Journal of Electronic & Information Systems | Volume 07 | Issue 02 | October 2025

Journal of Electronic & Information Systems
https://journals.bilpubgroup.com/index.php/jeis

ARTICLE

Sentiment Analysis of Short and Incomplete Texts Using Transformer-Based Attention Models

Reza Nouralizadeh Ganji [1], ***Nasim Tohidi*** [1,2*]

[1] *Artificial Intelligence Department, Faculty of Computer Engineering, K. N. Toosi University of Technology, Tehran 16317-14191, Iran*

[2] *Department of Computer Science, Wayne State University, Detroit, MI 48202, USA*

ABSTRACT

The sentiment is seen as valuable information that can represent people's opinions, and its analysis is regarded as an essential component of decision-making processes. Social media has led to an exponential increase in the volume of shared textual content, and natural language processing as a potential area provides a variety of cutting-edge, deep learning-based models for analyzing and understanding this content. However, short, incomplete, and noisy text containing misspellings and grammatical errors is prevalent in these postings, making sentiment analysis challenging. This paper proposes a sentiment polarity detection approach using a three-phased methodology for short and incomplete text. First, the model utilizes transformer-based mechanisms to automatically correct and complete texts, thereby eliminating the need for manual human annotation. In the second phase, denoising neural networks are learned to reconstruct representations of short and incomplete texts. In the third phase, outputs and trained weights from previous steps are used to predict the sentiment polarity of input text by applying the attention mechanism, convolution nets, and pooling layers. Experimental evaluations demonstrate that the proposed approach outperformed state-of-the-art models in sentiment classification. It achieved an F1 score of 89.96% on the Sentiment 140 dataset and 76.91% on the ACL 14 dataset, demonstrating that it excels in precise correction, effective learning, and accurate prediction.

Keywords: Sentiment Analysis; Natural Language Processing; Incomplete Texts; Transformers; Text; Deep Learning;

*CORRESPONDING AUTHOR:
Nasim Tohidi, Artificial Intelligence Department, Faculty of Computer Engineering, K. N. Toosi University of Technology, Tehran 16317-14191, Iran; Department of Computer Science, Wayne State University, Detroit, MI 48202, USA; Email: nasimtohidi@wayne.edu

ARTICLE INFO
Received: 11 August 2025 | Revised: 5 October 2025 | Accepted: 12 October 2025 | Published Online: 19 October 2025
DOI: https://doi.org/10.30564/jeis.v7i2.12431

CITATION
Ganji, R.N., Tohidi, N., 2025. Sentiment Analysis of Short and Incomplete Texts Using Transformer-Based Attention Models. Journal of Electronic & Information Systems. 7(2): 132–153. DOI: https://doi.org/10.30564/jeis.v7i2.12431

Social Media; Attention Mechanism

1. Introduction

The rapid and undeniable ascent of the Internet as a cutting-edge and influential communication technology has had far-reaching effects on how humans interact. As a consequence of these effects, attributes such as real-time data exchange and the overcoming of spatiotemporal constraints have manifested within internet-mediated platforms, exemplified by social networks [1] and recommender systems [2]. With the expansion of the Internet and the evolution of various social media platforms, including weblogs, microblogs, social networks, and collaborative knowledge repositories such as Wikipedia, these platforms have assumed a significant and dynamic role within their respective domains. Consequently, the significance of information dissemination and knowledge acquisition from peers has markedly escalated [3].

People's thoughts and opinions are among the most valuable and costly bits of information; we may learn about people's thoughts by analyzing their emotions in texts such as tweets, posts, comments, and chats [4]. As a result of its significance, Sentiment Analysis (SA) is one of the most actively explored areas in Natural Language Processing (NLP) [5]. It is a fascinating and essential field of study that aims to extract people's ideas, feelings, thoughts, attitudes, and emotions from written characters [6]. However, a fundamental obstacle poses a threat to such analyses, and that is the possibility of incompleteness when a key term is missing from a text or statement. Furthermore, if the word or phrase is not typed correctly, the text is incomplete. Short and incomplete texts are most typically created on sites where users are not compelled to write appropriately, use correct structure, or follow normal grammar conventions [5,7], such as in casual conversations, short messages, and texts published on social networks like Twitter (X). **Table 1** includes examples of short and incomplete texts.

Table 1. Examples of short and incomplete texts [8].

#	Text	Length
1	espn soccer net news ardiles backs maradona for world cup glory.	11
2	i'm presidential when i flow , yo it aint hard to tell that i ba rock the show obama self tablo.	21
3	iloveitwhen the lakers do they thang shoutout to all the laker haters.	12
4	i don care what nebody says , i love harry potter ahah.	12

One significant way for dealing with noisy data, such as short and incomplete texts, is to use a robust denoiser mechanism capable of taking incomplete text representations and reducing them to build correct representations appropriate for any text mining task [9]. On the other hand, four components are essential to discuss regarding this strategy. To begin, think about the process by which we convert textual documents into numerical representations. This has lately been performed by several powerful tools called transformers [10], such as BERT [11], RoBERTa [12], and others. Second, to appropriately design and train a supervised mechanism that reconstructs representations that correspond to incomplete texts, which could rely on the autoencoder family. Third, in order to accomplish precise classification, neural architectures use the sorted representation produced by the denoiser procedure. Fourth, in order to train the denoiser mechanism with gold supervision, it is necessary to obtain the complete form of each incomplete text that is presented in the intended dataset. Furthermore, this task is complicated by the fact that many datasets lack complete annotations. Obtaining them manually is challenging, time-consuming, expensive, and prone to human error.

In this paper, we provide a practical methodology that makes use of State-of-the-Art (SotA) techniques to address the issues of short and incomplete texts that arise throughout the SA process. To begin, we identify an inherent question concerning the datasets: Are complete forms of incomplete texts adequately annotated and prepared? Based on this question, we implement two approaches: if the aforementioned annotations are prepared, the task becomes simple, and the denoiser mechanism can be learned under supervision. If complete forms are not obtained, we devised a flexible phase to apply automatic correction, which is done in three steps using a variety of effective corrector approaches to produce complete forms of incomplete texts. Following that, the RoBERTa transformer is used to construct enriched representations of textual documents, and groups of artificial neurons are organized in an autoencoder formation for denoising. Finally, convolution and pooling layers are used in conjunction with an attention mechanism to accurately identify the sentiment polarity of texts. The contributions of this research are as follows:

- It performs SA more effectively than recent advanced models when dealing with short and incomplete texts, regardless of whether such texts have been annotated by human operators or not.
- It exploits an entirely automatic correction phase employing cutting-edge models, eliminating the requirement for human intervention in producing complete versions of incomplete texts.
- It utilizes stages that incorporate an appropriate combination and connection of transformer, denoising mechanism, attention mechanism, and convolution layers to carry out sentiment evaluation on short and incomplete texts.

This paper continues with the following structure. Related works on the area of SA are discussed in Section 2. Our proposed approach is described in detail in Section 3. In Sections 4 & 5, we discussed the methods of assessment used and the experimental results. Section 6 provides the conclusion and future work.

2. Related Works

Since then, many methods have been presented for identifying the underlying emotional tenor of a given piece of writing [13,14]. This section looks into some of the most influential papers that have provided fresh ways to construct word embeddings, build representations, develop model architectures, and apply attention mechanisms. The fundamental structure of transformers was first presented by Vaswani et al. [10] in 2017. Their transformer construction included both an encoder and a decoder. The encoder integrated into that model was made up of numerous layers, each of which contained processing blocks such as multi-head attention techniques and neural networks. There are other computational processing blocks like normalization and residual connections. The decoder can be thought of as having the same design as the encoder, with the addition of some kind of attention mechanism where attention weights are calculated by comparing the encoder outputs with those from earlier time steps. Language model researchers were the first enthusiastic explorers of transformers due to their remarkable capacity to generate rich, high-dimensional representations of texts and the words included within them. They were able to achieve considerable advances in NLP and SA research by utilizing the encoder portion of transformers and transfer learning [15]. Many studies and new architectures have been developed in this area in recent years [16]; some of the most notable are BERT, RoBERTa, T5, etc.

According to Shuang et al.'s [17] research, employing the information buried in the roles of Part-Of-Speech (POS) tags can be viewed as a viable technique for analyzing emotion. In the emotional cast of a text, words that perform the roles of adjectives, adverbs, and nouns are more effective actors. Consequently, by embedding the information pertaining to the roles of each POS tag, researchers recognized them as a significant factor in their model, and by applying gates, they refined these POS roles, which, along with the vectors pertaining to the words themselves, allowed them to make an accurate prediction of the polarity. Liu and Shen [18] have taken a novel approach to word embedding vectors and viewed polysemy as a significant difficulty in SA. For example, the word *LIGHT* transmits a good emotional polarity when paired with the target aspect *PHONE*, but a negative emotional polarity when paired with the target aspect *CAR*. Using a sequential neural network to modify the initial embedding vector of the words was their primary strategy for addressing this fundamental difficulty. Naseem et al. [19] regarded the occurrence of short texts as a fundamental issue in NLP and emotional analysis. Creating word embedding vectors using multiple methods, including transformers, was their primary strategy for addressing this fundamental difficulty. First, they employed dynamic programming to identify basic misspellings in each word. As soon as they were corrected, they provided enriched embedding vectors for the characters, POS tags, words, and the whole sentence. Then, they combined these vectors to create a single vector, which they utilized throughout the remainder of the work until they attained emotional polarity.

Another interesting and useful piece of knowledge in SA is common sense and syntax knowledge, which Zhou et al.[20] believed would be useful and desirable to include this knowledge in sentence representation, with the main innovation of their concept in the graph representation format for common sense and syntax information. In fact, the representation of each sentence is determined in such a way that each word and its associated entity are regarded as nodes in the set of common sense. In addition, if two nodes of the sentence and the set of common sense are connected, or if there is a syntactic connection between two nodes (words) within a phrase, there is an edge between them in the adjacency matrix. The determination of emotional polarity for multiple words was deemed a worthy problem by Liao et al.[21]. They mentioned that in order to determine the emotional polarity of a target aspect in a sentence, it is necessary to have an accurate comprehension of all target aspect words. Constructing word embedding vectors and providing their representations facilitates this accurate comprehension. However, if this element consists of multiple linguistic units, such as battery life, this challenge should be managed well. It can be noted that an important part of the initiative of their research is the appropriate use of the RoBERTa language model, which is regarded as a variation of the well-known BERT model, and its advantages include the ability to construct more accurate representations than prior models and being trained on a huge number of distinct corpora. After creating representations of the input texts with the aid of the attention mechanism and convolutional networks, the classification of sentiment would be completed.

By examining recurrent neural networks such as Long Short-Term Memory (LSTM)[22] and Gated Recurrent Units (GRU)[23], Basiri et al.[24] discovered that these types of networks can discover and extract relatively long dependencies in sentences, and by researching convolutional neural networks, they concluded that they can discover local features effectively. For this reason, they proposed a model that incorporates both types so that it can reap the benefits of both simultaneously. Song et al.[25] investigated the architecture based on the BERT transformer and discovered that the output of the BERT middle layers can contain relevant and helpful knowledge from various sentence levels. As a result, they devised a system in which all of the outputs of twelve different BERT layers were collected and then aggregated to generate a specific vector. After executing this aggregation, it was feasible to execute multiple processes, such as the attention mechanism, which enhanced the quality of identifying the polarity of the emotion.

Sergio and Lee[9] viewed incomplete texts as a major hindrance in SA; sentences encountered on social media platforms like Twitter. In addition, two factors emphasize the significance of paying attention to incomplete and noisy sentences. First, people who are regarded as sources of data generation in social media are not obligated to write complete sentences properly. Second, responsive chatbot tools frequently make errors when converting user voices to text, resulting in noisy and incomplete phrases. Their project, entitled "Stacked DeBERT", aimed to train a succession of neural networks to detect incomplete sentences in order to overcome this challenging obstacle. This technique empowers their model to reconstruct the representation of word embedding vectors in incomplete sentences. Once the representation of incomplete sentences reaches the level of complete sentences, thanks to the ability to rebuild them, the SA of incomplete data could be done with the same level of precision as complete sentences.

Mutinda et al.[26] studied the feasibility of using transformers, such as BERT, as trustworthy tools for handling short texts that are unstructured, semi-structured, and typically rich in colloquial language. In their research, they used a combination of the sentiment lexicon, N-grams, and the BERT word embedding model to vectorize words from a section of the input text. The sentiment lexicon is used to find the part of an input short text where sentiment information is located, the BERT algorithm is used to build word vectors from that section only, and a Convolutional Neural Network (CNN) is used as the deep neural network classifier for feature mapping and giving the output predictions.

The content-derived features extracted via diverse machine learning approaches are valued by Mirugwe et al[27]. Their research investigates the application of deep learning techniques for SA of Twitter data pertaining to the Ebola outbreak in Uganda. The study comparatively evaluates the efficacy of three prominent deep learning models: CNN, LSTM networks, and BERT. