86.06	86	83.47	81.79	80.58	83.47
	TF-IDF (4000)	86	85.98	86.23	86	80.17	79.6	79.07	80.17
	TF-IDF (5000)	93	93	93.02	93	84.3	83.03	82.12	84.30
	Word2Vec	68	67.79	68.47	68	84.3	81.55	79.90	84.30
	fastText	72	71.72	72.92	72	86.77	80.63	75.30	86.78
	Glove	67	66.84	67.34	67	85.12	81.06	78.96	85.12
	All	66	65.66	66.67	66	83.47	83.47	80.58	83.47
	TF-IDF (1000)	83	82.98	83.12	83	85.95	80.22	75.21	85.95
GAT	TF-IDF (2000)	87	86.94	87.74	87	83.47	81.02	79.33	83.47
	TF-IDF (3000)	91	90.96	91.82	91	81.82	78.1	74.70	81.82
	TF-IDF (4000)	92	92	92.07	92	80.99	77.66	74.60	80.99
	TF-IDF (5000)	86	85.99	86.06	86	85.95	84.81	84.11	85.95
	Word2Vec	68	67.95	68.12	68	86.78	80.63	75.30	86.78
	fastText	76	76	76	76	86.79	80.63	75.30	86.78
	Glove	72	71.72	72.92	72	86.78	80.63	75.30	86.78
	All	70	69.95	70.13	70	86.78	80.63	75.30	86.78
	TF-IDF (1000)	83	82.98	83.12	83	85.95	80.22	75.21	85.95
	TF-IDF (2000)	87	86.94	87.74	87	83.47	81.02	79.33	83.47

Table 12. Performance comparison of the proposed models with previous studies.

Dataset	Method		Accuracy (%)	Ref
	Features	Classification		
d_{id}	Word embedding	1D CNN	88.5	[12]
	Word embedding	LSTM	89	[3]
	-	BERT	82	[26]
	TF-IDF	GCN	93	Our
	5000			
d_{en}	TF-IDF	1D CNN	86.78	[12]
	3000			
	Glove	LSTM	86.78	[3]
	-	BERT	70	[33]
	-	COVID-Twitter-BERT	76	
	-	DistilBERT	80	[34]
	-	BERT-Base-Uncased	84	
	fastText	GAT	86.79	Our

The aforementioned optimal model outcomes indicate that the GCN performs proficiently on Indonesian datasets, whereas the GAT demonstrates superior efficacy compared to GCN with respect to English datasets. To ascertain the overall performance of the GCN and GAT models, an assessment of their average

accuracy was conducted, with the results presented in Figure 2.

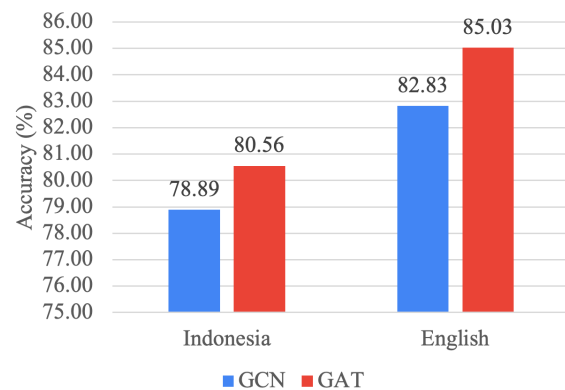


Figure 2. Comparison of average performance in GCN and GAT.

Figure 2 reveals that the average accuracy of the GAT model surpasses that of the GCN model across both Indonesian and English language datasets. The enhanced performance of GAT can be attributed to its implementation of attention mechanisms, which assign differential weights to relationships among words or documents. In textual data, it is recognized that not all words, documents, or similarity relations possess

equivalent levels of significance. GCN typically aggregates information from adjacent nodes uniformly, while GAT possesses the capability to select the most pertinent neighbouring nodes and diminish the impact of less significant or extraneous relations [29]. Consequently, GAT exhibits greater flexibility in leveraging TF-IDF features alongside word embeddings to effectively capture semantic and relational dynamics within text graphs.

In this research, the performance of models developed using Bag-of-Words (BoW) data inputs in conjunction with word embeddings was also examined. Figure 3 demonstrates that neither the BoW-based structured data input nor the word embedding approach is unequivocally superior.

