

Integrating SeqGAN and NLP Augmentation to Improve Generalization in Sentiment Classification Models

Shafin Mahmood¹, Sayma Akter Trina¹, Md.Easin Khondokar¹, Nuzhat Tabassum¹

¹Department of Computer Science, American International University-Bangladesh, Dhaka, Bangladesh

Article Info

Article history:

Received Feb 28, 2026

Revised Jun 22, 2026

Accepted Jul 19, 2026

Keywords:

Sentiment Analysis
GAN Augmentation
NLP Augmentation
Deep Learning
Machine Learning

ABSTRACT

Sentiment analysis is essential for understanding people's experiences and social media trends. However, challenges such as imbalanced datasets, limited data availability, and the informal nature of user-generated content often degrade a model's ability to achieve higher accuracy. This study presents that data augmentation using SeqGAN can effectively addresses these issues and improve prediction performance compared to traditional NLP-based augmentation techniques. A variety of Machine Learning (ML), Deep Learning (DL), hybrid, and GAN-based models were evaluated to predict sentiment from social media comments. After generating synthetic data using both SeqGAN and NLP-based techniques, all models were retrained, and notable improvements in accuracy were observed with the SeqGAN-augmented dataset. The use of synthetically generated data led to a statistically significant increase in classification performance, with a mean absolute accuracy improvement of approximately 4–5% across the full dataset. These findings highlight that GAN-based data augmentation, particularly through tailored synthetic sample generation, can enhance model robustness and boost sentiment prediction accuracy more effectively than conventional augmentation approaches.

Copyright © 2026 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Nuzhat Tabassum Department of Computer Science, American International University-Bangladesh
Dhaka, Bangladesh
Email: nuzhat.tabassum@aiub.edu

1. INTRODUCTION

Sentiment analysis is the process of analyzing emotions and opinions in text documents; it is also known as opinion mining. People frequently share and seek opinions on various online platforms. So, millions of opinions are available on the web. Opinion is a valuable insight that helps people understand different perspectives on any topic. The three main components of opinion mining are: *Opinion Holder*: The person or organization expressing the opinion; *Opinion Object*: The feature or entity being discussed; *Opinion Orientation*: The classification of the opinion as positive, negative, or neutral [1]. It is important to analyze these opinions automatically and build a model that can classify them as positive, negative, or neutral. Researchers have proposed various ML and DL models, such as Support Vector Machines (SVM) and Random Forest (RF), which have proven effective in sentiment classification tasks. Meanwhile, DL models, including Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), are used for automatic opinion classification [2, 3]. Traditional ML and DL models can perform well, capturing complex relationships to improve classification accuracy. Obtaining high-quality labeled datasets remains a major challenge. Additionally, language complexity, implicit sentiment, and expressive variation pose challenges for accurate text classification. Traditional data augmentation methods, such as synonym replacement, reverse translation, and word perturbation, can offer some improvement for sentiment classification [4], but fail to capture the diverse expressions and nu-

ances present in text. To address this issue, Generative Adversarial Networks (GANs) are a powerful approach used to generate synthetic data and create realistic synthetic samples. Traditional augmentation methods do not produce examples that are as high-quality and therefore may be less effective in improving the performance of ML models [5].

The main objective of this paper is to investigate how GANs can improve sentiment classification accuracy by generating synthetic data. It especially seeks to address the challenge of limited data and language complexity. By generating synthetic data using GAN, this paper finds better accuracy in predicting the classification. The rest of the paper is organized as follows: Section 2 provides a review of the various models used in sentiment analysis. Section 3 discusses the methodology of the study, focusing on how GANs are integrated into sentiment analysis to improve model performance by augmenting data. Section 4 provides the performance of those models for sentiment analysis, comparing models with and without the use of GAN-based data augmentation. Finally, Section 5 concludes by summarizing the key findings and discussing future work.

2. RELATED WORKS

Sentiment Analysis is one of the most significant research interest areas of Natural Language Processing (NLP), which is widely adopted across areas like social media monitoring, product review analysis, healthcare, finance, business intelligence, etc. Recent research has abandoned the use of traditional machine learning approaches for deep learning and transformer-based approaches to better capture contextual information and to boost the accuracy of the classification. Even with these achievements, it is still difficult to access high-quality labeled data, especially when the sample size is small, the data is imbalanced, or it comes from specialized areas. Due to this, data augmentation and generative text have become more and more popular for enhancing model robustness and generalization [6]. For sentiment analysis, ML and DL are both used to identify the category of sentiment (positive, negative, neutral). SVM can handle high-dimensional data and identify the best boundaries across classes, so it has become more popular for sentiment analysis. This was also used for spotting false information in jobs [7], particularly widely used on sites [8] such as Amazon. SVM was used to analyze sentiment in Twitter data and IMDb movie reviews in 2019 [9] with remarkable accuracy. Based on Bayes' theorem, the Naïve Bayes family was employed for text classification tasks such as sentiment analysis, spam filtering, article classification, and more [10]. It works in many NLP applications, especially when paired with feature selection and smoothing methods. Recently, many studies have proposed a hybrid model for sentiment analysis that combines Naïve Bayes, SVM, and DL models [11–13].

Transformer models have made great strides in sentiment classification, achieving better performance in learning contextual representations of words than traditional approaches. To overcome these challenges, Gong et al. introduced a lightweight transformer framework with text augmentation and knowledge distillation to enhance the performance of sentiment analysis under limited labeled data settings. With augmentation techniques, their approach showed competitive performance with a reduced computational cost, proving the effectiveness of using augmentation techniques with transformer architectures [14]. Likewise, transformer-based models and hybrid deep learning architectures have emerged as the current state-of-the-art for sentiment analysis, owing to their excellent contextual understanding and versatility in various application areas [15]. DistilBERT is more efficient for maintaining a large amount of language comprehension than BERT. It can accurately determine the text context. For sentiment analysis with emoji analysis to comprehend emotional dynamics in digital communication, DistilBERT was used. DistilBERT was also used to categorize public opinions during the COVID-19 epidemic [16–18]. The RoBERTa-LSTM model was used with GloVe data augmentation and achieved better accuracy [19]. A hybrid CNN-LSTM model achieved better results, and CNN and LSTM models achieved lower than hybrid in sentiment classification for two benchmark datasets. CNN was employed in their study to obtain higher-level features, while LSTM was employed for its ability to capture long-term dependencies between word sequences. The SVM technique is applied to the CNN-LSTM model to obtain better results than using an individual model for performing sentiment analysis [20].

On the other hand, researchers have explored the synthetic data generation technique to tackle the drawbacks of small and imbalanced datasets, in addition to enhancing classification models. SentiGAN and CatGAN were used to synthesize minority sentiment classes and showed that sentiment classification performance significantly improved when the minority classes were balanced [21]. They reported that the text generated by GAN can successfully alleviate the class imbalance problem and improve the reliability of the classifiers. SentiGAN and CatGAN can overcome those challenges and improve classification outcomes on

highly imbalanced educational datasets, as shown in the paper [22]. The study also showed, though, that it is difficult to achieve diversity and semantic consistency in the generation of synthetic text. To create realistic and sentiment-consistent synthetic data, Researchers have come up with a framework named SentiGEN, which integrates a modified T5 transformer with XLNet-based validation [23]. However, GANs are still an important part of the research field of high-quality text generation. To address this issue, Luo et al. suggested a framework that incorporates sentence compression and data screening techniques to produce quality synthetic sentences while retaining the sentiment information. Their approach proved to be very effective in solving the challenges of generating long texts and resulted in higher classification accuracy on several sentiment datasets, which makes SeqGAN one of the most promising approaches for augmenting sentences for sentiment [24]. Conditional GAN (cGAN) [22] is suitable to enhance sentiment classification accuracy in low-resource contexts. Studies suggested another GAN approach to augment the data for low-resource sentiment databases and boost the performance of textual classification. They also conducted data quality analysis by t-SNE visualization.

