International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

A Super Resolution Generative Adversarial Network with Bidirectional Encoder Representations from Transformers for Aspect-based Sentiment Analysis

Vidya M. S.[1], A. Mari Krithima[2], Ambika G. N.[3], Shankar R.[4]

Submitted:13/03/2024 **Revised**: 28/04/2024 **Accepted**: 05/05/2024

Abstract: Aspect-based sentiment analysis (ABSA) determines the sentiment polarity associated with features stated in a sentence or text. Current sentiment analysis algorithms based on aspect categories frequently need to account for the implicit context of aspect-category information. Existing models may yield acceptable results but often require more topic expertise. Current sentiment analysis algorithms based on aspect categories frequently need to account for the implicit context of aspect-category information. Existing models may yield acceptable results, but they often lack topic expertise. We introduce a novel technique that employs Super Resolution Generative Adversarial Networks (SRGAN) and Bidirectional Encoder Representations from Transformers (BERT) to solve the complexities of this problem and improve sentiment analysis accuracy. We offer a synergistic framework in this study that employs SRGANs to enhance the resolution and clarity of text representations, followed by incorporating BERT's contextual embeddings for aspect-based sentiment analysis. By integrating SRGAN's ability to build high-resolution text representations with BERT's contextualized language understanding, SRGANs-BERT overcomes the issues of aspect extraction, sentiment polarity identification, and context-dependent sentiment comprehension. The combination of SRGANs with BERT suggests a path forward for aspect-based sentiment analysis, with applications including customer feedback analysis, market research, and social media sentiment monitoring. We illustrate the usefulness of our suggested strategy through rigorous testing on benchmark data sets. Our findings show that combining SRGANs and BERT significantly improves aspect-based sentiment analysis's efficacy.

Keywords: *Aspect-based sentiment analysis (ABSA), Bidirectional Encoder Representations from Transformers (BERT), Sentiment polarity identification, Social media sentiment monitoring, Super Resolution Generative Adversarial Networks (SRGAN),*

1. Introduction

Natural language processing (NLP)'s complex research area is sentiment analysis or SA for short. In order to anticipate the total sentiment polarity of those aspects, a text classification job aims to detect the sentiment aspects and their related sentiment polarities in the text [1]. Aspect-based sentiment analysis (ABSA), document-level sentiment analysis, and sentence-level sentiment analysis are the three core subfields that make up the field of sentiment analysis [2, 3]. The amount of sentiment features and their corresponding sentiment polarities that are identified within a given text [4] is the primary distinction between these subfields. The goal of ABSA, a subset of text sentiment analysis [5], is to ascertain the attitude held towards particular aspect phrases contained inside a sentence. ABSA focuses on these aspect phrases as opposed to standard sentiment analysis, which examines the entirety of words [6].Since each aspect's sentiment is captured separately, this method offers a deeper understanding of user reviews.

As a direct consequence of the expansion of NLP and AI methodologies, Deep learning and machine learning algorithms have found widespread application inside ABSA [7]. In addition, neural network models that included the attention mechanism performed far better than traditional methods. Recently, it has been demonstrated that pre-trained language models like ELMo, OpenAI GPT, and BERT are efficient for various NLP tasks. BERT has mainly succeeded in Question Answering and Natural Language Inference (NLI). ALBERT [8] and Roberta [9] are two BERT variations that have also demonstrated good performance on particular tasks.

Additionally, [10] research has demonstrated that knowing a word's dependency on others might improve performance. For instance, you may construct a model using key-value memory networks to finish ABSA with word dependencies. For word semantic representation, we make use of weighted word dependence information. To be more specific, the memory system considers word dependence relations and associated dependency sorts. Then it combines information

[1]*Assistant Professor, Department of Computer Science & Engineering, BMSIT&M, Bangalore, India,*
Email: rvidyapai@bmsit.in
[2]*Assistant Professor, Department of Computer Science & Engineering, BMSIT&M, Bangalore, India,*
Email: krithi.a@bmsit.in
[3]*Assistant Professor Department of Computer Science & Engineering, BMSIT&M, Bangalore, India,*
Email: ambikagn@bmsit.in
[4]*Assistant Professor Department of Computer Science & Engineering, BMSIT&M, Bangalore, India,*
Email: shankar@bmsit.in

based on each of these factors contributions. The semantic relationship between each word in the raw sentence and the entire auxiliary phrase can help decide the weights when building the final representation using all the words in the primary sentence. This is done so that the relative importance of each word can be reflected in its weight. Super Resolution Generative Adversarial Networks have successfully produced high-resolution images from low-resolution inputs. These networks were initially created for image super-resolution tasks. A generator network is trained to make data identical to accurate data, while a discriminator network learns to tell the difference between the two. This is the fundamental notion underlying GANs. The generator-discriminator paradigm can be changed in the context of aspect-based sentiment analysis to improve the accuracy of sentiment predictions at the part equal.

However, the advantages of the ABSA work are limited when BERT is used directly. To better use pre-trained language models for ABSA, [11] used BERT to create sentence pairs by creating an auxiliary sentence, which had been proven successful. The goal was to increase confidence in the findings. Therefore, we took this step. Finding the best label for a collection of two sentences is the objective of sentence semantic matching [12]. The original ABSA activity has been transformed into a job of matching sentences semantically, which has led to a change in the prediction aims to focus on the linkages between sentences. Different semantic expressions can be produced by combining sentence pairs with varying resemblance degrees [13]. Relational learning should therefore receive adequate attention. Deeper relation learning between sentence pairings has yet to investigate thoroughly.

The main contributions of this work can be summed up by the traits listed below:

- To progress the clarity and quality of text demonstrations SRGANs are used in the synergistic framework to enhance the clarity and resolution of text representations.
- Contextual embedding from BERT is then used for aspect-based sentiment analysis.
- By combining SRGAN's ability to create high-resolution text representations with BERT's contextualized language understanding, SRGANs-BERT tackles the problems of aspect extraction, sentiment polarity identification, and context-dependent sentiment understanding.
- The combination of SRGANs with BERT provides applications for aspect-based sentiment analysis, including customer feedback analysis, market research, and sentiment monitoring on social media.
- Through thorough testing on benchmark data sets, we show the effectiveness of our strategy.

2. Literature Survey

GRACE (Graph-Attentive Cascaded Labelling) is a brand-new method for ABSA introduced by Luo et al. [14]. During the process of assigning a sentiment polarity label, GRACE makes use of a technique known as cascaded labeling. This helps to ensure that aspect terms interact more effectively with one another and places a more significant amount of emphasis on sentiment tokens. Two decoder modules comprise the model: one is responsible for aspect term extraction (ATE), and the other is for aspect sentiment classification (ASC). The model performs both of these tasks. GRACE incorporates a stacked multi-head attention mechanism in the ASC module to record the interaction between the several aspect terms. In addition to this, the model uses a gradient harmonization loss function to deal with the problem of imbalanced labels. Karimi et al. [15], who focused on enhancing the BERT model, presented some novel advancements that may be utilized for ABSA jobs. To improve the functionality of BERT for ABSA, the authors suggested adding two additional modules: parallel aggregation and hierarchical aggregation. In a study by Ben Veyseh et al. [16], syntax-based regulation and the GGCN (Graph-based Global Context Network) technologies were used to improve ABSA. Syntax, graph convolution and regulation (GCR), representation learning (RL), and model consistency (SMC) are some of the key modules that are combined in this method.

Yu and Zhang [17] introduced the Multi Weight Graph Convolutional network (MWGCN). Various attention-processing challenges must be resolved to differentiate aspect-relevant semantics and consider aspect location information. These difficulties include tackling the problem of the feature's long-distance reliance and the difficulty in precisely distinguishing such semantics. It is also difficult to discern these meanings. The MWGCN approach uses two distinct weighting algorithms: the Local Context Weighted Adjacency Graph and Multigrain Weighted Dot-Product Weighting (MGDW). Both of these weighing methods are dot-product weighting algorithms. While maintaining the semantics of the larger context as a whole, MGDW focuses more attention on the characteristics related to aspects of the data.

To increase the accuracy of their predictions, Tian et al. [18] employed the use of key–value memory networks to encode the dependency label on arcs within the dependency tree. Recently, considerable performance gains have been made by encoding syntactic structures with graph neural networks (GNNs). An innovative ensemble-based approach to the recognition of facial expressions of emotion was presented by Moung et al. [19]. By merging three different classification models—a customized CNN, InceptionV3, and ResNet50—, they exhibited improved performance in classifying positive and neutral emotions. This was done so that it could be compared to the performance of individual classifiers. A speech model was presented by Chen et al. [20] in the era of potent deep learning techniques to recognize the various emotions expressed by people's voices through an examination of the sounds their voices create. They captured voice representations by employing log-Mel spectrograms generated from speech data and combining attentional processes with bi-directional long short-term memory (Bi-LSTM), convolutional neural network (CNN), and other neural network architectures. This allowed them to evaluate the data in a way that allowed them to recognize the sounds made by the human voice. This allowed them to analyze the data in a way that allowed them to identify individual voices.

Arumugam and Nallaperumal [21] give two different approaches to the problem. The first approach, which we'll refer to as AASGCN, is an improvement on ASGCN that adds adaptive weights. These weights make it possible to convey the semantic significance of opinion aims more complexly and precisely. The second strategy, EISR and involving emotionally taxing data, is designed to be employed along with the sentiment analysis tactic. The research does not examine any potential downsides or blockages associated with the suggested changes. If these limitations were lifted, a more nuanced comprehension of how the proposed models function and how they can be used would become attainable.

Priyasad et al. [22] introduced an approach to detect emotions through audio-text analysis. They employed a Deep Convolutional Neural Network (DCNN) to process auditory characteristics. The methodology involved integrating two parallel components: a Bidirectional Long Short-Term Memory model and the DCNN to handle textual features. These two branches were combined using mid-level fusion techniques." Its average value for testing the compatibility of audio and text is 79.22%. Li et al. [23] created a unique framework called the bidirectional emotional recurrent unit (BiERU) to analyze the sentiments expressed in conversations. This method is characterized by its speed, concision, and high computational efficiency. Similarly, The RoBERTa-LSTM model for doing sentiment analysis was introduced by Tan et al. [24]. This model combines the benefits offered by the Transformer and sequence models. Even in instances where opinion and aspect words are physically separated from one another inside the phrase, Hou et al. [25] perform a decent job of establishing the relationship between the two types of words. There is no connection between the amount of GCN layers used in training and gains in performance.

2.1. Limitations of Existing System

- Some elements could be vague or context-dependent, making it challenging for models to determine precisely which aspect should be studied. This may lead to sentiment needing to be correctly classified.
- While aspect-based sentiment analysis offers sentiment ratings for various factors, it could miss subtle, nuanced expressions of sentiment. Texts containing nuanced feelings or complicated emotions may be complex for the algorithm to decipher correctly.
- Predefined elements and sentiment labels may add bias depending on the annotators' points of view. The model's predictions and fairness may be affected by this bias.
- Negatives and sentiment modifiers can considerably impact the sentiment expressed regarding a specific component. These language intricacies may be difficult for models to grasp appropriately, which could result in inaccurate sentiment classification.

2.2. Problem Identification

- In many circumstances, aspects addressed in a text are unique and are not frequently encountered in training data, which causes problems with data sparsity. Models must generalize sentiment analysis to include features not seen during training, which calls for strong approaches to deal with sparse data.
- It could be challenging to evaluate whether a particular component in a sentence has a good, negative, or neutral sentiment. Sentiment classification is difficult since aspects can be stated using a variety of linguistic constructions, idiomatic expressions, and negations.
- Aspects frequently occur in several situations, and the attitude towards them might change depending on the text. For accurate sentiment analysis, context comprehension and word-disambiguation techniques must be improved.
- Due to language use, vocabulary, and sentiment expressions unique to each domain, models trained on one domain may perform poorly on another. It might not be easy to adapt models to diverse domains while keeping them accurate.

3. Proposed System

This section proposes a novel method that combines BERT and Super Resolution Generative Adversarial Networks (SRGAN) to tackle the complexities of this issue and increase sentiment analysis accuracy. This study describes a synergistic method for enhancing the resolution and clarity of text representations that use SRGANs. These improved representations are paired with the contextual embeddings from BERT for ABSA analysis. By combining SRGAN's ability to build high-resolution text representations with BERT's contextualized language comprehension, SRGANs-BERT addresses the difficulties of aspect extraction, sentiment polarity identification, and context-dependent sentiment understanding. With applications such as customer feedback analysis, market research, and social media sentiment monitoring, the combination of SRGANs and BERT points to a promising direction for aspect-based sentiment analysis. Figure 1 displays the SRGANs-BERT block diagram.

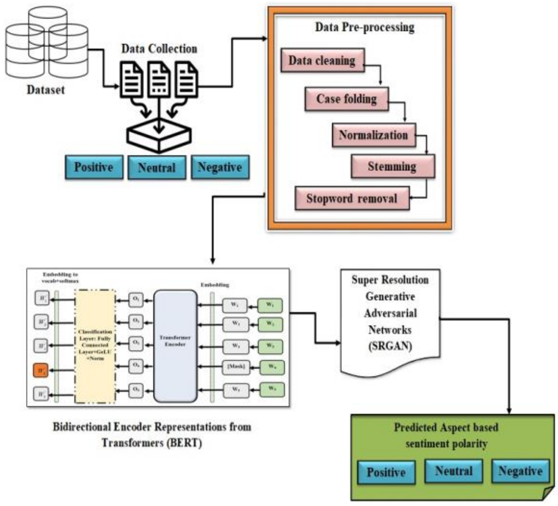

Fig.1. Block diagram of SRGANs-BERT method

3.1. Data Sets

Our LSOIT model was validated using the benchmarking datasets MAMS, Twitter, Rest15, Rest16, Rest14, Laptop, and MAMS-small to determine its efficacy. Table 1 contains comprehensive statistical data for each dataset.

- Twitter is a hub for social media and microblogging where users may share and discuss a wide range of content, including news, humor, opinions on current events, and personal feelings. Its use for sentiment analysis in numerous research papers has proven noteworthy. [26]
- The restaurant model is based on the SemEval 2016 restaurant dataset and its expanded version, which includes aspects, categories, and emotions. The primary restaurant applications involve sentiment analysis and the responsibilities that go along with it. [27].
- Reviews from Amazon.com were collated for ten laptops from six distinct brands: Asus, Acer, Samsung, Lenovo, MBP, and MSI. Given that it contains significant implicit sentiment data, this dataset is particularly beneficial to scholars researching sentiment analysis [28].
- MAMS is a difficult and well-liked dataset in the ABSA challenge setting. Its structure has two or more sides to each sentence, each with a different polarity of sentiment[29].

Table 1. Statistics of Evaluation Datasets

Dataset	Positive		Negative		Neutral	
	Train	Test/Dev	Train	Test/Dev	Train	Test/Dev
MAMS	3380	400/403	5042	607/604	2764	329/325
Laptop	994	341	464	169	870	128
Res15	912	326	36	34	256	182
Res16	1240	469	69	30	439	117
Twitter	1561	173	3172	346	1560	173
Res14	2164	728	637	196	807	196
MAMSsmall	1089	400/403	1627	607/604	892	329/325

3.2. Data collection

During this stage, reviews are gathered by employing the web scraping approach. A significant number of reviews were obtained from booking.com using Scrapy. These reviews previously split positive, negative, and neutral contents into three sections [30]. It optimizes the procedure and facilitates the differentiation of various qualities concerning relative positive, negative, and neutral opinions.

3.3. Pre-processing

Proper data processing was necessary since the raw text was used to collect the data for this investigation. To improve text extraction, we first preprocessed the datasets. A similar strategy may also drastically change how text is processed [31].

3.3.1. Data cleaning

Text documents in a dataset frequently consist of elements like letters, links, usernames, numbers, symbols, and other components, making text analysis complex. This method has been employed to detect and remove extraneous characters and data.

3.3.2. Case folding

Case folding was used to transform all uppercase in each word to lowercase to standardize the dataset based on the parameters.

3.3.3. Normalization

Slang phrases frequently appear in sentence texts taken from social media. Bokap, for instance, denotes "dad" in English or "ayah" in Indonesian. Such colloquialisms may provide additional vector dimensions, lengthening the computation [32]. Slang must be converted into formal language by employing its orthography.

3.3.4. Stemming

Stemming involves reducing the vocabulary space by eliminating prefixes and suffixes from the original word.

3.3.5. Stopword removal

In text processing, removing stopwords—words like the" and "also" that don't make sense in a sentence—is common.

3.4. Bidirectional Encoder Representations from Transformers (BERT)

A modular NLP system can be customized to suit a varied variety of specialized applications by leveraging the open-source capabilities of Google's transformers, specifically BERT [33]. It can infer context from neighboring texts, allowing it to comprehend the significance of contextually unclear languages. Figure 2 shows a graphic representation of BERT's underlying architecture.

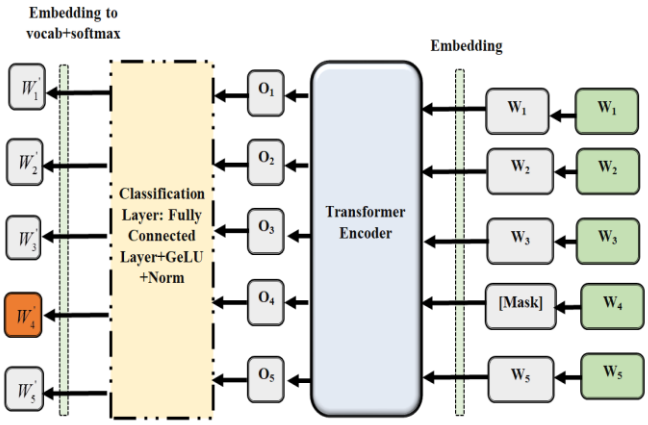

Fig. 2. Structure of BERT [33]

3.4.1. BERT for Contextual Learning

Essential word qualities must be extracted from the phrases after giving each word considerable thought. By creating an embedding for each word in the sentence, running the matching word embeddings to generate an embedding for $P=\sum w_i$ the sentence, and comparing the result to the vocabulary size lookup table in the BERT model, the BERT architecture does this feature extraction. Using the resulting embedding matrix $y=[y_1,y_2,y_3,...y_n]$, the input phrases were transformed into vector representations and fed into the BERT (Eq. 3).

$$H=\{h_1,h_2,...,h_n\}=BERT(y) \tag{1}$$

Where h represents the hidden forms of y at each moment.

3.5. Super Resolution Generative Adversarial Networks (SRGAN)

Since the initial suggestion by Ian Goodfellow et al. [34], Generative Adversarial Networks (GANs) have been accepted by various scientists. These GANs comprise two opposing components: the discriminator and the generator, which compete in a strategic minimax contest. The generator must simulate data distribution to create lifelike samples, intending to convince the discriminator of their validity. On the other hand, the discriminator is responsible for determining the origin of incoming samples. The effectiveness of these competing elements determines the associated costs for each network.

The complete procedure is summarised in the following formulation (Equation 1), in which the value function $U(D,G)$ competes with both the discriminator and the generator.

$$\min_G \max_D U(D,G)=F_y[\log D(y)]+F_x[\log(1-D(G(x)))] \tag{2}$$

D(G(z)) is the discriminator's estimated probability that a real data instance z is fake, and D(G(z)) is the discriminator's estimated probability that a fake instance z is likely to be fake if $D(y)$ is the discriminator's estimated probability that a real data instance y is real and $G(x)$ is the generator's output given noise x. Ex represents the expected value across all occurrences of accurate data and the expected value across all inputs to random number generators.

Our research aims to improve textual representations with low resolution. Our research focuses on creating high-resolution textual representations for which the generator accepts low-resolution textual inputs. The network's discriminator determines the source of information by analyzing either synthetic or actual high-resolution textual representations. The method that governs network weights includes two types of losses: content-based and adversarial. It is worth noting that the adversarial loss does not move the discriminator during training. In contrast to the adversarial loss, which influences the weights of the generative network, the content loss helps shape the resulting high-resolution textual representation. Figure 3 depicts the basic layout of the GAN training procedure.

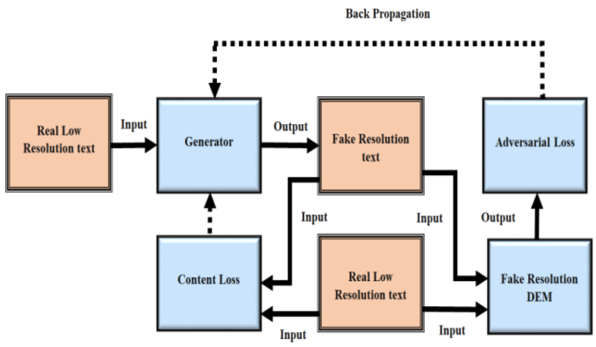

Fig. 3. Structure of the GAN training process

The process requires transforming low-resolution text representations into high-resolution equivalents, as seen in Figure 1. A discriminator compares the generated versions to the real ones to determine the legitimacy of these augmented text representations. The computation of an adversarial loss results from evaluating the discriminator's effectiveness, allowing adjustments to the weights of both the discriminator and the generator. Concurrently, the generator receives input via content loss assessment, which involves analyzing the pixel-level differences between the generated high-resolution text representations and their true counterparts.

3.5.1. Network Architectures

The network architecture comprises two key components: the discriminator and the generator, which function in opposition. The SRGAN [35] and EDSR [36] models serve as the architecture's core building elements. To explain how the generator works, it accepts low-resolution word representations as input and processes them via a convolutional layer with 64 feature maps. These representations are routed through a succession of residual blocks, the first of which is followed by twenty duplicated blocks. Each residual block comprises two convolutional layers, a three-dimensional kernel, and 64 feature mappings. Between these convolutional layers, an activation layer called ReLU is used. Intelligent connections, known as skip connections, exist between the later stages of the current residual block and the incoming data from the preceding component within each block. This integration aims to collect low-level features for improved generator performance. The final residual block's output is assessed similarly to its input data. The generator also has two more sections: upsampling blocks. To increase resolution, these blocks employ a 256-feature convolutional layer followed by sub-pixel convolutional layers. The output of the upsampling blocks is then sent through a final convolutional layer before being generated by the generator. The discriminator comprises nine convolutional layers with 33 filter kernels and 512 feature maps overall (increased from 64). A Leaky ReLU activation function with a coefficient of 0.2 is used after each convolutional layer. Striated convolutional layers are utilised to decrease the resolution of the text representation while tripling the number of features. The dense layers in the final convolutional layer are purposely placed before an adaptive average pooling layer. The network's discriminator output is created by finalizing the discriminator's output using a sigmoid function after the dense layers.

3.5.2. Loss Function

The loss functions used in neural networks will be examined in this section. The discriminator and the generator both use adversarial loss during training. The generator experiences content loss in addition to adversarial loss, which promotes faster convergence and results in finer data points.

3.5.3. Adversarial Loss

The GAN system's adversarial loss is a crucial element. This component serves as the discriminator's loss function in a novel way for the network. As anticipated, the adversarial feature significantly enhances the discriminator's ability to identify the origin of incoming data. During training, the discriminator and the generator use the adversarial loss, more precisely, the mean absolute error or L1 Loss. Eq. 2 expresses the formulation of the adversarial loss for the discriminator as follows:

$$I_{Dis} = \frac{1}{2m}\sum_{i=1}^{m} 1 - D(z_i)\,| + D(G(y_i))\,| \tag{3}$$

Where m is the entire amount of samples, zi is the amount of real high-resolution text representations, yi is the number of low-resolution text representations, and so on.

The generator uses adversarial loss to trick the discriminator to produce more plausible samples. The following is an estimate of the adversarial loss for the generator (Eq. 3):

$$I_{Ad_{Gen}} = \frac{1}{m}\sum_{i=1}^{m} 1 - D(G(y_i))\,| \tag{4}$$

Where m is the overall sample count, yi denotes low-resolution text representations.

3.5.4. Content Loss

The mean square error (MSE) is a commonly used optimization metric in numerous studies [37]. A loss function and an adversarial loss are used in conventional techniques to distinguish between actual and created data, capturing the quality of the synthesized information. Because textual representation incorporates numeric values relevant to earth surface elevations, using MSE to interpret the progress of the ongoing procedure is appropriate. MSE, in this context, acts as a meter for evaluating the efficacy of techniques in this sector.

$$I_{Con_{Gen}} = \frac{1}{m}\sum_{i=1}^{m} (y_i - z_i)^2 \tag{5}$$

Where m is the total number of samples, zi is the actual high-resolution text representations, and yi is the produced high-resolution text representations. The loss function for the generator (Eq. 5) mixes adversarial loss (Eq. 3) and content loss (Eq. 4) because many loss functions influence the generator:

$$I_{Gen} = I_{Con_{Gen}} + \alpha I_{Ad_{Gen}} \quad (6)$$

where $I_{Con_{Gen}}$ is content generator loss,

$I_{Ad_{Gen}}$ − Adversarial loss of generator,

α − Weight of adversarial loss.

4. Result and Discussion

This section presents details of the extensive trials assessing the SRGANs-BERT's classification performance and explainability. We contrast our model's classification performance with four commonly used, publicly accessible datasets' baselines. Additionally, we go over the analysis we did to confirm SRGANs-BERT's In this research using the existing systems are Selective attention based graph convolutional networks (SAGCN) [25], graph neural networks (GNNs)[18], bidirectional emotional recurrent unit (BiERU) [23], weight-oriented graph convolutional network (MWGCN) [17], Deep Convolutional Neural Network(DCNN)[22].

4.1. Experimental Setup

The 300-dimensional pre-trained word vector GloVe, to which Stanford University kindly contributed, served as the study's dictionary [38]. GloVe vectors are used to train the initial parameters of the untrained word vector. Since there is no official development dataset for the aspect-level dataset, a random selection of 20% of the initial training data was utilized as the development set, and the remaining 80% was used for training. Table 2 displays the experimental parameters for each neural network model: Results from five randomly initialized tests were averaged to ensure the correctness of the experiments as evaluation measures in this study, precision, recall, f1-score, accuracy, and macro-F1 are employed.

Table 2. Experimental parameter setting

Parameter	Value
Learning rate	0.001
λ	0.1
Optimizer	RMSProp
Epochs	15
Batch size	31
Decay rate	0.9
Dropout probability	0.5
Embedding dimension	300

4.2. Evaluation Metrics

To assess the effectiveness of models, we utilize the Accuracy and Macro - F1 (MF1) metric in Eqs. (7)–(11), the computation technique is illustrated. Accuracy displays the percentage of the categories in all correctly predicted samples. A performance classifier is typically better when its accuracy is higher. However, the accuracy does not reflect the classifier's performance when the dataset's imbalanced data distribution. The MF1 is used as an additional indicator. For a given sentiment polarity, the TPi represents the proportion of predicted samples. The FPi represents the proportion of samples from other categories incorrectly identified as samples of i. The FNi represents the proportion of samples from type i that were misidentified as samples from other classes. Pi denotes precision and recall rate by Ri. The average of all F1 categories is the MF1.

$$MF1 = \frac{1}{C}\sum_{i=1}^{C} F1_i \quad (7)$$

$$P_i = \frac{TP_i}{(TP_i + FN_i)} \quad (8)$$

$$R_i = \frac{TP_i}{(TP_i + FP_i)} \quad (9)$$

$$F1_i = \frac{2 \times P_i \times R_i}{P_i + R_i} \quad (10)$$

$$Accuracy = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C}(TP_i + FP_i + FN_i)} \quad (11)$$

4.2.1. Macro - F1 Analysis

Table 3. Macro - F1 Analysis for SRGANs-BERT method with existing systems

Dataset	SA-GCNN	GNNs	BiERU	MWGCN	DCNN	SRGANs-BERT
Rest14	91.97	93.98	92.13	93.99	91.87	95.99
Laptop14	91.34	92.34	93.57	92.88	90.44	95.26
Twitter	90.45	91.53	92.78	93.87	91.56	94.33
MAMS	92.14	90.99	93.91	93.99	92.55	95.78

In Figure 4 and Table 3, the SRGANs-BERT technique's Macro - F1 is contrasted with other widely used methods. The graph illustrates how the deep learning approach has enhanced performance with Macro - F1. For instance, the SRGANs-BERT model's Macro - F1 for the Rest14 dataset is 95.99%, whereas the Macro - F1 values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 91.97%, 93.98%, 92.13%, 93.99%, and 91.87%. Similarly, the SRGANs-BERT model's Macro - F1 for the Laptop14 dataset is 95.26%, whereas the Macro - F1 values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 91.34%, 92.34%, 93.57%, 92.88%, and 90.44%. Also, the SRGANs-BERT model's Macro - F1 for Twitter dataset is 94.33%, whereas the Macro - F1 values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 90.45%, 91.53%, 92.78%, 93.87%, and 94.33%. Like this, the suggested SRGANs-BERT model has a Macro - F1 of 95.78% under the MAMS dataset, compared to 92.14%, 90.99%, 93.91%, 93.99%, and 95.78% for the SA GCN, GNNs, BiERU, MWGCN, and DCNN models, respectively.

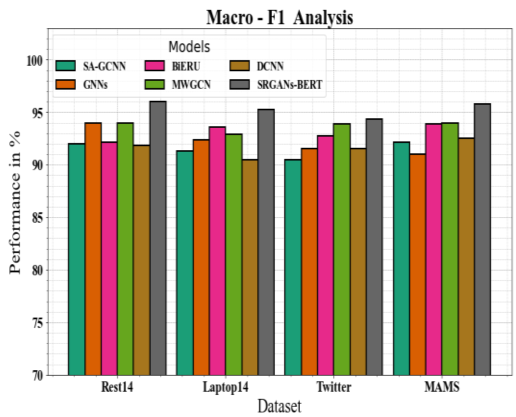

Fig.4. Macro - F1 Analysis for SRGANs-BERT method with existing systems

4.2.2. Precision Analysis

Table 4. Precision Analysis for SRGANs-BERT method with existing systems

Dataset	SA-GCNN	GNNs	BiERU	MWGCN	DCNN	SRGANs-BERT
Rest14	86.99	87.89	88.99	89.12	88.12	92.56
Laptop14	87.23	86.12	88.12	88.99	87.89	91.78
Twitter	86.56	85.99	87.67	88.56	86.54	90.99
MAMS	88.67	86.34	88.88	88.99	89.12	94.98

In Figure 5 and Table 4, the SRGANs-BERT technique's precision is contrasted with other widely used methods. The graph illustrates how the deep learning approach has enhanced performance with precision. For instance, the SRGANs-BERT model's precision for the Rest14 dataset is 92.56%, whereas the precision values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 86.99%,87.89%, 88.99%,89.12%, and 88.12%. Similarly, the SRGANs-BERT model's precision for the Laptop14 dataset is 91.78%, whereas the precision values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 87.23%, 86.12%,88.12%, 88.99%, and 87.89%. Also, the SRGANs-BERT model's precision for the Twitter dataset is 90.99%, whereas the precision values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 86.56%, 85.99%, 87.67%, 88.56%, and 86.54%. Like this, the suggested SRGANs-BERT model has a precision of 94.98% under the MAMS dataset, compared to 88.67%, 86.34%, 88.88%, 88.99%, and 89.12% for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models, respectively.