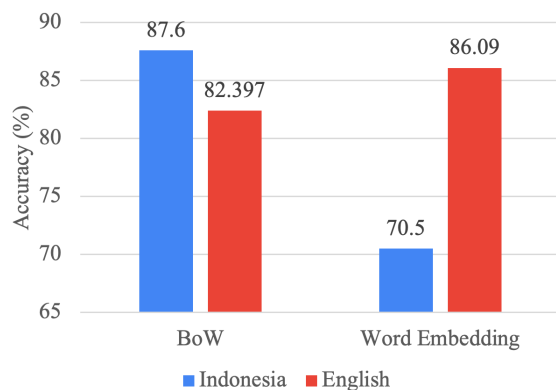


Figure 3. Comparison of average performance in BoW and word embedding.

In the context of Indonesian datasets, models that utilize Bag of Words (BoW) input exhibit superior performance in comparison to those applied to English datasets. This observation arises from the characteristics of the Indonesian language dataset, which comprises documents that are relatively concise in length or possess a limited number of words. Following the feature extraction process, which entails a variation in the number of features ranging from 1000 to 5000, minimal information is lost during the training phase, thereby enabling the model to function effectively. Conversely, English documents are characterised by a significantly greater word count. Consequently, the feature extraction phase results in considerable information loss, attributable to the restriction of features to a range of 1000 to 5000. This limitation constrains the model training process and detrimentally impacts the model's efficacy in classification tasks.

Models that use inputs derived from word-based feature extraction through embedding techniques in English scenarios are more effective, as the resulting information is comprehensive and free of truncation. This enhancement is due to converting each word in the document into a word embedding vector and then concatenating them.

The effect of the variation in the maximum number of features utilized in the BoW technique on the

performance metrics of the classification model is illustrated in Figure 4.

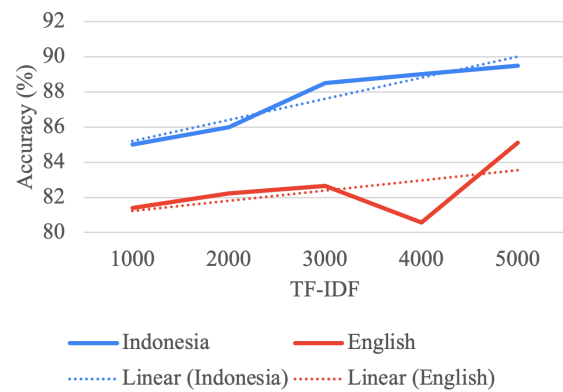


Figure 4. Average performance of models by maximal feature variation.

The values presented in Figure 4 are derived from average performance metrics calculated on the basis of maximum feature sets ranging from 1000 to 5000. Analyzing linear trends reveals that an increase in the number of utilized features correlates positively with an enhancement in model performance. This phenomenon occurs as a greater number of features provides a more comprehensive dataset for training the model [4].

In the context of this research, the efficacy of the model was further examined through structured data inputs employing word embedding methodologies. Figure 5 illustrates that features derived from FastText demonstrate strong performance across both Indonesian and English datasets. FastText employs an n-gram character-based representation, facilitating the model's ability to capture morphological nuances inherent in words. This characteristic renders FastText more adept at managing out-of-vocabulary (OOV) terms in comparison to GloVe and Word2Vec, which operate solely at the lexical level [48]. Moreover, FastText exhibits advantages in training efficiency, adaptability to specific datasets, and provides a more thorough semantic representation [46].

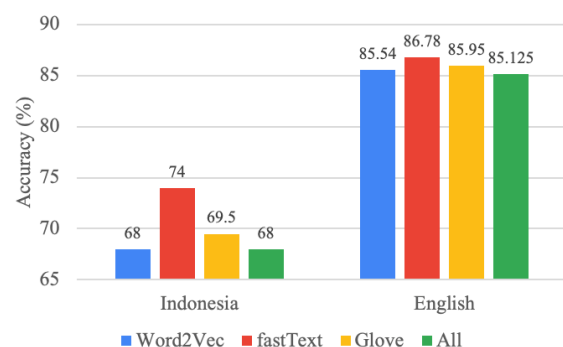


Figure 5. Average performance of models by word embedding method.