Several works have investigated traditional text enhancements to achieve model generalization as well. Synonym replacement, random insertion, random deletion, and random swapping have been applied to the dataset successfully in Dynamic Easy Data Augmentation (EDA) to enhance the dataset's feature diversity and reduce the overfitting effect when using deep convolutional networks. Experimental results showed that significant enhancement in terms of F1-score and robustness in classification can be achieved, especially in the case of a limited number of training samples [25]. For NLP applications like sentiment analysis, their survey noted that good synthetic text is increasingly crucial as it directly affects the results of the models trained on it [26]. Furthermore, researchers have explored the problem of text augmentation for GAN from a more general standpoint. In their systematic review about text augmentation with GANs, Silva et al. found that adversarial learning can significantly enhance text classification tasks by creating a variety of training instances [27]. Recently, augmentation using GAN has been successfully adapted for low-resource languages. Mohammadi and colleagues created a Persian sentiment analysis synthetic data generation framework using GANs and found significant improvements in classification accuracy when using synthetic data as an augmentation of the original data [28].

Previous works have shown positive results with the use of transformer-based models, NLP-based augmentation, and GAN-based synthetic text generation, there are still some technical challenges to be overcome. The majority of the current literature tests only a few classifiers, and it is not possible to assess whether a specific augmentation strategy is valid for a given learning paradigm. In addition, the previous studies are mainly focused on either the conventional NLP augmentation or the GAN augmentation separately, without a systematic comparison under the same experimental setups. Furthermore, the influence of various augmentation techniques on model robustness and generalization capability is not well studied across the different types of classifiers (machine learning, deep learning, hybrid, and generative adversarial network (GAN)), but only limited attention has been paid to how the augmentation methods impact these models. To overcome these shortcomings, this study compares and evaluates a large number of machine learning, deep learning, hybrid, and GAN-based sentiment classification models in a uniform experimental setup with original, NLP-enhanced, and generated data sets. The proposed evaluation allows us to gain a deeper insight into the impact of the synthetic text generated by SeqGAN on feature learning, model generalization, and the performance of the sentiment classification model in comparison to traditional NLP-based augmentation methods.

3. METHODOLOGY

The steps of our methodology are depicted in Figure 1. The steps begin with collecting the datasets. After that, we clean and preprocess the data and expand it using data augmentation. Later, we apply several ML, DL, and hybrid models on both augmented and non-augmented data. Finally, the work procedure ends with evaluating the results to assess the models' comparative performance.

3.1. Data Description

Twitter and Facebook were particularly significant tools for analysis because they support high-scale user participation and time-sensitive interactions. The experiments were conducted using the Sentiment140 dataset, which is publicly available on Kaggle: <https://www.kaggle.com/datasets/kazanova/sentiment140> [29]. The original dataset consists of 1,048,576 labeled instances belonging to two sentiment classes: positive and negative. A subset of 149,900 comments was selected from the Kaggle dataset for this experiment. Among these, 74,950 comments were labeled as negative (class label as 1) and 74,950 were labeled as

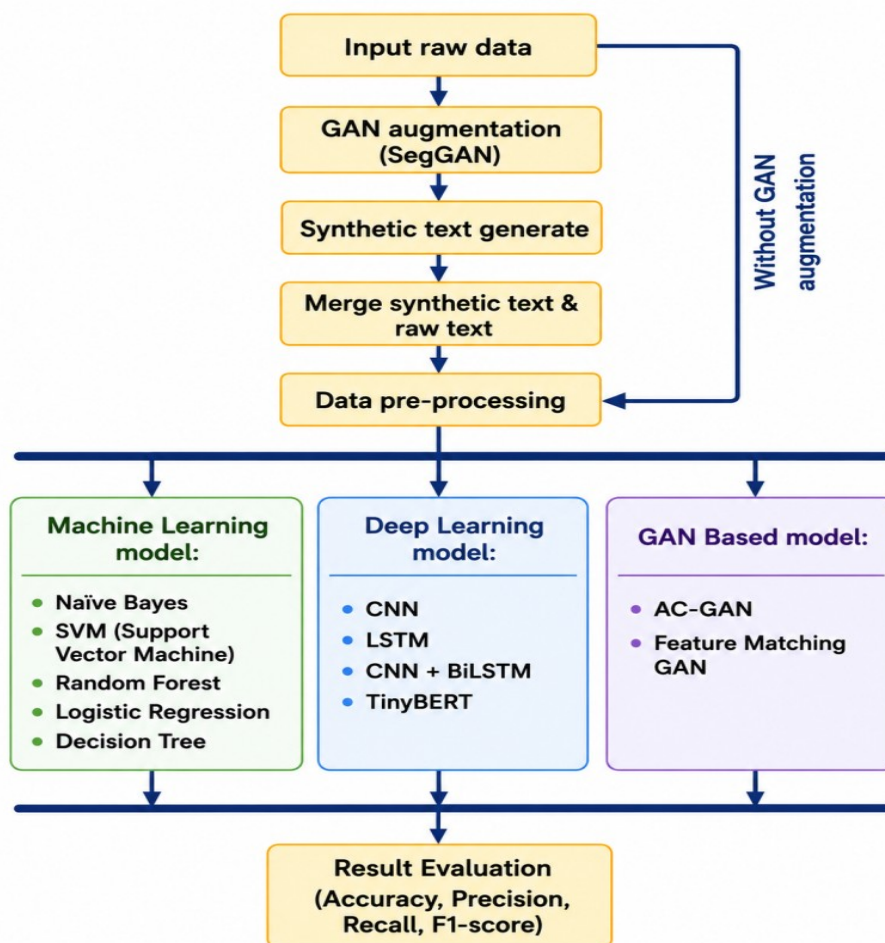


Figure 1. Experimental Workflow

positive (class label as 0). The selection process was random per category. This subset was used to reduce computational cost, maintain a balanced class distribution, and assess the performance of the models on a comparatively smaller dataset. Our experimental dataset is also available on Kaggle <https://www.kaggle.com/datasets/shafinmahmood498465/sentiment-binary-research-dataset?resource=download>

Since the dataset was already labeled, no manual annotation was performed from scratch. To verify the label quality, we first used VADER to check the consistency of the existing sentiment labels. Then, three annotators manually reviewed the sentiment assignments to verify the existing labels, focusing on finding ambiguous or inconsistent cases. Any disagreements were resolved through discussion until a consensus was reached.

3.2. Data Preprocessing

After finishing the labeling process, we moved on to data preprocessing. Since the comments were collected directly from Twitter, they were mostly unstructured and written in informal language. First, we converted all text to lowercase to keep everything consistent. Then, we expanded short forms like "isn't" to "is not" to make the meaning clearer. We removed unnecessary elements such as URLs, emails, hashtags, mentions, and non-ASCII characters. To reduce noise, we filtered out common stop words. We then applied part-of-speech (POS) tagging to identify the grammatical role of each word and used that information to perform lemmatization, converting words to their base forms based on their POS tags. Finally, we performed POS-based filtering and additional feature engineering to enhance the data. This cleaned, balanced dataset was then used for sentiment analysis.

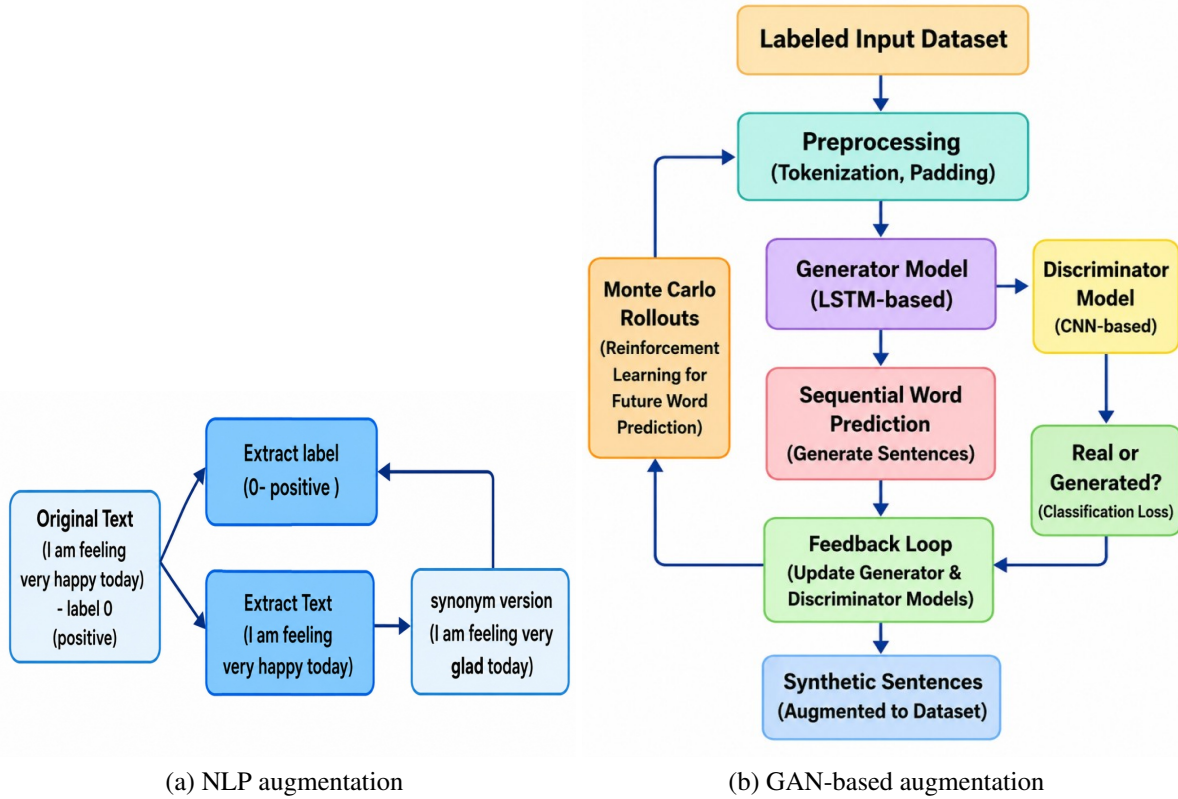


Figure 2. Augmentation procedure

3.3. Data Augmentation Methods

Our main goal is to see how augmentation improves classification results. So, in this stage of the experiment, we created synthetic data using two different approaches: NLP-based and GAN-based augmentation to improve data quality.

3.3.1. NLP-based Data augmentation

Figure 2a illustrates our applied synonym-based augmentation technique for generating new text. We generated two augmented versions of each input. Each row represents an input sample, and for augmentation, we extracted the original text and label. This procedure generated a synonym-based augmentation of the same sentence. If augmentation failed for very short sentences, it skipped that sentence and moved to the next sentence. After processing all the data, it randomly shuffled the augmented and original sentences to reduce bias. The pseudo-algorithm 1 represents the stepwise procedure of NLP-based augmentation. Let the original labeled dataset be defined as: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x_i = \{w_1, w_2, \dots, w_T\}$ denotes a tokenized text sequence and y_i represents its sentiment label. NLP-based augmentation generates new samples by applying synonym substitution while preserving semantic meaning. The augmented sentence is expressed as: $x'_i = f_{\text{syn}}(x_i)$, where $f_{\text{syn}}(\cdot)$ replaces selected tokens with semantically equivalent synonyms. The sentiment label remains unchanged: $y'_i = y_i$. The augmented dataset is defined as: $\mathcal{D}_{\text{NLP}} = \{(x'_i, y_i)\}_{i=1}^N$ and the final training set becomes: $\mathcal{D}_{\text{final}} = \mathcal{D} \cup \mathcal{D}_{\text{NLP}}$. This formulation increases lexical diversity while maintaining label consistency, providing moderate improvements in model generalization.

Input: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ (dataset)
 K (augmentations per sample)
 $\rho \in (0, 1]$ (replacement ratio)
 $\text{Syn}(\cdot)$ (synonym lexicon), $\text{Tok}(\cdot)$
Output: $\mathcal{D}_{NLP}$ (augmented dataset)

```

1 for  $i = 1$  to  $N$  do
2    $[w_1, \dots, w_T] \leftarrow \text{Tok}(x_i)$ ;
3    $M \leftarrow \max(1, \lfloor \rho T \rfloor)$ ;
4   for  $j = 1$  to  $K$  do
5      $S \leftarrow \emptyset$ ;
6     Select  $I \subseteq \{1, \dots, T\}, |I| = M$ ;
7     foreach  $t \in I$  do
8       if  $\text{Syn}(w_t) \neq \emptyset$  then
9          $w'_t \sim \text{Syn}(w_t)$ ;
10         $S \leftarrow S \cup \{(t, w'_t)\}$ ;
11      end
12    end
13    for  $t = 1$  to  $T$  do
14      if  $(t, w'_t) \in S$  then
15         $w_t^{(j)} \leftarrow w'_t$ ;
16      else
17         $w_t^{(j)} \leftarrow w_t$ ;
18      end
19    end
20     $x_i^{(j)} \leftarrow \text{Detok}(w^{(j)})$ ;
21     $\mathcal{D}_{NLP} \leftarrow \mathcal{D}_{NLP} \cup \{(x_i^{(j)}, y_i)\}$ ;
22  end
23 end

```

Algorithm 1: NLP-based data augmentation (synonym substitution)

3.3.2. GAN-based Data augmentation

To address data imbalance and scarcity in our dataset, we applied synthetic data generation using SeqGAN, a technique well-suited for enhancing diversity in sentiment analysis tasks (Figure 2b). In GAN architecture, there are 2 components: one is the generator, which is built with an LSTM network and learns to produce realistic sentences by capturing sequential dependencies in text. It then sequentially predicts the next word conditioned on previous words. The other is the discriminator, which utilizes a CNN-based classifier and evaluates whether a sentence is real or generated. Based on the discriminator's feedback, the generator improves over time. In order to improve the generator and predict future words, Monte Carlo roll outs are used in conjunction with reinforcement learning. As a result, SeqGAN gradually produces more realistic and varied text samples to enrich the dataset. Pseudo Algorithm 2 depicts the detailed procedure of this method. SeqGAN models text generation as a sequential decision process. Given a sequence $X = (x_1, x_2, \dots, x_T)$, the generator G_θ defines the probability of generating a full sentence as: $G_\theta(X) = \prod_{t=1}^T G_\theta(x_t | x_{1:t-1})$. Since text generation is non-differentiable, the generator is optimized using reinforcement learning by maximizing the expected reward: $J(\theta) = \mathbb{E}_{X \sim G_\theta}[R(X)]$, where the reward $R(X)$ is provided by the discriminator D_ϕ . The policy gradient update is given by: $\nabla_\theta J(\theta) \approx \sum_{t=1}^T \mathbb{E}[Q_{D_\phi}(x_{1:t-1}, x_t) \nabla_\theta \log G_\theta(x_t | x_{1:t-1})]$. The discriminator is trained to distinguish real and generated samples by minimizing the loss: $\mathcal{L}_D = -\mathbb{E}_{X \sim \mathcal{D}}[\log D_\phi(X)] - \mathbb{E}_{X \sim G_\theta}[\log(1 - D_\phi(X))]$. After training, the generator produces synthetic sequences X_{syn} , which are merged with the original dataset: $\mathcal{D}_{\text{GAN}} = \mathcal{D} \cup \{(X_{\text{syn}}, y_{\text{syn}})\}$. This adversarial formulation enables SeqGAN to generate diverse and contextually coherent synthetic text, significantly enhancing data richness and improving sentiment classification performance.

