African Journal of Advanced Pure and Applied Sciences (AJAPAS)
Online ISSN: 2957-644X
Volume 4, Issue 3, 2025
Page No: 158-170
Website: https://aaasjournals.com/index.php/ajapas/index
ISI 2025: 1.126 SJIFactor 2024: 6.752 معامل التأثير العربي: 1.62

The impact of Text Representation on Classification Accuracy

Mona Ali Mohammed [1 *], Reem.Abdalhadi Alsunousi [2], Anwar Alhenshiri [3]
[1,2] Computer Science Department, Faculty of Science, Omar Al-Mukhtar University, Al Bayda, Libya
[3] Computer Science Department, Faculty of Information Technology, Misurata University, Misurata, Libya

تأثير تمثيل النص على دقة التصنيف

منى علي محمد[1*]، ريم عبدالهادي السنوسي[2] ، أنور الهنشيري[3]
[2،1] قسم الحاسوب، كلية العلوم، جامعة عمر المختار، البيضاء، ليبيا
[3] كلية تقنية المعلومات، جامعة مصراتة، مصراتة، ليبيا

*Corresponding author: mona.boshebh@omu.edu.ly

Received: June 11, 2025	Accepted: August 02, 2025	Published: August 12, 2025

Abstract:

People gladly use social media to express their opinions. Emotional techniques of analysis sometimes capacitate community to harness the treasure of significant data that is included in unstructured social media information. We must ensure that the dataset is a high-quality one before the design and deployment of machine learning models. However, to run ML algorithms on text data we need to convert the content into numerical representations by one of the text representation methods during the data preprocessing stage. This study explores the usage of eight text representation methods with three ML models evaluated on a Twitter sentiment classification dataset. In particular, the experiments aim to test the impact of text length in the results classification. For this reason, the experiments were conducted with different ranges of text lengths. However, to run the experiments and evaluate the results we use common classification algorithms which include: Logistic Regression, Naïve Bayes, and Support Vector Machines. The results showed that the best performance for all models was when using Count Vectorizer to represent the text with N-gram range (2,3,4), respectively, followed by TF-IDF, while Doc2Vec, Word2Vec, and GloVe performed averagely. For the text length, the model's performance decreases as the text length increases with all models. It was also noted that Doc2Vec, Word2Vec, and GloVe kept the performance of the models despite the change in text length and generally gave average accuracy compared to Count Vectorizer. However, the SVM classifier has surpassed all the other techniques in the whole experiment.

Keywords: Text representation, classification task, classification datasets, Naïve Bayes model, Logistic Regression classifier, Support Vector Machines.

الملخص

يستخدم الناس وسائل التواصل الاجتماعي للتعبير عن آرائهم. تُمكّن تقنيات التحليل العاطفي أحيانًا المجتمع من الاستفادة من كنز البيانات المهمة المُضمنة في معلومات وسائل التواصل الاجتماعي غير المُهيكلة. يجب التأكد من جودة مجموعة البيانات قبل تصميم نماذج التعلم الآلي ونشرها. ومع ذلك، لتشغيل خوارزميات التعلم الآلي على بيانات النصوص، نحتاج إلى تحويل المحتوى إلى تمثيلات رقمية باستخدام إحدى طرق تمثيل النصوص خلال مرحلة المعالجة المسبقة للبيانات. تستكشف هذه الدراسة استخدام ثماني طرق لتمثيل النصوص، مع تقييم ثلاثة نماذج تعلم آلي على مجموعة بيانات لتصنيف المشاعر على تويتر. تهدف التجارب تحديدًا إلى اختبار تأثير طول النص في تصنيف النتائج. لهذا السبب، أُجريت التجارب على نطاقات مختلفة من أطوال النصوص. ولإجراء التجارب وتقييم النتائج، نستخدم خوارزميات تصنيف شائعة تشمل: الانحدار اللوجستي، بايز الساذج، وآلات المتجهات الداعمة. أظهرت النتائج أن أفضل أداء لجميع النماذج كان عند استخدام Count Vectorizer لتمثيل النص ذي النطاق N-gram (2،3،4) على التوالي، يليه TF-IDF، بينما كان أداء Doc2Vec وWord2Vec وGloVe متوسطًا. بالنسبة لطول النص، انخفض أداء النموذج مع زيادة طوله في جميع النماذج. كما لوحظ

أن Doc2Vec وWord2Vec وGloVe حافظت على أداء النماذج على الرغم من تغير طول النص، وقدمت عمومًا دقة متوسطة مقارنةً بـ Count Vectorizer. كذلك تفوق مصنف SVM على جميع التقنيات الأخرى في التجربة بأكملها.

الكلمات المفتاحية: تمثيل النصوص، مهمة التصنيف، مجموعات بيانات التصنيف، نموذج بايز الساذج، مُصنِّف الانحدار اللوجستي، آلات المتجهات الداعمة.

Introduction

Availability of communication media, social media has evolved into a powerful and influential communication channel in the modern digital age. These platforms have greatly impacted people's daily lives recently. people gladly use the social media to expel their opinions freely, arguments and feelings in broad range of themes [2, 4] social media provides an opportunity for business to connect with their customers by advertisements and speaking directly to the customers to know their perspective of products and services [3, 11, 15, 20, 30]

Besides, many customers provide reviews and feelings about diver's services and output, User reviews and estimating on multiple service providers and stage encourage sellers to improve their existing goods, systems, or services. As social media platforms gain widespread usage, customers continuously produce massive volumes of content and feedback [3, 11, 30, 39]. As the amount of information continues to grow, effectively extracting useful information from this enormous volume of data have become a critical issue [22, 39]. This requires more resources in terms of manpower and time to manage and transform unstructured information into significance insights which can aid in making the decision [10, 23]. In addition to that, monitoring public sentiment in a well-timed demeanor, Timely analysis of public sentiment has become essential for both governmental bodies and business entities in today's fast-paced environment. Lastly, the swift and significant advancement of data technology has made social media stages like Twitter an essential part of modern life to participate in their views with the rest of the planet globe. Sentiment recognition and sentiment analysis both are critical fields in natural language processing. Although these toponyms are sometimes employe inter- substitutable, they vary in some techniques.