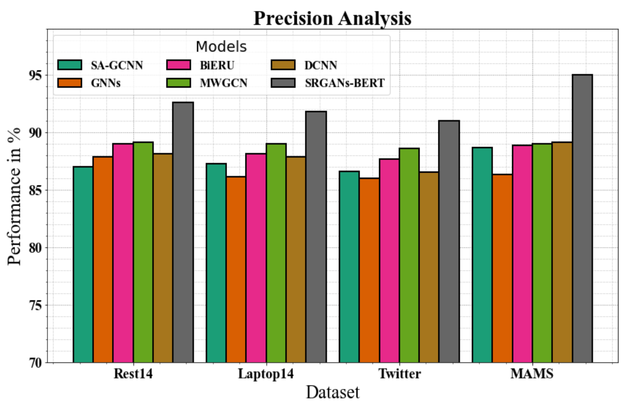

Fig.5. Precision Analysis for SRGANs-BERT method with existing systems

4.2.3. Recall Analysis

Table 5. Recall Analysis for SRGANs-BERT method with existing systems

Dataset	SA-GCNN	GNNs	BiERU	MWGCN	DCNN	SRGANs-BERT
Rest14	83.98	84.54	83.66	84.99	84.67	86.87
Laptop14	83.16	83.99	82.78	83.87	83.44	87.66
Twitter	82.19	83.87	81.55	84.21	82.76	85.55
MAMS	84.66	85.77	83.99	84.88	85.34	89.32

Figure 6 and Table 5 compare the SRGANs-BERT technique's recall performance to those of other widely utilized techniques. The graphical representation displays the deep learning methodology's enhanced recall. Notably, for the Rest14 dataset, the SRGANs-BERT model has a recall of 86.87%, while the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models have recall values of 83.98%, 84.54%, 83.66%, 84.99%, and 84.67%, respectively. Similarly, the SRGANs-BERT model's recall for the Laptop14 dataset is 87.66%, whereas the recall values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 83.16%, 83.99%, 82.78%, 83.87%, and 83.44%. Also, the SRGANs-BERT model's recall for the Twitter dataset is 85.55%, whereas the recall values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 82.19%, 83.87%, 81.55%, 84.21%, and 82.76%. Like this, the suggested SRGANs-BERT model has a recall of 89.32% under the MAMS dataset, compared to 84.66%, 85.77%, 83.99%, 84.88%, and 85.34% for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models, respectively.

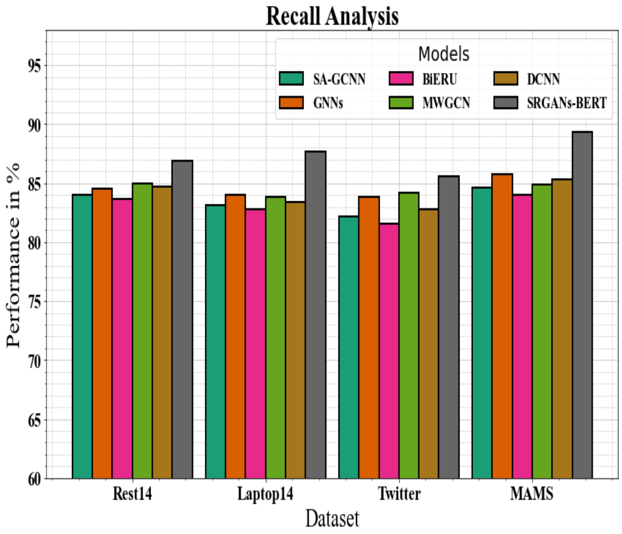

Fig.6. Recall Analysis for SRGANs-BERT method with existing systems

4.2.4. F1-Score Analysis

Table 6. F1-Score Analysis for SRGANs-BERT method with existing systems

Dataset	SA-GCNN	GNNs	BiERU	MWGCN	DCNN	SRGANs-BERT
Rest14	72.99	72.99	74.88	75.78	75.89	80.98
Laptop14	72.87	73.98	72.44	74.67	75.87	79.34
Twitter	71.56	72.67	73.98	74.12	75.67	78.81
MAMS	73.52	77.89	78.12	77.98	77.34	82.91

Figure 7 and Table 6 compare the f1-scores of the SRGANs-BERT technique to those of various frequently used methods. The graphical representation demonstrates the deep learning approach's higher performance regarding the f1-score. When analyzing the Rest14 dataset, for example, the SRGANs-BERT model achieves a f1-score of 80.98%. The SA-GCN, GNNs, BiERU, MWGCN, and DCNN models, on the other hand, achieve f1-score values of 72.99%, 72.99%, 74.88%, 75.78%, and 75.89%, respectively. Similarly, the SRGANs-BERT model's f1-score for the Laptop14 dataset is 79.34%, whereas the f1-score values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 72.87%, 73.98%, 72.44%, 74.67%, and 75.87%. Also, the SRGANs-BERT model's f1-score for the Twitter dataset is 78.81%, whereas the f1-score values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 71.56%, 72.67%, 73.98%, 74.12%, and 75.67%. Like this, the suggested SRGANs-BERT model has a f1-

score of 82.91% under the MAMS dataset, compared to 73.52%, 77.89%, 78.12%, 77.98%, and 77.34% for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models, respectively.