Input: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ (real dataset)
 G_θ (generator, LSTM), D_ϕ (discriminator, CNN)
 T (max sequence length), M (MC rollouts)
 α, β (learning rates)
 E_g, E_d (update epochs)
 N_{syn} (synthetic samples)

Output: $\mathcal{D}_{GAN}$ (augmented dataset)

1 Pre-train Generator (MLE):

2 Minimize

$$\mathcal{L}_{MLE}(\theta) = - \sum_{(x,y) \in \mathcal{D}} \sum_{t=1}^T \log G_\theta(x_t | x_{1:t-1})$$

3 Pre-train Discriminator:

$$\mathcal{L}_D(\phi) = -\mathbb{E}_{X \sim \mathcal{D}}[\log D_\phi(X)] - \mathbb{E}_{X \sim G_\theta}[\log(1 - D_\phi(X))]$$

4 **for** $e = 1$ **to** E_g **do**

5 Sample sequences $X \sim G_\theta$;

6 $Q \leftarrow 0$;

7 **for** $t = 1$ **to** T **do**

8 $s_t \leftarrow x_{1:t}$;

9 **for** $m = 1$ **to** M **do**

10 $\hat{X}^{(m)} \sim G_\theta(\cdot | s_t)$;

11 $r^{(m)} \leftarrow D_\phi(\hat{X}^{(m)})$;

12 **end**

13 $Q_{D_\phi}(s_t, x_t) = \frac{1}{M} \sum_{m=1}^M r^{(m)}$;

14 $g_t = Q_{D_\phi}(s_t, x_t) \nabla_\theta \log G_\theta(x_t | x_{1:t-1})$;

15 $Q \leftarrow Q + g_t$;

16 **end**

17 $\theta \leftarrow \theta + \alpha Q$;

18 **for** $d = 1$ **to** E_d **do**

19 $X_{\text{real}} \sim \mathcal{D}$;

20 $X_{\text{fake}} \sim G_\theta$;

21 $\phi \leftarrow \phi - \beta \nabla_\phi \mathcal{L}_D(\phi)$;

22 **end**

23 **end**

24 **for** $n = 1$ **to** N_{syn} **do**

25 $X_{\text{syn}}^{(n)} \sim G_\theta$;

26 Assign $y_{\text{syn}}^{(n)}$;

27 $\mathcal{D}_{\text{syn}} \leftarrow \mathcal{D}_{\text{syn}} \cup \{(X_{\text{syn}}^{(n)}, y_{\text{syn}}^{(n)})\}$;

28 **end**

29 $\mathcal{D}_{GAN} = \mathcal{D} \cup \mathcal{D}_{\text{syn}}$;

Algorithm 2: GAN-based data augmentation (SeqGAN)

Following data augmentation, the size of the experimental dataset increased to 244,727 comments in total. Figure 3 illustrates before and after augmentation word-cloud for positive word in our dataset. Similarly, figure 4 represents the word cloud for negative sentiment texts before and after augmentation, respectively. Furthermore, Figure 5 and Figure 6 represents the top 20 most frequent positive and negative words in the dataset before and after data augmentation

3.4. Machine Learning, Deep Learning and GAN Models

To measure the effectiveness of the proposed data augmentation scheme based on SeqGAN, multiple machine learning, deep learning, hybrid, and GAN-based models were implemented for sentiment classifica-

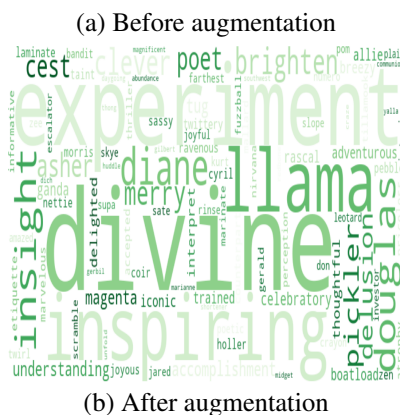


Figure 3. Word cloud for positive sentiment

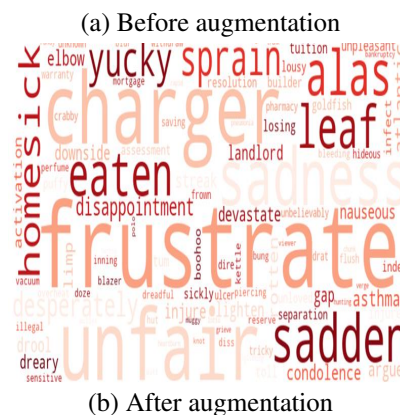
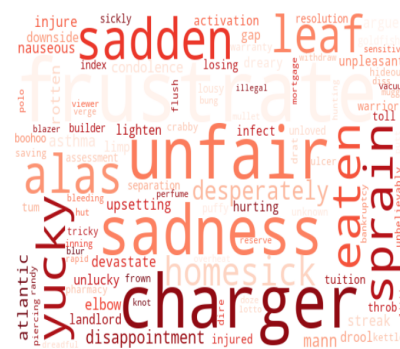


Figure 4. Word cloud for negative sentiment

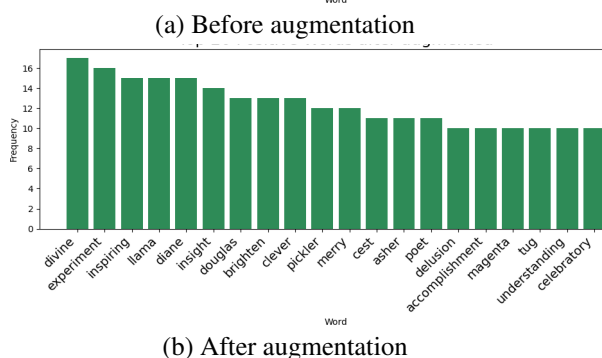
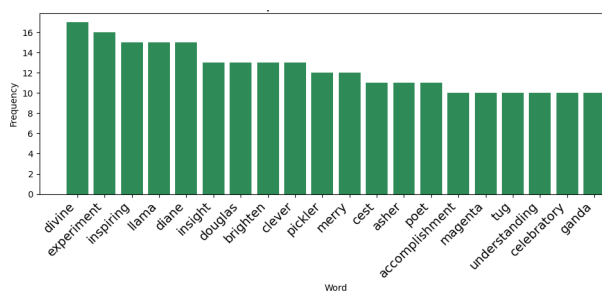


Figure 5. Top 20 frequent positive words for the dataset

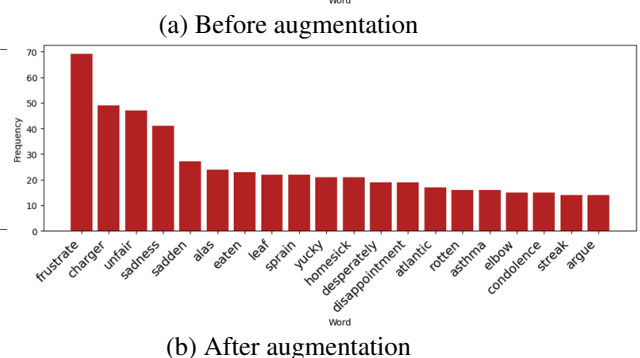
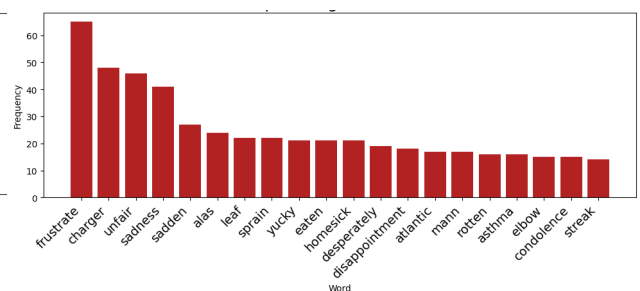


Figure 6. Top 20 frequent negative words for the dataset

tion. The models have been chosen to depict various learning paradigms and are prevailing benchmark models for text classification. By fixing the training and testing sets as they were originally used, a fair comparison of the performance of the models on the original dataset, the NLP-augmented dataset, and the SeqGAN-augmented dataset was made, which allowed for a thorough analysis of the effect of the various augmentation

methods.