Emotional analysis is a way to assess whether information is negative, positive, or unbiased. Per contra, sentiment detection is a way to identify distinct creatural feelings such as joy, depression, or anger. Terms such as "sentiment detection," "affective computing," "sentiment analysis," and "emotion identification" are often used interchangeably, although they may differ slightly in scope or application [10,21]

Sentiment analysis techniques capacitate organizations, governments, and individuals to harness the opulence of significance data included in unstructured social media information. Researchers are exploring the use of big data techniques for data storage, access, and processing efficiency for effective data analysis, leading to the development of automated methods for extracting sentiment or opinions from text [3,11, 20, 30, 39]

Prior to developing machine learning (ML) models, it is essential to ensure that the dataset is properly refined and accurately represents the underlying information, allowing the model to learn effectively. To apply machine learning algorithms to text data, it must first be transformed into a numerical format using appropriate text representation techniques as part of the preprocessing pipeline. [3,11, 12, 20, 30, 39]

Effective text representation plays a crucial role in ensuring high performance in classification tasks. The quality of the initial feature representation establishes a baseline that even advanced ML models may struggle to surpass [18, 29]. 35, 37]. However, the existing literature shows a variety of text representation methods used in sentiment classifiers, which makes it difficult for researchers and practitioners to identify the most appropriate method for their specific use case or domain of interest. The study explores the usage of eight text representation methods with three ML models evaluated on the Twitter sentiment classification dataset. In particular, the experiments aimed to evaluate how variations in text length influence classification performance; for this reason, the experiments were conducted to examine the influence of text length on classification outcomes; therefore, the experiments were executed with a varied range of text lengths. However, the experimental evaluation was carried out using widely adopted classification algorithms, including Naïve Bayes (NB), Logistic Regression (LR), and Support Vector Machines (SVM).[1, 7, 19]

The structure of this study is organized as follows: Section 2 reviews related work; Section 3 outlines the methodology and tools employed. Section 4 describes the experimental procedures, while Section 5 discusses the results and provides interpretation. Finally, Section 6 concludes the study with key findings and recommendations.

Literature Review

One of previous research [24] the researchers traced the development of text representations since the 1970s, The study traced the progression of text representation methods from early techniques like regular expressions to more advanced vector-based models used for encoding raw textual input. Additionally, it emphasized the evolution of

the NLP field, shifting from rule-based and statistical methods to data-driven and context-aware representations. Each embedding approach was discussed in terms of its structure, the problems it attempts to solve, its constraints, and its real-world applications. Another study [41]reviewed sophisticated text representation techniques, involving embeddings and vectorization, prioritizing methods like word2vec, GloVe, and transformer-based embeddings, which convert text into numerical forms for machine processing and improve performance in multiple NLP tasks. The results showed that the success of NLP applications, such as sentiment analysis, machine translation, and information retrieval, can be highly influenced by the choice of text representation technique. The authors of the paper [17] addressed a broad range of machine learning algorithms, including Support Vector Machines (SVM), Random Forest, Logistic Regression (LR), and Naïve Bayes (NB). As for deep learning strategies, it considered models such as Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks. RoBerta, and deployment of BERT embeddings in Recurrent Neural Networks were examined. The experimentation included massive datasets taken from two social media platform, Reddit and Twitter. By highlighting the varying effects of TF-IDF and BoW, the findings highlighted which techniques were most impactful for stress detection and contributed to improving the design of future detection systems. The study[33] offered a detailed compilation of the latest text representation (TR) methods, in order to create linguistically rich embeddings, the study clarified the methods motivation, and explained how they deploy the distributional hypothesis. In addition, the study provided a labeled genealogy of TR methods, performed a detailed analysis from a conceptual and chronological perspective, and outlined methods that marked significant NLP advancements due to their particular architectural design, leading to advanced performance on downstream tasks. The study by [24] discussed the newest developments in text representation and embedding techniques, such as pre-trained language models for scientific text, biomedical language representation models, and multi-modal embeddings for emotion recognition. Additionally, it examined the employments and limitations of these techniques, presenting a comprehensive perspective on how representation models and embedding approaches have evolved within the field of NLP. The researchers in[34] shed light on the impact of NLP on requirement engineering tasks. The review conducted a comprehensive analysis of how different representations have been applied in the context of requirement engineering (RE), outlining the primary themes addressed, prevailing research trends, existing limitations, and proposed directions for future investigation. The study conducted by[5] introduced a new classification task centered on tweet topics and provided two related datasets for training and testing tweet classification models. The report addressed the challenges that would be faced when categorizing social media content, also they examined the shortcomings of existing solutions, such as topic modeling and topic classification

Material and methods

This study employed a set of supervised ML techniques aimed at addressing text classification tasks, where each data instance is associated with a predefined categorical label. Supervised learning models are trained using annotated datasets that contain both input features and their corresponding class outputs. The classification task in this research was approached using three distinct algorithms: the probabilistic Naïve Bayes (NB) classifier, the linear-based Logistic Regression (LR) model, and the margin-optimization method known as Support Vector Machines (SVM). These algorithms were selected based on their balance between interpretability, computational efficiency, and suitability for handling sparse and high-dimensional data, such as that found in textual content.

NAIVE BAYES CLASSIFIER (NB)

It is considered among the most practical and commonly applied algorithms in the field of data mining. It offers a simple yet effective approach to building classification models by assigning categories to input vectors composed of "n" features, with each class selected from a predefined limited group of categories [28] The model is grounded in Bayes' Theorem and operates by estimating the posterior probability to determine the most likely class for a given input. This relationship is formulated as follows:

$$P(A\backslash B) = (P(B\backslash A).P(A))/\,p(B)$$

NB has demonstrated strong performance across a variety of complex, real-world applications. Its popularity stems from its conceptual simplicity, ease of implementation, and low computational cost. Compared to many alternative classification algorithms, NB typically requires smaller training datasets and involves estimating fewer model parameters[6, 28]. The core assumption behind this model is that each feature contributes independently to the probability of a given class, regardless of the presence or values of other features. This underlying principle is formalized as follows:

$$X = (x1, x2, x3, \ldots\ldots\ldots, xn)$$

$$P(y\backslash x1, x2, \ldots . . xn) = \frac{p(x1\backslash y).\, p(x2\backslash y) \ldots . . p(xn\backslash y).\, p(y)}{p(x1).\, p(x2) \ldots . . p(xn)}$$

$$P(y\backslash x1, x2, \ldots . . xn) \propto p(y).\prod_{i=1}^{n} p(xi\backslash y)$$

$$y = \text{argmax}_y\, [p(y).\prod_{i=1}^{n} p(xi\backslash y)]$$

SUPPORT VECTOR MACHINE (SVM)

It is a supervised learning algorithm designed to address both classification and regression problems[6]. It functions by identifying an optimal decision boundary—referred to as a hyperplane—within a feature space of N dimensions, where N represents the number of input attributes [27]. The primary goal of SVM is to construct this hyperplane in a way that separates instances of different classes with the maximum possible margin, allowing for better generalization on unseen data.
To formulate the classification task, SVM seeks the optimal hyperplane that maximizes the margin between data classes. This is achieved by solving the following optimization problem:

$$\underset{w,b}{\text{minimize}} \frac{1}{2}||w||^2$$

subject to the constraint:

$$y_i(w^T x_i + b) \geq 1 \quad for\ i = 1,2,3,\cdots,m$$

Here, w is the weight vector that defines the orientation of the hyperplane, and b is the bias term. The constraint ensures that each training instance is classified correctly and lies on the appropriate side of the margin[8].