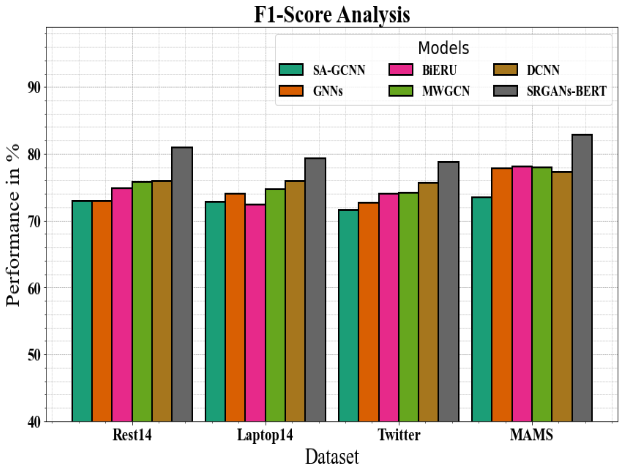

Fig.7. F1-Score Analysis for SRGANs-BERT method with existing systems

4.2.5. Accuracy Analysis

Table 7. Accuracy Analysis for SRGANs-BERT method with existing systems

Dataset	SA-GCNN	GNNs	BiERU	MWGCN	DCNN	SRGANs-BERT
Rest14	90.56	91.48	92.23	93.99	95.55	96.97
Laptop14	93.45	94.78	93.17	95.87	96.13	97.12
Twitter	90.34	90.04	91.98	91.16	92.54	96.99
MAMS	91.67	92.97	93.49	95.98	96.48	99.99

Figure 8 and Table 7 compare the accuracy of the SRGANs-BERT technique to other well-established methodologies. The graphical representation vividly displays the deep learning approach's better accuracy performance. On the Rest14 dataset, the SRGANs-BERT model achieves a remarkable accuracy of 96.97%, whereas other models such as SA-GCN, GNNs, BiERU, MWGCN, and DCNN achieve accuracy values of 90.56%, 91.48%, 92.23%, 93.99%, and 95.55%, respectively. Similarly, the SRGANs-BERT model's accuracy for the Laptop14 dataset is 97.12%, whereas the accuracy values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 93.45%, 94.78%, 93.17%, 95.87%, and 96.13%. Also, the SRGANs-BERT model's accuracy for the Twitter dataset is 96.99%, whereas the accuracy values for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models are 90.34%, 90.04%, 91.98%, 91.16%, and 92.54%. Like this, the suggested SRGANs-BERT model has an accuracy of 99.99% under the MAMS dataset, compared to 91.67%, 92.97%, 93.49%, 95.98%, and 96.48% for the SA-GCN, GNNs, BiERU, MWGCN, and DCNN models, respectively.

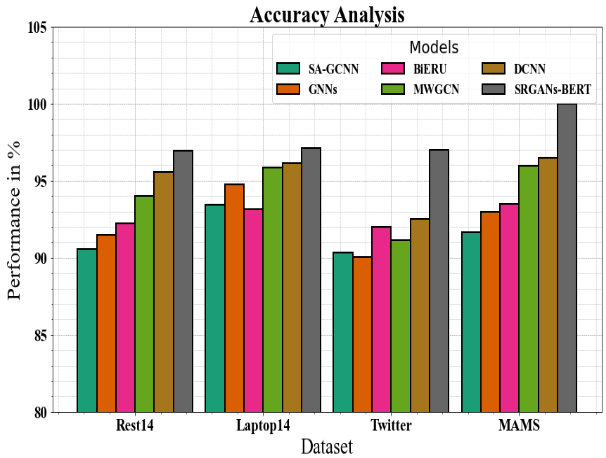

Fig.8. Accuracy Analysis for SRGANs-BERT method with existing systems

5. Conclusion

In conclusion, the proposed framework for Aspect-based Sentiment Analysis—which combines an SRGAN with a BERT—is a novel and promising approach to solving the issues with sentiment analysis. We have investigated the power of deep learning algorithms in solving the complex task of identifying sentiment nuances within specific features of text throughout this work. Combining SRGAN and BERT creates a comprehensive model that gathers fine-grained sentiment data and raises the bar for aspect-based sentiment analysis. It combines picture super-resolution and contextual language understanding abilities. The model displays heightened sensitivity to small sentiment shifts associated with diverse aspects by creating high-resolution representations of textual data and utilizing BERT's contextualized embeddings, contributing to more accurate and nuanced sentiment analysis. The suggested framework SRGANs-BERT exhibits outstanding performance with the following experimental results: Macro - F1 value of the Rest14 dataset is 95.99%, the Laptop14 dataset is 95.26%, the Twitter dataset is 94.33%, and the MAMS dataset is 95.78%. The precision value of the Rest14 dataset is 92.56%, the Laptop14 dataset is 91.78%, the Twitter dataset is 90.99%, and the MAMS dataset is 94.98%. The recall value of the Rest14 dataset is 86.87%, the Laptop14 dataset is 87.66%, the Twitter dataset is 85.55%, and the MAMS dataset is 89.32%. The F1-Score value of the Rest14 dataset is 80.98%, the Laptop14 dataset is 79.34%, the Twitter dataset is 78.81%, and the MAMS dataset is 82.91%. The accuracy value of the Rest14 dataset is 96.97%, the Laptop14 dataset is 97.12%, the Twitter dataset is 96.99%, and the MAMS dataset is 99.99%.

In the future, we'll need more reliable and realistic measurements to assess model performance. Another more effective method is replacing conicity similarity with other statistical metrics to get a sharp fall in the loss function of explainability between different linguistic neurons. Furthermore, more than explainability based on linguistic norms is required for corpora with ambiguous linguistic rules. Our upcoming study uses neuro-symbolic AI to perform understandable sentiment analysis.

References

[1] J. Li, B. Chiu, S. Shang, and L. Shao, "Neural text segmentation and its application to sentiment analysis", *IEEE Trans. Knowl. Data Eng.* 34, 828–842, 2022.

[2] C. Chen, Z. Teng, Z. Wang, and Y. Zhang, 2022. "Discrete opinion tree induction for aspect-based sentiment analysis," *in: ACL, Association for Computational Linguistics.* pp. 2051–2064.

[3] R. Li, H. Chen, F. Feng, Z. Ma, X. Wang, and E. H. Hovy, "Dual graph convolutional networks for aspect-based sentiment analysis", *in: ACL, Association for Computational Linguistics*. pp. 6319–6329, 2021.

[4] H. H. Do, P. W. Prasad, A. Maag, and A. Alsadoon, "Deep learning for aspect-based sentiment analysis: a comparative review", *Expert systems with applications* 118, 272–299, 2019.

[5] D. Li, K. Ahmed, Z. Zheng, S. A. H. Mohsan, M. H. Alsharif, M. Hadjouni, and S. M. Mostafa, "Roman Urdu Sentiment Analysis Using Transfer Learning," *Appl. Sci.* 12 (20), 10344, 2022.

[6] H. Zhao, Z. Liu, X. Yao, and Q. Yang, "A machine learning-based sentiment analysis of online product reviews with a novel term weighting and feature selection approach", *Informat. Process. Manage*. 58 (5), 102656, 2021.

[7] H. Tang, D. Ji, C. Li, and Q. Zhou, "Dependency graph enhanced dual transformer structure for aspect-based sentiment classification", *In Proceedings of the 58th Annual meeting of the association for computational linguistics*, pp. 6578–6588, 2021.

[8] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, Albert: "A lite bert for self-supervised learning of language representations", *arXiv preprint*, 2019 arXiv:1909.11942.

[9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen et al. "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint*, 2019, arXiv:1907.11692.

[10] A. Žunić, P. Corcoran, and I. Spasić, "Aspect-based sentiment analysis with graph convolution over syntactic dependencies", *Artificial Intelligence in Medicine*, 119, Article 102138, 2021. http://dx.doi.org/10.1016/j.artmed.2021.102138

[11] C. Sun, L. Huang, and X. Qiu, "Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence," *In Proceedings of NAACL-HLT*, 2019, pp. 380–385.

[12] K. Zhang, L. Wu, G. Lv, M. Wang, E. Chen, and S. Ruan, "Making the relation matters: Relation of relation learning network for sentence semantic matching" *In Proceedings of the AAAI conference on artificial intelligence*, vol. 35 no. 16, pp. 14411–14419, 2021.

[13] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. R. Bowman, and N. A. Smith, "Annotation artifacts in natural language inference data", *In NAACL-HLT*, 2018, (2).

[14] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, and P. Wang, "K-bert: Enabling language representation with knowledge graph," *In: Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34, No. 03, pp. 2901–2908, 2020.

[15] A. Karimi, L. Rossi, and A. Prati, "Improving BERT performance for aspect-based sentiment analysis," *arXiv*:2010.11731, 2020.

[16] A. P. Ben Veyseh, N. Nouri, F. Dernoncourt, Q. H. Tran, D. Dou, and T. H. Nguyen, "Improving aspect-based sentiment analysis with gated graph convolutional networks and syntax-based regulation", *In: Find. Assoc. Comput. Linguist. Find. ACL EMNLP* 2020, pp 4543–4548, 2020.

[17] B. Yu, and S. Zhang, "A novel weight-oriented graph convolutional network for aspect-based sentiment analysis," *J. Supercomput*. 79, 947–972, 2023. https://doi.org/ 10.1007/s11227-022-04689-9.

[18] Y. Tian, G. Chen, and Y. Song, "Enhancing aspect-level sentiment analysis with word dependencies", *in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, 2021, pp. 3726–3739.

[19] M. Chen, and X. Zhao, "A multi-scale fusion framework for bimodal speech emotion recognition", *in: Interspeech*, 2020, pp. 374–378.

[20] E.G. Moung, C.C. Wooi, M.M. Sufian, C.K. On, and J.A. Dargham, "Ensemble-based face expression recognition approach for image sentiment analysis", *Int. J. Electr. Comput. Eng*. 12 (3) (2022) 2588.

[21] C. Arumugam, and K. Nallaperumal, "EIAASG: Emotional Intensive Adaptive Aspect-Specific GCN for sentiment classification," *Knowl.-Based Syst*. 260, 110149, 2023.

https://doi.org/10.1016/j.knosys.2022.110149.

[22] D. Priyasad, T. Fernando, S. Denman, S. Sridharan, and C. Fookes, "Attention driven fusion for multi-modal emotion recognition," *in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing",* ICASSP, IEEE, 2020, pp. 3227–3231.

[23] W. Li, W. Shao, S. Ji, and E. Cambria, "BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis", *Neurocomputing* 467, 2022, 73–82.

[24] K.L. Tan, C.P. Lee, K.S.M. Anbananthen, K.M. Lim, "RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network", *IEEE Access* 10, 2022, 21517–21525.

[25] X. Hou, J. Huang, G. Wang, P. Qi, X. He, B. Zhou, "Selective attention based graph convolutional networks for aspect-level sentiment classification", 83–93, 2021 arXiv:1910.10857.

[26] K. Wang, W. Shen, Y. Yang, X. Quan, and R. Wang, "Relational graph attention network for aspect-based sentiment analysis", 2020. *arXiv preprint arXiv:2004.12362*.

[27] K. Sun, R. Zhang, S. Mensah, Y. Mao, and X. Liu, "Aspect-level sentiment analysis via convolution over dependency tree", *In Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP),* 2019, pp. 5679-5688.

[28] H. Wu, C. Huang, and S. Deng, "Improving aspect-based sentiment analysis with Knowledge-aware Dependency Graph Network," *Information Fusion*, *92*, 289-299, 2023.

[29] C. Chen, Z. Teng, Z. Wang, and Y. Zhang, "Discrete opinion tree induction for aspect-based sentiment analysis," In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022,* pp. 2051-2064.

[30] Kumar, Ravindra, Husanbir Singh Pannu, and Avleen Kaur Malhi. "Aspect-based sentiment analysis using deep networks and stochastic optimization." *Neural Computing and Applications* 32 (2020): 3221-3235.

[31] C. H. Lin, and U. Nuha, "Sentiment analysis of Indonesian datasets based on a hybrid deep-learning strategy", *Journal of Big Data*, *10*(1), 88, 2023.

[32] K. Zeng, Z. Pan Y. Xu, and Y. Qu, "An ensemble learning strategy for eligibility criteria text classifcation for clinical trial recruitment: algorithm development and validation," *JMIR Med Inform*. 2020;8(7): e17832

[33] A. Mohan, A. M. Nair, B. Jayakumar, and S. Muraleedharan, "Sarcasm Detection Using Bidirectional Encoder Representations from Transformers and Graph Convolutional Networks". *Procedia Computer Science*, *218*, 93-102, 2023.

[34] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, *63*(11), pp.139-144, 2022.

[35] C. Ledig et al. "Photo-realistic single image super-resolution using a generative adversarial network," *CVPR*. 2017, 2(3):105– 114. https://doi.org/10.1109/CVPR.2017.19.

[36] B. Lim et al. "Enhanced deep residual networks for single image super-resolution," *In: Proceedings of the IEEE Conference on computer vision and pattern recognition workshops*. 2017

[37] Demiray, Z. Bekir, Muhammed Sit, and Ibrahim Demir. "D-SRGAN: DEM super-resolution with generative adversarial networks." *SN Computer Science* 2, 2021 1-11.

[38] J. Pennington, R. Socher, and C.D. "Manning, Glove: Global vectors for word representation", *in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543