3.4.1. ML Models

Five common machine learning (ML) algorithms have been used for sentiment classification to establish a good baseline: Support Vector Machine (SVM), Random Forest (RF), Naïve Bayes (NB), Decision Tree (DT), and Logistic Regression (LR). SVM is well suited for high-dimensional text classification problems, since it can determine an optimum decision surface between the sentiment classes [30, 31]. RF was chosen as it is robust, has the capacity to mitigate overfitting, and demonstrates robust ensemble learning classification accuracy. Feature vectors were generated using CountVectorizer, and the model was trained using 100 decision trees, with handling of missing values handled by removing them or filling them with blanks. TF-IDF features were used to build a hierarchical decision rules-based model with a DT, which offers an interpretable classification model [32]. The implementation of multinomial Naïve Bayes was chosen due to its computational efficiency and its effectiveness in the case of sparse text data [33]. For the baseline linear classification, Logistic Regression [34] was used, and TF-IDF features were learned for a linear decision boundary for binary sentiment prediction.

3.4.2. DL Models

Various deep learning models, such as CNN, Bi-LSTM, TinyBERT, 128-dimensional, and a hybrid CNN-BiLSTM model, were implemented to assess the efficiency of this approach. We used a 128-dimensional embedding layer, a 1-dimensional CNN with L2 regularization, global max-pooling, a dropout layer with a dropout rate of 0.3, and a dense layer of 64 neurons before the final sigmoid output layer in our CNN [35]. Adam optimizer and binary cross-entropy loss were used to optimize the model. In the sequential feature learning, a Bi-LSTM architecture was used, as it is capable of modeling long-range contextual dependencies in both directions [36, 37]. Two stacked Bi-LSTM layers with 256 and 128 hidden units were followed by dropout layers (0.5), L2 regularization, a dense layer with 128 ReLU neurons, and an output sigmoid layer with a learning rate of 0.001. To compare the contextual language representations with less computational complexity than conventional BERT [38], TinyBERT was added as a lightweight transformer-based model. This model was then optimized using the Hugging Face Trainer API with a 128-dimensional maximum sequence length, batch size, and learning rate set to 128, 128, and (2×10^{-5}) , respectively. Lastly, a CNN-Bi-LSTM model was proposed, which leverages the CNN's potential to capture local n-gram features and the Bi-LSTM's capacity in learning long-range sequential dependencies [39, 40]. Architecture: 128-dimensional embedding layer, SpatialDropout1D (0.2), Conv1D (64, 5), Bi-LSTM and dense (ReLU) layers before binary classification.

3.4.3. GAN-Based Models:

Two GAN-based classifiers, namely, Auxiliary Classifier GAN (AC-GAN) and Feature Matching GAN, were implemented to explore the usefulness of adversarial learning for sentiment classification, in addition to conventional ML and DL models. In AC-GAN, an auxiliary classifier is introduced to predict authenticity and sentiment class for each sample [41]. The model was trained with adversarial learning with the binary cross-entropy loss function and class prediction with the categorical cross-entropy loss function, both with a learning rate of 0.0002. To achieve training stability and optimization, the generator used feature matching rather than the maximum confusion in the discriminator; the Feature Matching GAN was also implemented [42]. The discriminator was trained to predict the authenticity of the samples and their sentiment labels, and the generator was trained to generate samples with representations that were close to those generated by the discriminator. To this end, this strategy aimed to generate more representative synthetic samples and offers an extra comparison to the proposed augmentation framework based on SeqGAN.

3.5. Experimental Setup

To provide an equal and fair assessment, all the experiments were performed with the same training and testing conditions for all the models. A well-cleaned, balanced dataset was split into 80% train and 20% test data, with a fixed random state of 42 for the split. The original dataset, the NLP-augmented dataset, and the SeqGAN-augmented dataset were tested. The features used for traditional machine learning models included TF-IDF or CountVectorizer features, while the features used for deep learning and transformer-based models were tokenized text sequences. The hyperparameters were chosen by an initial trial to ensure stable convergence, and similar parameters were used throughout all models. All the hyperparameters used for each model are summarized in Table 1.

Table 1. Hyperparameter settings of experimental models

Parameter	SVM	Logistic Re- gres- sion	Random Forest	Naïve Bayes	Decision Tree	CNN	Bi- LSTM	TinyBERT	CNN- BiLSTM	AC-GAN	Feature Match- ing GAN
Test size	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
Random state	42	42	42	42	42	42	42	42	42	42	42
Input repre- sentation	TF- IDF	TF-IDF	Count Vector- izer	TF-IDF	TF-IDF	Embedding	Embedding	Tokenizer	Embedding	TF-IDF	TF-IDF
Max fea- tures	5000	–	–	5000	–	–	–	–	–	–	–
Kernel	Linear	–	–	–	–	–	–	–	–	–	–
Estimators	–	–	100	–	–	–	–	–	–	–	–
Embedding dimension	–	–	–	–	–	128	128	–	128	–	–
Hidden lay- ers / Units	–	–	–	–	–	Conv1D (128 filters)	256 → 128	–	Conv1D + BiL- STM (128)	Generator / Discrim- inator	Generator / Discrim- inator
Kernel size	–	–	–	–	–	5	–	–	5	–	–
Batch size	–	–	–	–	–	64	128	128	64	64	64
Epochs	–	–	–	–	–	50	10	3	15	30	30
Learning rate	–	–	–	–	–	0.001	0.001	2×10^{-5}	0.001	0.0002	0.0002
Activation	–	–	–	–	–	ReLU	ReLU	–	ReLU / Sigmoid	ReLU / Tanh	LeakyReLU
Dropout	–	–	–	–	–	0.3	0.5	–	0.2 / 0.6	–	–
Loss func- tion	–	Binary	–	–	–	Binary Cross- Entropy	Binary Cross- Entropy	Cross- Entropy	Binary Cross- Entropy	Binary Cross- Entropy + Cate- gorical Cross- Entropy	Feature Matching + Binary Cross- Entropy
Optimizer	–	–	–	–	–	Adam	Adam	AdamW	Adam	Adam	Adam

4. RESULT AND DISCUSSION

The model efficacy was evaluated using standard metrics that included Accuracy, Precision, Recall, F1-score, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Matthews Correlation Coefficient (MCC). Each of these metrics was applied consistently to all models and experimental settings to allow direct performance comparison. In the paper [43], the formula for Accuracy, Precision, F1-score, and Recall, including both true positive and true negative values (Table 2). Let α , β , γ , and δ represent True Positive, False Positive, False Negative, and True Negative, respectively.

Table 2. Evaluation metrics

Metric	Formula	Metric	Formula
Precision (P)	$\frac{\alpha}{\alpha + \beta}$	MAE	$\frac{1}{N} \sum_{i=1}^N y_i - \hat{y}_i $
Recall (R)	$\frac{\alpha}{\alpha + \gamma}$	MSE	$\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$
F1 Score	$\frac{2 \cdot (P \cdot R)}{P + R}$	MCC	$\frac{(\alpha \cdot \delta) - (\beta \cdot \gamma)}{\sqrt{(\alpha + \beta)(\alpha + \gamma)(\delta + \beta)(\delta + \gamma)}}$
Accuracy	$\frac{\alpha + \delta}{\alpha + \beta + \gamma + \delta}$		

4.1. Experiment Result Analysis

Table 3 presents the classification accuracy of the evaluated models on the original dataset as well as on the datasets augmented using NLP and GAN-based techniques. It can be observed that SeqGAN enhancement across the board improved classification performance over both the standard dataset and NLP-based enhancement. Classic classifiers (SVM, Naïve Bayes, Decision Tree) experienced moderate gains, with about a 2 percent increase, whereas DL classifiers showed notable increments: CNN improved from 0.72 to 0.84, and

Table 3. Performance comparison (accuracy) of the evaluated models under different data augmentation strategies

Algorithm	Normal Dataset	NLP Augment	GAN Augment
Logistic Regression	0.78	0.80	0.80
RF	0.85	0.87	0.86
SVM	0.77	0.78	0.80
Naïve Bayes	0.72	0.73	0.79
Decision Tree	0.82	0.85	0.83
CNN	0.72	0.73	0.84
LSTM	0.78	0.82	0.82
CNN + BiLSTM	0.78	0.80	0.82
AC-GAN	0.73	0.73	0.77
Feature Matching GAN	0.75	0.75	0.79
TinyBERT	0.77	0.79	0.80

LSTM from 0.78 to 0.82. RF obtained the very best performance overall (0.86) with SeqGAN. This validates that adversarial enhancement allows for enhanced generalizability, especially for deep and hybrid architectures.