LOGISTIC REGRESSION (LR)

It is a supervised statistical learning model commonly used for binary classification tasks [39, 40]. It models the relationship between a set of independent features (predictors) and a categorical target variable by estimating the probability that a given input belongs to a particular class. This probability is obtained by applying the sigmoid function to a linear combination of the input features and corresponding weights [28].

The prediction function is expressed as:

$$P(y = 1 \mid x) = \sigma(w^\top x + b) = \frac{1}{1 + e^{-(w^\top x + b)}}$$

In this formulation, x denotes the input feature vector, while w represent the corresponding weight vector. The term b serves as the bias or interception, adjusting the decision boundary. The function $\sigma(z)$, known as the sigmoid function, transforms the linear combination $(w^T x + b)$ into a probability value bounded between 0 and 1, allowing for probabilistic interpretation of the model's output.

This formulation enables the model to produce outputs interpretable as probabilities, making LR particularly suitable for problems with binary outcomes. Despite being a linear model, it performs well in many practical scenarios due to its simplicity, computational efficiency, and ease of interpretation[16].

PERFORMANCE EVALUATION CRITERIA

Evaluating the effectiveness of classification models is a crucial step in determining their reliability and generalization capability. In sentiment analysis tasks, several performance indicators are commonly utilized, including accuracy, precision, recall, F1-score, computational time, and memory usage. Among these, accuracy, precision, recall, and the F1-measure are the most frequently reported in the literature due to their clarity and relevance to classification outcomes[39].

To calculate these metrics, the confusion matrix is typically employed. It provides a tabular representation of the model's predictions by comparing them with actual labels, thus offering insight into the model's strengths and weaknesses. Table 1 illustrates the general structure of the confusion matrix using four key components:

True Positives (A): instances correctly predicted as positive,
False Positives (B): negative cases incorrectly predicted as positive,
False Negatives (C): positive cases incorrectly predicted as negative,
True Negatives (D): instances correctly predicted as negative

Table 1: Classification models evaluation (Confusion Matrix).

<table>
<tr><th colspan="2" rowspan="2">Confusion Matrix</th><th colspan="2">Targer</th><th colspan="2" rowspan="2">--</th></tr>
<tr><th>Positive</th><th>Negative</th></tr>
<tr><td rowspan="2">Model</td><td>Positive</td><td>A</td><td>B</td><td>Positive predictive value</td><td>a/(a+b)</td></tr>
<tr><td>Negative</td><td>C</td><td>D</td><td>Negative predictive value</td><td>d/(c+d)</td></tr>
<tr><td colspan="2" rowspan="2">--</td><td>Sensitivity</td><td>Specificity</td><td colspan="2" rowspan="2">Accuracy=(a+d)/(a+b+c+d)</td></tr>
<tr><td>a/(a+c)</td><td>d/(b+d)</td></tr>
</table>

Using these components, the following performance metrics can be derived:

Accuracy: represents the ratio of correct predictions (true positives and true negatives) to the total number of evaluated cases.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision (Positive Predictive Value): reflects the proportion of true positive predictions among all positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall (Sensitivity): represents the proportion of correctly predicted positive cases.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Specificity: Indicates the percentage of accurately determined true negatives.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

F1-score: harmonic mean of precision and recall, particularly useful for imbalanced datasets.

$$F1 - \text{score} = 2.\ \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

These evaluation metrics are particularly valuable in applications where misclassification errors carry different consequences, enabling a more nuanced and reliable interpretation of model behavior.

The Used Data Set

In ML, models are developed by applying suitable training algorithms to labeled datasets. During the training phase, the model learns underlying patterns and relationships from the input data while adjusting its internal parameters accordingly. To assess the model's generalization capability, a separate testing dataset is used to evaluate how well the learned patterns apply to unseen data [19].

The dataset used for the study is Twitter Sentiment Dataset [9, 13, 42] which includes 162,980 entries typically contains tweets labeled with sentiment information, which can be positive, negative, or neutral. The length of tweet between 3-250 words. The dataset has been classified based on text lengths (see Figure1).

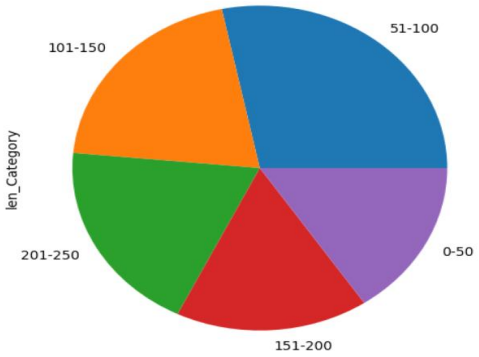

Figure 1: The dataset has been classified based on text lengths

Each of the entries represents the text of the tweet, every example in the dataset is labeled with one of three classes values. positive, negative and neutral.
The dataset was partitioned into two subsets: 75% allocated for training, allowing the model to learn underlying patterns, and 25% reserved for testing, enabling the evaluation of the model's ability to generalize and predict on previously unseen data.

Experiments

Pre-processing of the datasets

The dataset utilized in this study contains diverse user-generated opinions expressed in various linguistic forms. As the data is already labeled into three sentiment categories—positive, negative, and neutral—this facilitates structured sentiment analysis. Enhancing the quality of raw data through pre-processing is essential, as it directly influences model performance. This step typically involves removing redundant tokens, punctuation marks, and inconsistencies, thereby improving data clarity and processing efficiency. Pre-processing is especially important when working with unstructured and heterogeneous text streams. Consequently, applying a range of pre-processing techniques becomes necessary to extract meaningful patterns from large-scale textual data and convert it into a standardized and analyzable representation[2, 31].
In this phase, several preprocessing operations are applied to enhance the quality of the raw text before analysis. These include converting all text to lowercase, removing punctuation, hashtags, and URLs, and correcting common textual inconsistencies [18, 29,39] Following these steps, the cleaned text undergoes tokenization, where it is broken down into individual terms to facilitate further analysis. To enable machine learning models to process this textual input, the resulting tokens are subsequently converted into numerical formats through vectorization techniques [12].