This research further explored the amalgamation of features from the three distinct word embedding techniques; however, the average performance observed was inferior to that of models utilizing a

singular word embedding approach. The decline in performance metrics may be attributable to the elevated dimensionality of the merged dataset, which, while potentially offering more comprehensive information, may simultaneously introduce greater noise. In the case of Graph Attention Networks (GAT), lower-dimensional dense inputs are frequently more beneficial, as GAT computes attention weights among adjacent nodes. Excessively noisy features can complicate the attention mechanism, rendering it more computationally intensive and less stable [27]. Conversely, Graph Convolutional Networks (GCN) exhibit a higher tolerance for high-dimensional inputs, thus suggesting that a decline in model performance may arise from data characterised by significant noise [29].

3.3 Limitations and Future Work

This research demonstrated that the efficacy of models generated through various implementations of TF-IDF based feature extraction methodologies and word embedding in conjunction with GNN-based algorithms surpassed that of earlier investigations employing CNN algorithms, LSTM, and BERT variants. Nonetheless, there remains potential for additional enhancement in performance, as the current accuracy stands at 93% for Indonesian datasets and 86.79% for English datasets.

This research is not without its limitations regarding the dimension of feature representation. In the context of TF-IDF based features, the feature count is constrained to a range of 1000 to 5000 features, resulting in the likelihood that significant information within the documents may be omitted, particularly in English datasets characterized by longer document lengths. This scenario may inhibit the optimisation of the model training process, as not all textual data is utilized in the construction of data representations. Furthermore, the application of TF-IDF fails to adequately capture the intricacies of word order and semantic nuances, thereby rendering the model predominantly reliant on the frequency of word occurrences within the documents. Additional constraints are evident in the utilisation and integration of word embedding features. While word embedding is capable of generating superior semantic representations, the findings of this study indicate that the amalgamation of Word2Vec, FastText, and GloVe does not necessarily yield improvements in model performance. This may be due to an excessive increase in feature dimensions that is laden with noise, which compromises the stability of the learning process.

For future inquiries, please explore more effective strategies for feature representation, particularly for high-dimensional textual data. The application of TF-IDF may be enhanced by varying the number of features more broadly or by merging word embedding features to ensure the data encapsulates sequential information and semantic meanings. However, this approach carries the potential risk of generating high-dimensional data, necessitating its integration with feature selection and dimensionality reduction methodologies.

In future research, it is also essential to contemplate the deployment of alternative forms of input representation for GCN and GAT, such as sentence-based representation embeddings or contextual embeddings like BERT, RoBERTa, or IndoBERT. Moreover, further exploration into developing a hybrid methodology that combines these two variants of GNN with other approaches is also deemed necessary.

4. Conclusions

This study shows that GNN-based models can effectively classify disease outbreak messages in both Indonesian and English datasets. For the Indonesian dataset, the best GCN model achieved the highest accuracy of 93% using TF-IDF with 5000 features, while the best GAT model achieved 92% using TF-IDF with 4000 features. For the English dataset, the best GCN model achieved 86.77% using fastText, while the best GAT model achieved the highest accuracy of 86.79% using fastText. These results indicate that TF-IDF is more effective for the shorter Indonesian texts, while fastText is more suitable for the longer English texts because it can preserve richer semantic and morphological information.

Compared with previous studies, the proposed models achieved better performance than several text classification approaches based on CNN, LSTM, and BERT variants. In the Indonesian dataset, the proposed GCN model with TF-IDF 5000 outperformed 1D CNN, LSTM, and BERT, which achieved accuracies of 88.5%, 89%, and 82%, respectively. In the English dataset, the proposed GAT model with fastText slightly outperformed previous CNN and LSTM models with 86.78% accuracy, as well as BERT-based models such as BERT, COVID-Twitter-BERT, DistilBERT, and BERT-Base-Uncased. The key contribution of this work is the investigation of GCN and GAT combined with different feature extraction techniques, including TF-IDF and word embedding, for multilingual disease outbreak message classification.

However, this study has several limitations. The TF-IDF features were limited to 1000–5000 features, which may cause information loss, especially in longer English texts. In addition, combining Word2Vec, fastText, and GloVe did not always improve performance because the concatenated representation may increase feature dimensionality and noise. Future work should explore more optimal feature representation strategies, including feature selection, dimensionality reduction, sentence embedding, contextual embedding such as BERT, RoBERTa, or IndoBERT, and hybrid models that combine GCN or GAT with other deep learning architectures to improve classification performance and model stability.

Declarations

AI Usage Statement

The authors used Scopus AI only to assist in identifying

relevant references related to the research topic. SciSpace and Grammarly were used solely to improve the manuscript's readability, grammar, and language clarity. The authors confirm that they critically reviewed and verified all AI-assisted outputs. All scientific content, research design, data analysis, interpretation of results, arguments, conclusions, and final decisions presented in this manuscript are entirely the responsibility of the authors.

Author Contributions

Irwan Budiman: conceptualization, methodology, writing - original draft; Mohammad Reza Faisal: data curation, software, writing - original draft; Muliadi: resources, investigation, validation; Muhammad Rafi: software, data curation; Husnil Kamil: supervision, writing - review & editing, visualization; Luther Alexander Latumakulita: supervision, writing - reviewing and editing; Luu Duc Ngo: software, validation.

Funding Statement

This research was supported by the Directorate of Research and Community Service, Directorate General of Research and Development, Ministry of Higher Education, Science, and Technology of the Republic of Indonesia, under Main Contract Number: 075/C3/DT.05.00/PL/2025 and Derivative Contract Number: 1396/UN8.2/PG/2025.

Conflict of Interest Statement

The authors declare no conflict of interest.

Ethical Clearance

The social media dataset analyzed in this work was obtained from a publicly available dataset released by previous researchers and hosted on GitHub for research purposes. The dataset was used only for developing and evaluating an AI-based text classification model. This study did not collect new data directly from X users, did not involve direct interaction with users, and did not attempt to identify any individual user represented in the dataset. Personally identifiable information, including usernames, user IDs, profile links, and other direct identifiers, was not collected, disclosed, or reported in the manuscript.

Data and Code Availability

The dataset used in this study is publicly available on [GitHub](#). The source code developed for this study is available from the corresponding author upon reasonable request.

References

- [1] L. Tan, J. Guo, S. Mohanarajah, and K. Zhou, "Can we detect trends in natural disaster management with artificial intelligence? A review of modeling practices," *Natural Hazards*, vol. 107, no. 3, pp. 2389–2417, 2021, [doi:10.1007/s11069-020-04429-3](#).
- [2] S. Muppidi, P. S. Rao, and M. Rama Krishna Murthy, "Identification of natural disaster affected area using twitter," in *Advances in Decision Sciences, Image Processing, Security and Computer Vision*, Springer, 2020, pp. 792–801, [doi:10.1007/978-3-030-24322-7_92](#).
- [3] R. Amalia et al., "Identification of Social Media Posts Containing Self-reported COVID-19 Symptoms using Triple Word Embeddings and Long Short-Term Memory," *Telematika*, vol. 17, no. 1, pp. 27–38, 2024, [doi:10.35671/telematika.v17i1.2774](#).
- [4] R. A. Rizian, I. Budiman, M. R. Faisal, D. Kartini, F. Indriani, and U. A. Ahmad, "Enhancing Classification of Self-Reported Monkeypox Symptoms on Social Media Using Term Frequency-Inverse Document Frequency Features and Graph Attention Networks," *Jurnal Teknik Informatika (Jutif)*, vol. 6, no. 6, pp. 5865–5881, 2025, [doi:10.52436/1.jutif.2025.6.6.5482](#).
- [5] Y. Zhang, M. Ibaraki, and F. W. Schwartz, "Disease surveillance using online news: Dengue and Zika in tropical countries," *J. Biomed. Inform.*, vol. 102, p. 103374, 2020, [doi:10.1016/j.jbi.2020.103374](#).
- [6] R. Meckawy, D. Stuckler, A. Mehta, T. Al-Ahdal, and B. N. Doebbeling, "Effectiveness of early warning systems in the detection of infectious diseases outbreaks: a systematic review," *BMC Public Health*, vol. 22, no. 1, p. 2216, 2022, [doi:10.1186/s12889-022-14625-4](#).
- [7] A. E. Wilson, C. U. Lehmann, S. N. Saleh, J. Hanna, and R. J. Medford, "Social media: a new tool for outbreak surveillance," *Antimicrobial Stewardship & Healthcare Epidemiology*, vol. 1, no. 1, p. e50, 2021, [doi:10.1017/ash.2021.225](#).
- [8] Y. Liscano, L. A. Anillo Arrieta, J. F. Montenegro, D. Prieto-Alvarado, and J. Ordoñez, "Early warning of infectious disease outbreaks using social media and digital data: A scoping review," *Int. J. Environ. Res. Public Health*, vol. 22, no. 7, p. 1104, 2025, [doi:10.3390/ijerph22071104](#).
- [9] R. Lamsal, M. R. Read, and S. Karunasekera, "CrisisTransformers: Pre-trained language models and sentence encoders for crisis-related social media texts," *Knowl. Based. Syst.*, vol. 296, p. 111916, 2024, [doi:10.1016/j.knosys.2024.111916](#).
- [10] A. Pak, A. Ziyaden, T. Saparov, I. Akhmetov, and A. Gelbukh, "Word Embeddings: A Comprehensive Survey," *Computacion y Sistemas*, vol. 28, no. 4, pp. 2005 – 2029, 2024, [doi:10.13053/CyS-28-4-5225](#).
- [11] Sutriawan, S. Rustad, G. F. Shidik, and Pujiono, "Performance Evaluation of Text Embedding Models for Ambiguity Classification in Indonesian News Corpus: A Comparative Study of TF-IDF, Word2Vec, FastText BERT, and GPT," *Ingenierie des Systemes d'Information*, vol. 30, no. 6, pp. 1469 – 1482, 2025, [doi:10.18280/isi.300606](#).
- [12] M. Khairie, M. R. Faisal, R. Herteno, I. Budiman, F. Abadi, and M. I. Mazdadi, "The Effect of Channel Size on Performance of 1D CNN Architecture for Automatic Detection of Self-Reported COVID-19 Symptoms on Twitter," in *2023 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2023, pp. 621–625, [doi:10.1109/ISITIA59021.2023.10220444](#).
- [13] K. Yusupov, M. R. Islam, I. Muminov, M. Sahlabadi, and K. Yim, "Comparative Analysis of Machine Learning and Deep Learning Models for Email Spam Classification Using TF-IDF and Word Embedding Techniques," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 231, pp. 114 – 122, 2025, [doi:10.1007/978-3-031-76452-3_11](#).
- [14] C. M. Rohith, K. B. Sivachandra, N. Mohan, and S. S. Kumar, "Impact of Fine-Tuned Embeddings on Machine Learning-Based Causal Detection," in *2025 5th International Conference on Intelligent Technologies, CONIT 2025*, [doi:10.1109/CONIT65521.2025.11167597](#).
- [15] K. M. A. Kumar, S. G. Anusha, H. N. Sharath, M. Milind, and M. Charan, "Impact of Different Embeddings on Text Classification," in *2024 International Conference on Recent Advances in Science and Engineering Technology (ICRASET)*, 2024, pp. 1–6, [doi:10.1109/ICRASET63057.2024.10895636](#).
- [16] A. M. Alayba and M. Altamimi, "Optimization of Arabic text classification using SVM integrated with word embedding models on a novel dataset," *International Journal of*

- Advanced and Applied Sciences*, vol. 12, no. 9, pp. 140 – 151, 2025, [doi:10.21833/jjaas.2025.09.013](https://doi.org/10.21833/jjaas.2025.09.013).
- [17] M. Danish and M. Amjad, *Comparative Analysis of Word-Embedding Techniques Using LSTM Model*. 2023. [doi:10.1201/9781003371380-3](https://doi.org/10.1201/9781003371380-3).
 - [18] E. M. Dharma, F. L. Gaoi, H. L. H. S. Warnars, and B. Soewito, "the Accuracy Comparison Among Word2Vec, Glove, and Fasttext Towards Convolution Neural Network (Cnn) Text Classification," *J. Theor. Appl. Inf. Technol.*, vol. 100, no. 2, pp. 349–359, 2022, [doi:10.35870/jimik.v4i3.289](https://doi.org/10.35870/jimik.v4i3.289).
 - [19] W. Zhao, L. Zhu, M. Wang, X. Zhang, and J. Zhang, "WTL-CNN: a news text classification method of convolutional neural network based on weighted word embedding," *Conn. Sci.*, vol. 34, no. 1, pp. 2291 – 2312, 2022, [doi:10.1080/09540091.2022.2117274](https://doi.org/10.1080/09540091.2022.2117274).
 - [20] M. A. B. Al-Tarawneh, O. Al-irr, K. S. Al-Maaitah, H. Kanj, and W. H. F. Aly, "Enhancing Fake News Detection with Word Embedding: A Machine Learning and Deep Learning Approach," *Computers*, vol. 13, no. 9, 2024, [doi:10.3390/computers13090239](https://doi.org/10.3390/computers13090239).
 - [21] S. Bodapati, H. Bandarupally, R. N. Shaw, and A. Ghosh, "Comparison and analysis of RNN-LSTMs and CNNs for social reviews classification," in *Advances in Applications of Data-Driven Computing*, Springer, 2021, pp. 49–59.
 - [22] Z. Zhai, X. Zhang, F. Fang, and L. Yao, "Text classification of Chinese news based on multi-scale CNN and LSTM hybrid model," *Multimed. Tools Appl.*, pp. 1–14, 2023, [doi:10.1007/s11042-023-14450-w](https://doi.org/10.1007/s11042-023-14450-w).
 - [23] A. C. M. V. Srinivas, Ch. Satyanarayana, Ch. Divakar, and K. P. Sirisha, "Sentiment Analysis using Neural Network and LSTM," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1074, no. 1, p. 012007, 2021, [doi:10.1088/1757-899x/1074/1/012007](https://doi.org/10.1088/1757-899x/1074/1/012007).
 - [24] N. Rai, D. Kumar, N. Kaushik, C. Raj, and A. Ali, "Fake News Classification using transformer based enhanced LSTM and BERT," *International Journal of Cognitive Computing in Engineering*, vol. 3, pp. 98–105, 2022, [doi:10.1016/j.ijcce.2022.03.003](https://doi.org/10.1016/j.ijcce.2022.03.003).
 - [25] M. Khadhraoui, H. Bellaaj, M. Ben Ammar, H. Hamam, and M. Jmaiel, "Survey of BERT-base models for scientific text classification: COVID-19 case study," *Applied Sciences*, vol. 12, no. 6, p. 2891, 2022, [doi:10.3390/app12062891](https://doi.org/10.3390/app12062891).
 - [26] I. Budiman et al., "Classification Performance Comparison of BERT and IndoBERT on SelfReport of COVID-19 Status on Social Media," *Journal of Computer Sciences Institute*, vol. 30, pp. 61–67, 2024, [doi:10.35784/jcsi.5564](https://doi.org/10.35784/jcsi.5564).
 - [27] N. R. Paul and R. C. Balabantaray, "Disaster related tweet classification method based on BERT and GAT," *International Journal of Information Technology*, pp. 1–13, 2025, [doi:10.1007/s41870-024-02389-6](https://doi.org/10.1007/s41870-024-02389-6).
 - [28] Z. Chen, K. Huang, L. Wu, Z. Zhong, and Z. Jiao, "Relational Graph Convolutional Network for Text-Mining-Based Accident Causal Classification," *Applied Sciences*, vol. 12, no. 5, 2022, [doi:10.3390/app12052482](https://doi.org/10.3390/app12052482).
 - [29] H. Ren et al., "Graph convolutional networks in language and vision: A survey," *Knowl. Based. Syst.*, vol. 251, 2022, [doi:10.1016/j.knsys.2022.109250](https://doi.org/10.1016/j.knsys.2022.109250).
 - [30] S. M. Haider Rizvi, R. Imran, and A. Mahmood, "Text classification using graph convolutional networks: A comprehensive survey," *ACM Comput. Surv.*, vol. 57, no. 8, pp. 1–38, 2025, [doi:10.1145/3714456](https://doi.org/10.1145/3714456).
 - [31] P. Pham, L. T. T. Nguyen, W. Pedrycz, and B. Vo, "Deep learning, graph-based text representation and classification: a survey, perspectives and challenges," *Artif. Intell. Rev.*, vol. 56, no. 6, pp. 4893–4927, 2023, [doi:10.1007/s10462-022-10265-7](https://doi.org/10.1007/s10462-022-10265-7).
 - [32] A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph attention networks: a comprehensive review of methods and applications," *Future Internet*, vol. 16, no. 9, p. 318, 2024, [doi:10.3390/fi16090318](https://doi.org/10.3390/fi16090318).
 - [33] A. Z. Klein, A. Magge, K. O'Connor, J. I. Flores Amaro, D. Weissenbacher, and G. Gonzalez Hernandez, "Toward using Twitter for tracking COVID-19: a natural language processing pipeline and exploratory data set," *J. Med. Internet Res.*, vol. 23, no. 1, p. e25314, 2021, [doi:10.2196/25314](https://doi.org/10.2196/25314).
 - [34] A. Z. Klein, S. Kunatharaju, K. O'Connor, and G. Gonzalez-Hernandez, "Automatically identifying self-reports of COVID-19 diagnosis on Twitter: an annotated data set, deep neural network classifiers, and a large-scale cohort," *J. Med. Internet Res.*, vol. 25, p. e46484, 2023, [doi:10.2196/46484](https://doi.org/10.2196/46484).
 - [35] A. Tyagi, V. K. Jain, and V. Kumar, "Benchmark Text Preprocessing Techniques in Natural Language Processing," in *Proceedings - 2024 4th International Conference on Innovative Sustainable Computational Technologies, CISCT 2024*, 2024, [doi:10.1109/CISCT62494.2024.11134188](https://doi.org/10.1109/CISCT62494.2024.11134188).
 - [36] Y. Chai, Y. Liu, W. Li, B. Zhu, H. Liu, and Y. Jiang, "An interpretable wide and deep model for online disinformation detection," *Expert Syst. Appl.*, vol. 237, 2024, [doi:10.1016/j.eswa.2023.121588](https://doi.org/10.1016/j.eswa.2023.121588).
 - [37] N. A. Semary, W. Ahmed, K. Amin, P. Pławiak, and M. Hammad, "Enhancing machine learning-based sentiment analysis through feature extraction techniques," *PLoS One*, vol. 19, no. 2 February, 2024, [doi:10.1371/journal.pone.0294968](https://doi.org/10.1371/journal.pone.0294968).
 - [38] Z. Shuai et al., "Comparison of different feature extraction methods for applicable automated ICD coding," *BMC Med. Inform. Decis. Mak.*, vol. 22, no. 1, 2022, [doi:10.1186/s12911-022-01753-5](https://doi.org/10.1186/s12911-022-01753-5).
 - [39] A. Pak, A. Ziyaden, T. Saparov, I. Akhmetov, and A. Gelbukh, "Word Embeddings: A Comprehensive Survey," *Computacion y Sistemas*, vol. 28, no. 4, pp. 2005 – 2029, 2024, [doi:10.13053/CyS-28-4-5225](https://doi.org/10.13053/CyS-28-4-5225).
 - [40] M. N. Salim and B. S. Mustafa, "A Survey on Word Representation in Natural Language Processing," in *AIP Conference Proceedings*, 2022, [doi:10.1063/5.0121147](https://doi.org/10.1063/5.0121147).
 - [41] B. Samatha, N. Karyemsetty, D. S. Kumar, D. K. Rao, G. Mani, and T. Syamsundararao, "Analysis of a Multichannel Learning Mechanism for Speech Detection in Social Networks," in *Proceedings of the International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics, IICITCEE 2023*, 2023, pp. 1066 – 1071. [doi:10.1109/IICITCEE57236.2023.10091086](https://doi.org/10.1109/IICITCEE57236.2023.10091086).
 - [42] A. Balamurali and B. Ananthanarayanan, *Develop a Neural Model to Score Bigram of Words Using Bag-of-Words Model for Sentiment Analysis*, vol. 1–4. 2022, [doi:10.4018/978-1-6684-6303-1.ch034](https://doi.org/10.4018/978-1-6684-6303-1.ch034).
 - [43] C. Sun et al., "Attention-based graph neural networks: a survey," *Artif. Intell. Rev.*, vol. 56, no. Suppl 2, pp. 2263–2310, 2023.
 - [44] G. Lachaud, P. Conde-Cespedes, and M. Trocan, "Mathematical Expressiveness of Graph Neural Networks," *Mathematics*, vol. 10, no. 24, 2022, [doi:10.3390/math10244770](https://doi.org/10.3390/math10244770).
 - [45] A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph Attention Networks: A Comprehensive Review of Methods and Applications," *Future Internet*, vol. 16, no. 9, 2024, [doi:10.3390/fi16090318](https://doi.org/10.3390/fi16090318).
 - [46] C. N. Tulu, "Experimental Comparison of Pre-Trained Word Embedding Vectors of Word2Vec, Glove, FastText for Word Level Semantic Text Similarity Measurement in Turkish," *Advances in Science and Technology Research Journal*, vol. 16, no. 4, pp. 147 – 156, 2022, [doi:10.12913/22998624/152453](https://doi.org/10.12913/22998624/152453).