Overall, balanced improvement across evaluation scores is presented in Table 4. SeqGAN sampling significantly improved recall and F1-scores for sequential schemes: recall for CNN+BiLSTM increased from 0.61 to 0.84, and for LSTM, from 0.76 to 0.85. While NLP-based enhancement at times led to high precision (e.g., CNN 0.86, RF 0.88), SeqGAN consistently outperformed in recall and F1 performance, which showed enhanced class sensitivity and robustness towards data imbalance.

Table 4. Comparison of precision, recall, and F1-score for all evaluated models under different data augmentation strategies

Algorithm	Precision			Recall			F1-score		
	Normal	GAN	NLP	Normal	GAN	NLP	Normal	GAN	NLP
Logistic Regression	0.79	0.76	0.82	0.76	0.75	0.77	0.77	0.76	0.79
RF	0.88	0.89	0.88	0.80	0.81	0.84	0.84	0.85	0.86
SVM	0.79	0.81	0.79	0.74	0.70	0.75	0.76	0.78	0.77
Naïve Bayes	0.71	0.80	0.71	0.77	0.76	0.75	0.74	0.78	0.73
Decision Tree	0.86	0.88	0.88	0.76	0.77	0.81	0.81	0.82	0.85
CNN	0.71	0.84	0.86	0.70	0.81	0.82	0.71	0.83	0.84
LSTM	0.78	0.79	0.80	0.76	0.85	0.84	0.77	0.83	0.82
CNN + BiLSTM	0.76	0.81	0.82	0.61	0.84	0.78	0.68	0.82	0.80
AC-GAN	0.74	0.79	0.75	0.69	0.73	0.70	0.72	0.76	0.73
Feature Matching GAN	0.76	0.81	0.77	0.72	0.75	0.72	0.74	0.78	0.74
TinyBERT	0.77	0.79	0.78	0.77	0.82	0.81	0.77	0.80	0.79

Table 5. Comparison of MAE, MSE, and MCC for experimental models using Normal, GAN-augmented, and NLP-augmented datasets.

Algorithm	MAE			MSE			MCC		
	Normal	GAN	NLP	Normal	GAN	NLP	Normal	GAN	NLP
Logistic Regression	0.21	0.24	0.19	0.21	0.24	0.19	0.56	0.59	0.60
RF	0.15	0.14	0.13	0.15	0.14	0.13	0.70	0.71	0.73
SVM	0.22	0.20	0.21	0.22	0.20	0.21	0.55	0.59	0.56
Naïve Bayes	0.27	0.20	0.27	0.27	0.20	0.20	0.44	0.58	0.45
Decision Tree	0.17	0.16	0.14	0.17	0.16	0.14	0.65	0.67	0.71
CNN	0.28	0.16	0.15	0.28	0.16	0.15	0.42	0.67	0.69
LSTM	0.22	0.17	0.18	0.22	0.17	0.18	0.55	0.64	0.63
CNN + BiLSTM	0.28	0.18	0.19	0.28	0.18	0.19	0.43	0.64	0.61
AC-GAN	0.26	0.22	0.26	0.26	0.22	0.26	0.46	0.54	0.47
Feature Matching GAN	0.24	0.20	0.24	0.24	0.20	0.24	0.50	0.58	0.51
TinyBERT	0.22	0.19	0.20	0.22	0.19	0.20	0.54	0.60	0.59

Table 5 compares the error metrics (MAE and MSE) and the Matthews correlation coefficient (MCC) of different models to assess the influence of data augmentation on classification performance. It revealed that SeqGAN augmentation diminished MAE and MSE for the majority of the models, yet increased Matthews Correlation Coefficient (MCC). The DL models benefited the most: CNN+BiLSTM MAE diminished from 0.28 to 0.18 and MCC from 0.43 to 0.64; MCC for LSTM increased from 0.55 to 0.64. RF using SeqGAN provided the strongest overall performance (MAE = 0.14, MCC = 0.71), which validates its reliability as the top performing classifier. Overall, data aided by SeqGAN always enhanced sentiment classification performance, recall, and F1-scores for deep/hybrid-based methods, whereas error rates were reduced, with the strongest

performer being RF.

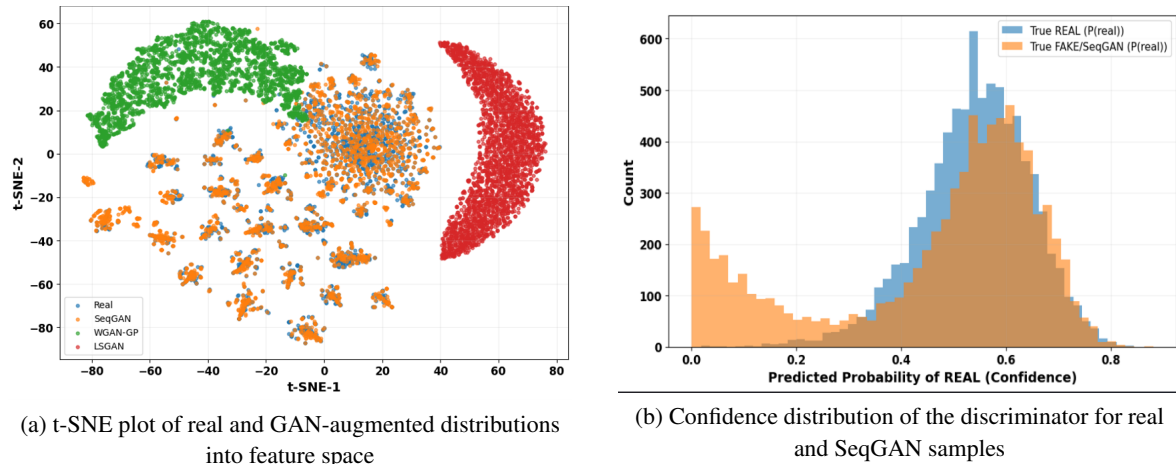


Figure 7. Visualization of GAN-based data augmentation and discriminator behavior

A closer examination of the evaluation metrics (shown in Tables 3, Table 4, and Table 5) reveals that in most of the machine learning and deep learning models, data augmentation always boosted the model accuracy, and SeqGAN achieved better performance than the conventional NLP-based augmentation. CNN exhibited the highest gain in accuracy (from 0.72 to 0.84) since the variety and contextually realistic examples produced by SeqGAN helped this model to learn richer local semantics patterns, and also prevented overfitting. The combination of synthetic sequences also resulted in significant improvements in recall and F1-score for LSTM and CNN+BiLSTM, highlighting how the extra sequences enhance their ability to understand long-term context and correctly identify minority sentiment patterns. In contrast, TinyBERT showed only marginal gains, as the pretrained contextual representations already yield good language understanding and require less additional data augmentation. Random Forest showed the best overall result, indicating that by mixing balanced synthetic samples with ensemble learning, it can generate more effective decision boundaries and better generalise. Moreover, for the majority of the models, the MAE and MSE were decreased, and the MCC was increased after applying the SeqGAN augmentation, which shows that the number of prediction errors are lesser and the agreement between the predicted and actual classes of sentiment is stronger. In general, these findings indicate that SeqGAN is able to produce more semantically coherent and diverse training samples than conventional NLP augmentation, which results in better feature learning, class balance, and classification performance regardless of the model architectures. In general, the results of the experiments show that the use of data augmentation had a positive effect on sentiment classification for almost all the models assessed. Both NLP-based and SeqGAN-based augmentations enhanced results over the original dataset; however, the performance of SeqGAN was found to be superior with respect to generalization, prediction error, and classification scores in the majority of the scenarios. The results show that creating a diverse and contextually rich synthetic data set is a good approach to boost the sentiment classification performance, especially for deep learning and hybrid models trained with small or imbalanced data sets.