Text Representation

Text representation is a important step in NLP, as it bridges the gap between human language and the computational methods used for analysis. because ML runs mathematical operations and algorithms, there are various methods to represent text data which convert preprocessed text into numerical features [18, 29]. These are the most important ways to represent the text:

Count Vectorizer: Represent each tweet as a vector of word counts, which produces a frequency count of all words present in the input text. By default, Count Vectorizer will tokenize input text into 1-grams known as unigrams. However, based on the dataset, it could be beneficial to incorporate more context and sequences of *n* depending on the value of *n* bigrams such as 2-grams(bigrams), 3-grams(trigrams) and 4-grams(quadgrams)[32, 38].

Term Frequency-Inverse Document Frequency (TF-IDF): The encoding of the input text into a numerical form such as vectors or matrices is an abbreviation for TF-IDF. This commonly used algorithm converts text into a meaningful representation of numbers which is used to fit ML algorithm for prediction [36].
Word Embeddings: Maps words to continuous vector spaces where similar words are close together for example Word2Vec and GloVe [14, 25]

Document Embeddings: It represents entire documents into vectors of fixed size in a continuous vector space. These embeddings capture the semantic meaning and context of the entire document for example Doc2Vec[26].

In our study, the used dataset has the text of the tweet which saved in categorical form. To represent it numerically we will employ Count Vectorizer, Count Vectorizer with (2-grams), Count Vectorizer with (3-grams) Count Vectorizer with (4-grams), TF-IDF, Doc2Vec,Word2Vec and GloVe.

TRAINING AND CLASSIFICATION

The experiments were carried out to perform classification tasks, subsequently, the results have been analyzed to illustrate the difference in the model's performance with various text representation by examining the influence of text length. However, LR, SVM, and NB will learn to perform a classification tasks. The block diagram of experiments is presented (see Figure 2)

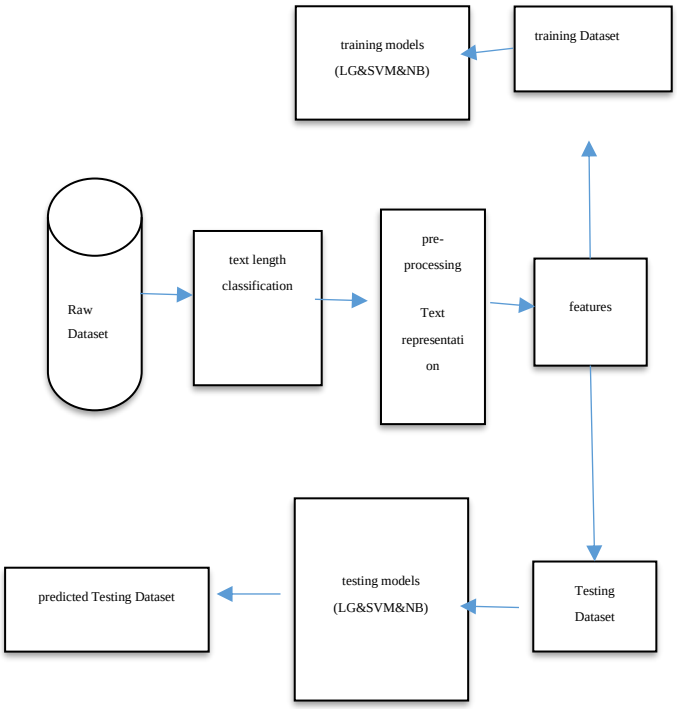

Figure 2: The block diagram of experiments

Results and discussion

To evaluate the models, a series of ten classification experiments were conducted, each aimed at identifying sentiment labels from textual data. This section presents the outcomes of these experiments along with concise analytical observations to illustrate and compare the models' performance.

Logistic Regression Model (LR)
First, the results of LR model with different text representation with examining the effect of text length were evaluated as presented in Table 2.

Table 2: The results of LR model with different text representation.

Len_Category	TfidfVectorizer	CountVectorizer	ngram_(1,2)	ngram_(1,3)	ngram_(1,4)	Word2Vec	Doc2Vec	GloVe
0-50	0.92	0.95	0.92	0.91	0.9	0.7	0.67	0.71
51-100	0.86	0.94	0.91	0.9	0.88	0.58	0.5	0.57
101-150	0.79	0.89	0.86	0.84	0.83	0.52	0.49	0.54
151-200	0.76	0.84	0.81	0.79	0.78	0.56	0.54	0.57
201-250	0.73	0.84	0.81	0.79	0.78	0.58	0.56	0.59

See Figure 3 for the performance of the LR model on short sentences (0–50 words) using the selected text representation techniques. The best performance was shown while using Count Vectorizer, closely following it TF-IDF and N-gram range (2,3,4), on the other hand, using Doc2Vec, Word2Vec and GloVe have shown less performance.

See Figure 4 for the performance of the LR model on sentences (51–100 words) using the selected text representation techniques. Count Vectorizer still providing the best performance, and as for N-gram range (2,3,4),

it continues its good performance, while TF-IDF shows decreasing in its performance. on the other hand, using Doc2Vec, Word2Vec and GloVe have shown the least performance, especially when using Doc2Vec for representation.

See Figure 5, Figure 6, Figure 7, we notice that the performance dropped in all of the experiments while extending the length of the sentence. Also, we notice that when using Doc2Vec, Word2Vec and GloVe have not shown as much of a decrease as the others, which indicates that they were not greatly affected by the length of the sentences.

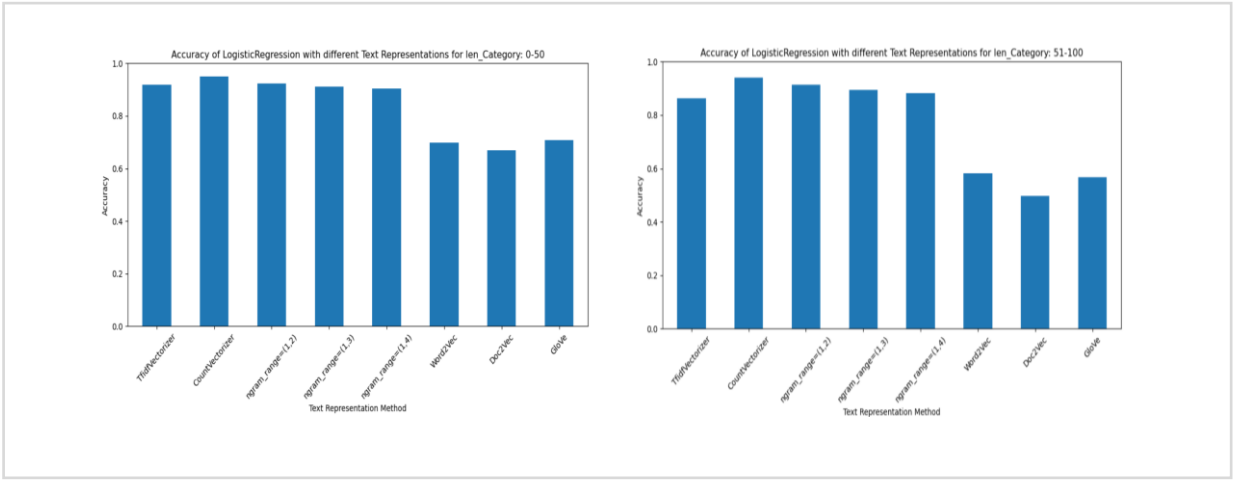

Figure 3: LR performance in the (0–50) range .

Figure 4: LR performance in the (51–100) range.

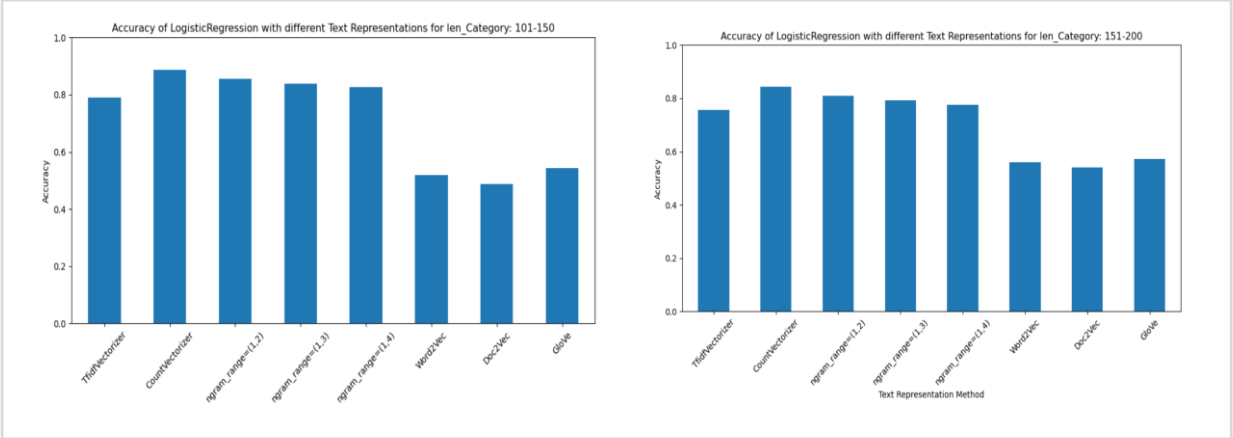

Figure 5: LR performance in the (101–150) range .

Figure 6: LR performance in the (151–200) range.

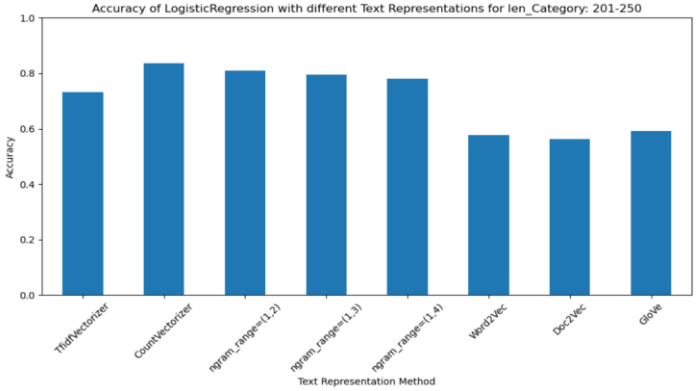

Figure 7: LR performance in the (201–250) range.

Support Vector Machine (SVM)

Second, the results of SVM model with different text representations with examining the effect of text length were evaluated as presented in Table 3.

Table: 3 The results of SVM model with different text representation

Len_Category	TfidfVectorizer	CountVectorizer	ngram_(1,2)	ngram_(1,3)	ngram_(1,4)	Word2Vec	Doc2Vec	GloVe
0-50	0.93	0.97	0.95	0.94	0.93	0.72	0.67	0.71
51-100	0.87	0.96	0.94	0.92	0.9	0.58	0.51	0.57
101-150	0.8	0.89	0.87	0.85	0.84	0.52	0.48	0.54
151-200	0.77	0.84	0.82	0.81	0.8	0.56	0.54	0.57
201-250	0.74	0.82	0.81	0.8	0.79	0.58	0.56	0.59

See Figure 8 for the performance of the SVM model on short sentences (0–50 words) using the selected text representation techniques. The best performance was shown while using Count Vectorizer, closely following it TF-IDF and N-gram range (2,3,4), on the other hand, using Doc2Vec, Word2Vec and GloVe have shown less performance.

See Figure 9 for the performance of the SVM model on sentences (51–100 words) using the selected text representation techniques. Count Vectorizer still providing the best performance, and as for N-gram range (2,3,4), it continues its good performance, while TF-IDF shows decreasing in its performance. on the other hand, using Doc2Vec, Word2Vec and GloVe have shown the least performance, especially when using Doc2Vec for representation.

See Figure 10, Figure 11, Figure 12, we notice that the performance dropped in all of the experiments while extending the length of the sentence. Also, we notice that when using Doc2Vec, Word2Vec and GloVe have not shown as much of a decrease as the others, which indicates that they were not greatly affected by the length of the sentences.

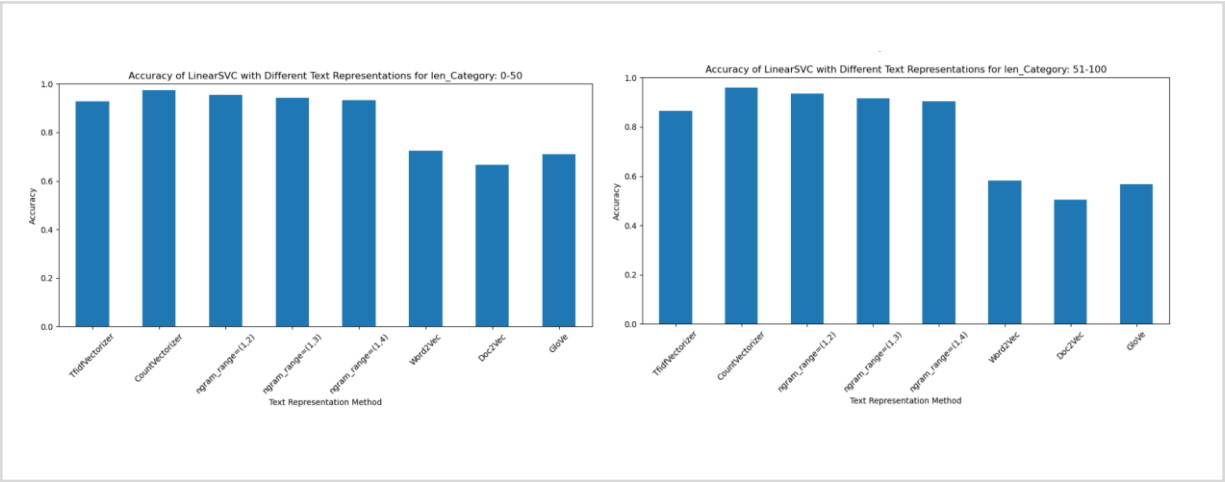

Figure 8: SVM performance in the (0–50) range.

Figure 9: SVM performance in the (51–100) range.

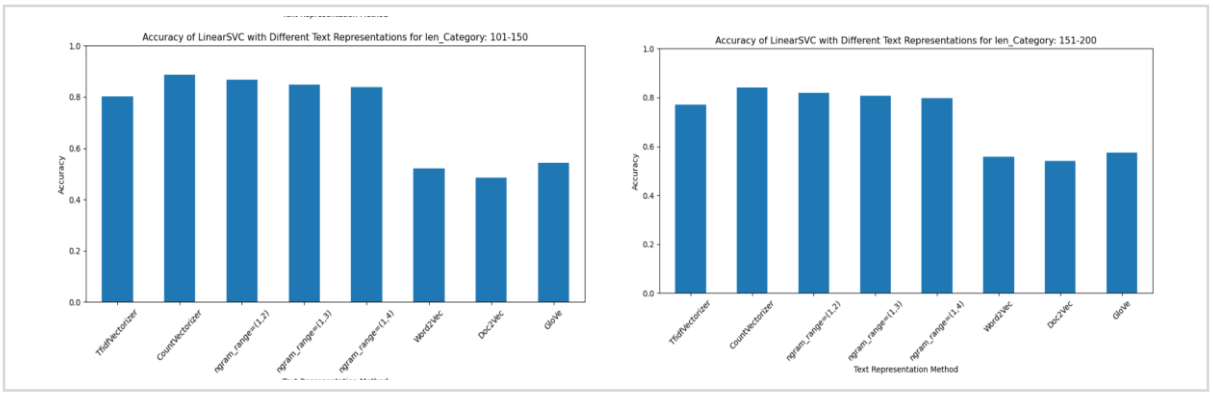

Figure 10: SVM performance in the (101–150) range

Figure 11: SVM performance in the (151–200) range

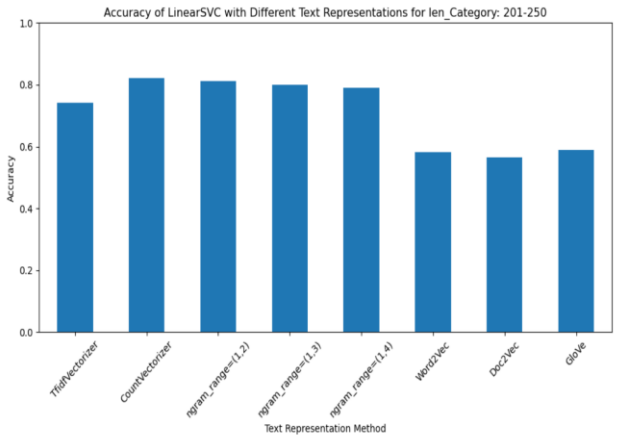

Figure 12: SVM performance in the (201–250) range.

Naive Bayes classifier

Third, the results of NB classifier model with different text representations examining the effect of text length were evaluated as presented in Table 4.

Table 4: The results of NB model with different text representation

Len_Category	TfidfVectorizer	CountVectorizer	ngram_(1,2)	ngram_(1,3)	ngram_(1,4)	Word2Vec	Doc2Vec	GloVe
0-50	0.90	0.91	0.88	0.87	0.86	0.64	0.64	0.64
51-100	0.82	0.84	0.80	0.78	0.78	0.52	0.46	0.52
101-150	0.64	0.72	0.62	0.58	0.57	0.48	0.48	0.49
151-200	0.62	0.68	0.61	0.59	0.59	0.54	0.54	0.54
201-250	0.62	0.68	0.61	0.61	0.61	0.57	0.57	0.57

See Figure 13 for the performance of the NB Model on short sentences (0–50 words) using the selected text representation techniques. The best performance was shown while using Count Vectorizer, closely following it TF-IDF and N-gram range (2,3,4), on the other hand, using Doc2Vec, Word2Vec and GloVe have shown less performance

See Figure 14 for the performance of the NB Model on sentences (51–100 words) using the selected text representation techniques. Count Vectorizer still providing the best performance, and as for N-gram range (2,3,4), it continues its good performance, while TF-IDF shows decreasing in its performance. on the other hand, using Doc2Vec, Word2Vec and GloVe have shown the least performance, especially when using Doc2Vec for representation.

See Figure 15, Figure 16, Figure 17, we notice that the performance dropped in all of the experiments while extending the length of the sentence. Also, we notice that when using Doc2Vec, Word2Vec and GloVe have not shown as much of a decrease as the others, which indicates that they were not greatly affected by the length of the sentences.

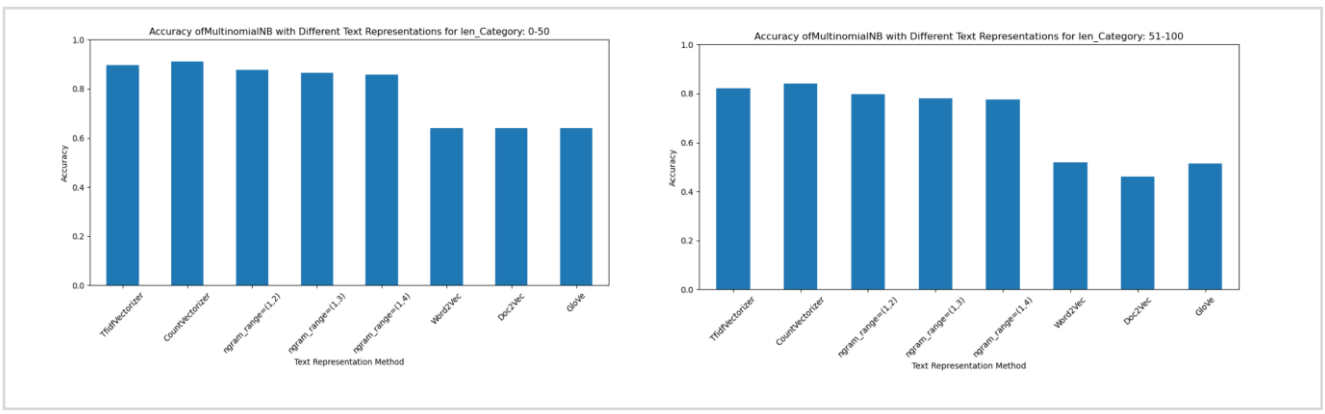

Figure 13: NB performance in the (0–50) range. **Figure 14:** NB performance in the (51–100) range.

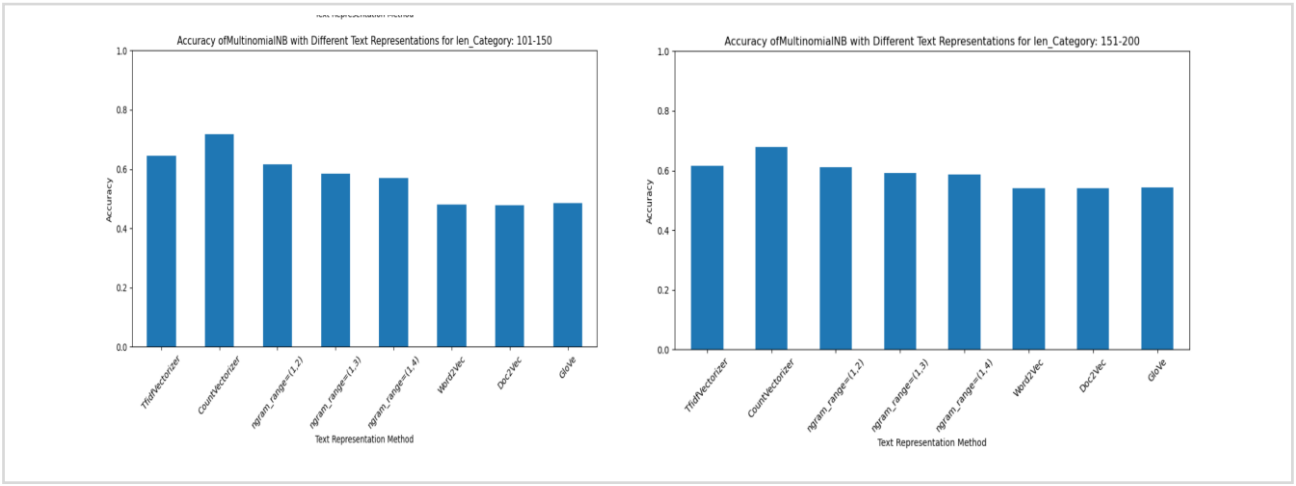

Figure 15: NB performance in the (101–150) range

Figure 16: NB performance in the (151–200) range

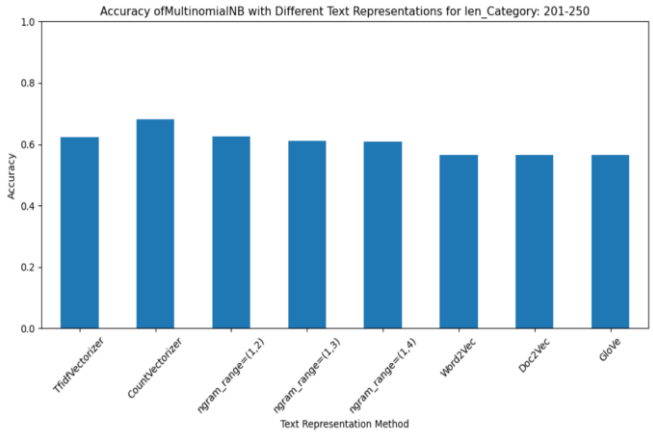

Figure 17: NB performance in the (201–250) range

In general, the results showed that the best performance for all models was when using Count Vectorizer to represent the text with N-gram range (1,2,3,4,) respectively, and then (TF-IDF) while Doc2Vec, Word2Vec and GloVe showed an average performance. For the text length, the model's performance decreases as the text length increases with all models. This suggests that the model clearly benefits from count-based text representations for short texts. It was also noted that (Doc2Vec, Word2Vec and GloVe.) kept the performance of the models despite the change in text length and generally gave average accuracy compared to (Count Vectorizer). However, throughout all experiments the SVM classifier had the best performance among all other techniques. In contrast, NB has shown the worst performance compared with LR Model and SVM. In addition, we observed that NB works better with short sentences, as the length of the sentence increases it showed a high drop in performance.

Conclusion

Our study explores the usage of eight text representation methods with three ML models evaluate on twitter sentiment classification Dataset. In particular, the experiments aim to test the impact of text length in the results classification. For this reason, the experiments were tested using a various range of text lengths. By comparing the accuracies of the models, we can conclude that Count Vectorizer gave the best performance for all models in all experiments. Additionally, for the length of text, the model's performance was inversely proportion to the text length with all text representation techniques. In summary, in order to make high-precision models the algorithms should understand better and learn the patterns in the dataset. Therefore, a good text representation method should be chosen.

References

[1] M. Abualhaj, A. A. Abu-Shareha, M. O. Hiari, Y. Alrabanah, M. Al-Zyoud, and M. A. Alsharaiah, "A paradigm for DoS attack disclosure using machine learning techniques," International Journal of Advanced Computer Science and Applications, 2022. [Online]. Available: search.proquest.com.

[2] M. M. Agüero-Torales et al., "A cloud-based tool for sentiment analysis in reviews about restaurants on TripAdvisor," Procedia Computer Science, vol. 162, pp. 392–399, 2019, doi: 10.1016/J.PROCS.2019.12.002.

[3] A. Alamsyah and D. M. Ginting, "Analyzing employee voice using real-time feedback," in 2018 4th International Conference on Science and Technology (ICST), 2018, doi: 10.1109/ICSTC.2018.8528569.

[4] N. Al-Bakri, J. F. Yonan, and A. T. Sadiq, "Tourism companies assessment via social media using sentiment analysis," Baghdad Science Journal, vol. 19, no. 2, 2022, doi: 10.21123/bsj.2022.19.2.0422.

[5] D. Antypas, "Twitter Topic Classification," 2022.

[6] R. Blanquero et al., "Variable selection for Naïve Bayes classification," Computers & Operations Research, vol. 135, p. 105456, 2021, doi: 10.1016/J.COR.2021.105456.

[7] Ü. Çavuşoğlu, "A new hybrid approach for intrusion detection using machine learning methods," Applied Intelligence, vol. 49, no. 7, pp. 2735–2761, 2019, doi: 10.1007/S10489-018-01408-X.

[8] H. Elaidi, Y. Elhaddar, Z. Benabbou, and H. Abbar, "An idea of a clustering algorithm using support vector machines based on binary decision tree," in 2018 International Conference on Intelligent Systems and Computer, 2018.

[9] H. Elaidi, Y. Elhaddar, Z. Benabbou, and H. Abbar, "An idea of a clustering algorithm using support vector machines based on binary decision tree," in 2018 International Conference on Intelligent Systems and Computer, 2018.

[10] W. A.-R. Finance, "Stock market reactions to domestic sentiment: Panel CS-ARDL evidence," Elsevier, 2020.

[11] O. Grljević et al., "Opinion mining in higher education: a corpus-based approach," Enterprise Information Systems, vol. 16, no. 5, 2022, doi: 10.1080/17517575.2020.1773542.

[12] N. Gulsoy and S. Kulluk, "A data mining application in credit scoring processes of small and medium enterprises commercial corporate customers," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 9, no. 3, 2019, doi: 10.1002/WIDM.1299.

[13] H. Elaidi, Y. Elhaddar, Z. Benabbou, and H. Abbar, "An idea of a clustering algorithm using support vector machines based on binary decision tree," in 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), 2018.

[14] E. Hindocha et al., "Short-text Semantic Similarity using GloVe word embedding," International Research Journal of Engineering and Technology, 2019.

[15] A. F. Ibrahim et al., "COVID19 Outbreak: A Hierarchical Framework for User Sentiment Analysis," 2021. doi: 10.32604/cmc.2021.xxxxxx.

[16] G. James et al., An Introduction to Statistical Learning with Applications in R, 2nd ed., 2021.

[17] K. Deulkar, M. Narvekar, P. Gandhi, D. Gada, and S. Kamath, "Evaluating the Influence of Text Representation techniques on Diverse Machine Learning Algorithms for Stress Detection in Social media users," in 2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE), 2024.

[18] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," 2019.

[19] T. Ma, K. Yamamori, and A. Thida, "A comparative approach to Naïve Bayes classifier and support vector machine for email spam classification," in 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), 2020.

[20] C. A. Martín et al., "Using Deep Learning to Predict Sentiments: Case Study in Tourism," Complexity, vol. 2018, p. 7408431, 2018, doi: 10.1155/2018/7408431.

[21] M. Munezero, C. S. Montero, E. Sutinen, and J. Pajunen, "Are they different? Affect, feeling, emotion, sentiment, and opinion detection in text," IEEE Transactions on Affective Computing, 2014.

[22] R. Nagamanjula and A. Pethalakshmi, "A novel framework based on bi-objective optimization and LAN2FIS for Twitter sentiment analysis," Social Network Analysis and Mining, vol. 10, no. 1, 2020, doi: 10.1007/S13278-020-00648-5.

[23] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," Social Network Analysis and Mining, vol. 11, no. 1, pp. 1–19, 2021, doi: 10.1007/S13278-021-00776-6.

[24] R. Patil et al., "A Survey of Text Representation and Embedding Techniques in NLP," IEEE Access, vol. 11, pp. 36120–36146, 2023, doi: 10.1109/ACCESS.2023.3266377.

[25] J. Pennington, R. Socher, and C. Manning, "GloVe: Global Vectors for Word Representation," 2014.

[26] L. Pfahler et al., "Learning low-Rank document embeddings with weighted nuclear norm regularization," in 2017 International Conference on Data Science and Advanced Analytics (DSAA), 2017, pp. 21–29, doi: 10.1109/DSAA.2017.46.

[27] D. Pisner et al., "Support vector machine," Elsevier.

[28] K. Priya, M. S. C. R. Kypa, M. M. S. Reddy, and G. R. M. Reddy, "A novel approach to predict diabetes by using Naive Bayes classifier," in 2020 4th International Conference on Trends in Electronics and Information Technologies (ICOEI), 2020.
[29] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," Journal of Machine Learning Research, vol. 21, pp. 1–67, 2020.
[30] M. M. Rahman and M. N. Islam, "Exploring the Performance of Ensemble Machine Learning Classifiers for Sentiment Analysis of COVID-19 Tweets," in Advances in Intelligent Systems and Computing, 2022, pp. 383–396, doi: 10.1007/978-981-16-5157-1_30.
[31] A. Rehman, A. K. Malik, B. Raza, and W. Ali, "A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis," Multimedia Tools and Applications, vol. 78, no. 18, pp. 26597–26613, 2019, doi: 10.1007/s11042-019-07788-7.
[32] S. Khan et al., "Fake News Classification using Machine Learning: Count Vectorizer and Support Vector Machine," Journal of Computing & Biomedical Informatics, vol. 4, no. 01, 2023, doi: 10.56979/401/2022/85.
[33] P. Siebers et al., "A Survey of Text Representation Methods and Their Genealogy," IEEE Access, vol. 10, pp. 96492–96513, 2022, doi: 10.1109/ACCESS.2022.3205719.
[34] R. Sonbol et al., "The Use of NLP-Based Text Representation Techniques to Support Requirement Engineering Tasks: A Systematic Mapping Review," IEEE Access, 2022, doi: 10.1109/ACCESS.2022.3182372.
[35] F. Song et al., "A comparative study on text representation schemes in text categorization," Pattern Analysis and Applications, vol. 8, pp. 199–209, 2005, doi: 10.1007/S10044-005-0256-3.
[36] M. I. Syafaah and L. Lestandy, "Emotional Text Classification Using TF-IDF and LSTM," 2022.
[37] J. S.-C. Systems, "Comparative analysis of text representation methods using classification," Taylor & Francis, 2014.
[38] T. Turki and S. S. Roy, "Novel Hate Speech Detection Using Word Cloud Visualization and Ensemble Learning Coupled with Count Vectorizer," Applied Sciences, vol. 12, no. 13, 2022, doi: 10.3390/app12136611.
[39] Q. Xu et al., "A systematic review of social media-based sentiment analysis: Emerging trends and challenges," Elsevier.
[40] S. Menard, Logistic Regression: From Introductory to Advanced Concepts and Applications. [Online]. Available: https://books.google.com.ly/books?hl=en&lr=&id=JSJzAwAAQBAJ&oi=fnd&pg=PP1&dq=21+Logistic+Regression+chapter+book+&ots=uarD41j9SV&sig=DG_LiCICowcqu6iQFyu6AGATVPY&redir_esc=y#v=onepage&q&f=false, last accessed Jul. 31, 2024.
[41] "Text Representation Techniques: A Unified Overview," in Language Intelligence, Wiley, pp. 305–306, 2025, doi: 10.1002/9781394297290.app1.
[42] Twitter Sentiment Dataset. [Online]. Available: kaggle.com.25. Pennington, J. et al.: GloVe: Global Vectors for Word Representation.