To further validate the effectiveness of SeqGAN-based augmentation, the quality of the generated samples is analyzed in terms of their similarity to the real data distribution and the diversity of the synthesized samples. Figure 7 (a) illustrates the extent to which the SeqGAN projection has a strong overlap with the manifold representing the real data. Contrary to the SeqGAN projection, the WGAN-GP and LSGAN projections have isolated points forming distinct clusters, thus indicating a large gap between the two distributions. The equally distributed samples in the SeqGAN projection indicate diversity while avoiding mode collapse, thus explaining the improvement in the performance of the sentiment analysis task achieved by the SeqGAN. Figure 7 (b) demonstrates strong overlap between the real data samples and the SeqGAN-generated samples for the mid-to-high probability range, signifying strong realism and similarity. While having a minor tail for the low-confidence samples, the lack of bimodal separation ensures strong adversarial convergence. This trend ensures the increments in the recall rate, F1-Score, and MCC values, thus signifying that the SeqGAN-generated data improves the robustness of the model instead of adding noise data.

5. CONCLUSION

This study shows that using SeqGAN for data augmentation greatly improves sentiment analysis compared to both the original data and traditional NLP methods. The models learned better and performed more accurately because SeqGAN created more diverse and realistic examples, while the NLP techniques only gave a small boost. On average, SeqGAN improved performance by about 4–5% across different models. The RF classifier achieved a benchmarking result compared to other Models, reaching 86% accuracy. The expanded dataset also greatly benefited the DL and hybrid models. Although the results are encouraging, this study has some limitations. It focused only on two sentiment classes and used English-language data. In the future, this approach can be expanded to generalize SeqGAN towards multi-class sentiment and cross-lingual sentiment analysis, incorporate hybrid strategies for augmentation, and consider lightweight transformer-based generators for large-scale deployability. To make the models even more robust. Overall, the results show that SeqGAN-based augmentation is a powerful and effective way to improve sentiment classification performance and help models deal better with limited or imbalanced data.

The proposed framework for augmenting with SeqGAN proved to be very effective in improving the performance of sentiment classification, but there are several avenues for further studies. The proposed framework can be extended to multi-class sentiment analysis by adding neutral and fine-grained emotion classes in the future. Secondly, the experiments were run with English-language social media information. Evaluating the proposed approach on multilingual and cross-lingual datasets would help to gain more understanding of the generalization capability of the proposed approach across different languages and domains. Furthermore, future work could combine large pre-trained language models like BERT, RoBERTa, or even the more recent large language models (LLMs) with SeqGAN to create more contextually rich and semantically accurate synthetic text. To further enhance the classification robustness and overall predictive accuracy, the proposed augmentation framework can be applied to other domain-specific datasets, including healthcare, finance, and product reviews, and more complex model architectures of GANs and diffusion-based text generation models can be explored.

ACKNOWLEDGMENT

We would like to express our gratitude to American International University-Bangladesh (AIUB) for providing funding and support throughout this research.

REFERENCES

- [1] R. Sharma, S. Nigam, and R. Jain, "Polarity detection at sentence level," *Int J Comput Appl*, vol. 86, no. 11, pp. 29–33, 2014.
- [2] V. Umarani, A. Julian, and J. Deepa, "Sentiment analysis using various ML and DL techniques," *J Niger Soc Phys Sci*, vol. 3, pp. 385–394, 2021.
- [3] F. Jemai, M. Hayouni, and S. Baccar, "Sentiment analysis using ML algorithms," in *2021 IEEE IWCMC*, 2021, pp. 775–779.
- [4] C. Shorten, T. Khoshgoftar, and B. Furht, "Text data augmentation for DL," *J Big Data*, vol. 8, 2021.
- [5] P. Ghadekar, M. Jamble, A. Jaybhay, B. Jagtap, A. Joshi, and H. More, "Text data augmentation using generative adversarial networks, back translation and EDA," in *2023*, pp. 391–401.
- [6] H. Q. Abonizio, E. C. Paraiso, and S. Barbon, "Toward text data augmentation for sentiment analysis," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 5, pp. 657–668, 2022.
- [7] Z. Khanam, B. Alwasel, H. Sirafi, and M. Rashid, "Fake news detection using ML approaches," *IOP Conf Ser: Mater Sci Eng*, vol. 1099, p. 012040, 2021.
- [8] W. Cavnar, J. Trenkle *et al.*, "N-gram-based text categorization," in *Proc SDAIR-94, 3rd Annu Symp Doc Anal Inf Retr*, 1994, p. 14.
- [9] İ. Tarımer, A. Çoban, and A. Kocaman, "Sentiment analysis on IMDB movie comments and twitter data by ML and vector space techniques," *arXiv preprint arXiv:1903.11983*, 2019.

- [10] A. McCallum, K. Nigam *et al.*, “A comparison of event models for naive bayes text classification,” in *AAAI-98 Workshop Learn Text Categorization*, 1998, pp. 41–48.
- [11] J. Rennie, L. Shih, J. Teevan, and D. Karger, “Tackling the poor assumptions of naive bayes text classifiers,” in *Proc 20th Int Conf Mach Learn (ICML-03)*, 2003, pp. 616–623.
- [12] L. Jiang, C. Li, S. Wang, and L. Zhang, “Deep feature weighting for naive bayes and its application to text classification,” *Eng Appl Artif Intell*, vol. 52, pp. 26–39, 2016.
- [13] C. Aggarwal and C. Zhai, “A survey of text classification algorithms,” in *Mining Text Data*. Springer, 2012, pp. 163–222.
- [14] X. Gong, W. Ying, S. Zhong, and S. Gong, “Text sentiment analysis based on transformer and augmentation,” *Frontiers in Psychology*, vol. Volume 13 - 2022, 2022.
- [15] K. Alahmadi, S. Alharbi, J. Chen, and X. Wang, “Generalizing sentiment analysis: a review of progress, challenges, and emerging directions,” *Social Network Analysis and Mining*, vol. 15, 04 2025.
- [16] A. Igali, A. Abdrakhman, Y. Torekhan, and P. Shamo, “Tracking emotional dynamics in chat conversations: A hybrid approach using DistilBERT and emoji sentiment analysis,” *arXiv preprint arXiv:2408.01838*, 2024.
- [17] F. Masoumi and M. Bahrani, “Utilizing distilBERT transformer model for sentiment classification of COVID-19’s persian open-text responses,” *arXiv preprint arXiv:2212.08407*, 2022.
- [18] X.-Y. Fu, C. Chen, M. Laskar, S. Gardiner, P. Hiranandani, and S. Tn, “Entity-level sentiment analysis in contact center telephone conversations,” *arXiv preprint arXiv:2210.13401*, 2022.
- [19] K. Tan, C. Lee, K. Anbananthen, and K. Lim, “RoBERTa-LSTM: a hybrid model for sentiment analysis with transformer and recurrent neural network,” *IEEE Access*, vol. 10, pp. 21 517–21 525, 2022.
- [20] M. Cliche, “BB.twtr at semeval-2017 task 4: Twitter sentiment analysis with CNNs and LSTMs,” *arXiv preprint arXiv:1704.06125*, 2017.
- [21] A. S. Imran, R. Yang, Z. Kastrati, S. M. Daudpota, and S. Shaikh, “The impact of synthetic text generation for sentiment analysis using GAN based models,” *Egyptian Informatics Journal*, vol. 23, no. 3, pp. 547–557, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S110866522000342>
- [22] R. Yang and M. Edalati, “Using GAN-based models to sentimental analysis on imbalanced datasets in education domain,” *arXiv preprint arXiv:2108.12061*, 2021.
- [23] P. Sundarreson and S. Kumarapathirage, “SentiGEN: Synthetic data generator for sentiment analysis,” *Journal of Computing Theories and Applications*, vol. 1, pp. 461–477, 04 2024.
- [24] J. Luo, M. Bouazizi, and T. Ohtsuki, “Data augmentation for sentiment analysis using sentence compression-based SeqGAN with data screening,” *IEEE Access*, vol. 9, pp. 99 922–99 931, 2021.
- [25] K. Wahyu Trisna, J. Huang, Y. Chen, and I. G. J. E. Putra, “Dynamic text augmentation for robust sentiment analysis: Enhancing model performance with EDA and multi-channel CNN,” *IEEE Access*, vol. 13, pp. 31 978–31 991, 2025.
- [26] P. Eigenschink, T. Reutterer, S. Vamosi, R. Vamosi, C. Sun, and K. Kalcher, “Deep generative models for synthetic data: A survey,” *IEEE Access*, vol. 11, pp. 47 304–47 320, 2023.
- [27] K. Silva, B. Can, R. Sarwar, F. Blain, and R. Mitkov, “Text data augmentation using generative adversarial networks – a systematic review,” *Journal of Computational and Applied Linguistics*, vol. 1, pp. 6–38, 07 2023.

- [28] M. Mohammadi, M. R. Amin, and S. Tavakoli, "Boosting sentiment analysis in Persian through a GAN-based synthetic data augmentation method," in *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, M. El-Haj, A. Haddad, C. Amol, S. Ahmadi, H. Paterson III, I. Ezeani, S. Ezzini, and P. Rayson, Eds. Abu Dhabi, UAE: Association for Computational Linguistics, Jan. 2025, pp. 54–63. [Online]. Available: <https://aclanthology.org/2025.abjadnlp-1.7/>
- [29] Kazanova, "Sentiment140 dataset with 1.6 million tweets," <https://www.kaggle.com/datasets/kazanova/sentiment140>, 2019, accessed: 2026-07-04.
- [30] M. Ahmad, S. Aftab, and I. Ali, "Sentiment analysis of tweets using SVM," *Int J Comput Appl*, vol. 177, pp. 975–8887, 2017.
- [31] M. Rezwanul, A. Ali, and A. Rahman, "Sentiment analysis on twitter data using KNN and SVM," *Int J Adv Comput Sci Appl*, vol. 8, 2017.
- [32] M. Aufar, R. Andreswari, and D. Pramesti, "Sentiment analysis on youtube social media using decision tree and random forest algorithm: A case study," in *2020 International Conference on Data Science and Its Applications (ICoDSA)*. IEEE, 2020, pp. 1–7.
- [33] H. Parveen and S. Pandey, "Sentiment analysis on twitter data-set using naive bayes algorithm," in *2016 2nd Int Conf Appl Theor Comput Commun Technol (iCATccT)*, 2016, pp. 416–419.
- [34] P. S. Reddy, D. R. Sri, C. S. Reddy, and S. Shaik, "Sentimental analysis using logistic regression," *International Journal of Engineering Research and Applications*, vol. 11, no. 7, pp. 36–40, 2021.
- [35] M. Umer, Z. Imtiaz, M. Ahmad, M. Nappi, C. Medaglia, G. Choi, and A. Mehmood, "Impact of convolutional neural network and fasttext embedding on text classification," *Multimed Tools Appl*, vol. 82, 2022.
- [36] D. Murthy, S. Allu, B. Andhavarapu, and M. Bagadi, "Text based sentiment analysis using LSTM," *Int J Eng Res Technol* V9, 2020.
- [37] J. Vimali and S. Murugan, "A text based sentiment analysis model using bi-directional LSTM networks," in *2021 6th Int Conf Commun Electron Syst (ICCES)*, 2021, pp. 1652–1658.
- [38] Y. Lee, J. Saxe, and R. Harang, "CATBERT: context-aware tiny BERT for detecting social engineering emails," *arXiv preprint arXiv:2010.03484*, 2020.
- [39] M. Rhanoui, M. Mikram, S. Yousfi, and S. Barzali, "A CNN-BiLSTM model for document-level sentiment analysis," *Mach Learn Knowl Extr*, vol. 1, no. 3, pp. 832–847, 2019.
- [40] F. Sun and N. Chu, "Text sentiment analysis based on CNN-BiLSTM-attention model," in *2020 International Conference on Robots & Intelligent System (ICRIS)*, 2020, pp. 749–752.
- [41] R. Shu, H. Bui, and S. Ermon, "Ac-gan learns a biased distribution," in *NIPS Workshop on Bayesian Deep Learning*, vol. 8, 2017, p. 34.
- [42] Y. Liu, Q. Zhao, and C. Jiang, "Conditional image generation using feature-matching GAN," in *2017 10th Int Congr Image Signal Process, BioMedical Eng Informatics (CISP-BMEI)*, 2017, pp. 1–5.
- [43] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

BIOGRAPHIES OF AUTHORS

Shafin Mahmood   Shafin is a B.Sc. Computer Science student at American International University-Bangladesh. His areas of research are Artificial Intelligence, Machine Learning, Deep Learning, Natural Language Processing, generative adversarial networks, and high-accuracy AI-driven decision frameworks. He is co-author of the 2025 MDPI Drones paper "Multi-Agent Actor–Critic Frameworks for UAV Swarm Networks" (Q1, IF 4.4). He is committed to creating intelligent distributed systems and describing theoretical models as concrete, reproducible research outputs. He can be contacted at: shafin26103@gmail.com and personal website <https://shafin-mahmood.github.io/>



Sayma Akter Trina   Sayma is a B.Sc. student in Computer Science at the American International University-Bangladesh, with a strong academic focus on artificial intelligence, machine learning, and data-driven systems. Her research interests include machine learning, deep learning, and algorithm optimization. She is dedicated to building interpretable and scalable intelligent systems that integrate theoretical advancements with real-world applications. She can be contacted at trinasayma5191@gmail.com



Md. Easin Khandokar  Easin is a B.Sc. student in Computer Science at the American International University-Bangladesh (AIUB) with research interests in Machine Learning, Deep Learning, Natural Language Processing, and Cyber Security. His research and development focus lies in designing and implementing AI-based systems that integrate deep learning and NLP to solve real-world challenges through intelligent automation and secure computing solutions. He can be contacted at: easinkhandokarcse@gmail.com.



Nuzhat Tabassum   Nuzhat Tabassum is an Assistant Professor at the American International University-Bangladesh (AIUB), Faculty of Science and Technology, Department of Computer Science. She completed her M.Sc. in Artificial Intelligence from Warsaw University of Technology (WUT), Warsaw, Poland, and B.Sc. in Computer Science and Engineering from Chittagong University of Engineering and Technology, Bangladesh. Prior to AIUB, she worked as an Assistant Professor at the Military Institute of Science and Technology (MIST), and before that, she was a Lecturer at BRAC University. She also worked as a Junior AI Engineer at Smartlife Robotics, Warsaw, Poland, and as a BI Specialist at Robi Axiata Limited. Nuzhat has success and experience in both industry and academia. She is committed to developing her abilities and making a significant impact in her profession. Her field of interest includes Deep learning, Machine learning, Natural Language Processing, Data Mining, Image Processing, and Augmented Reality.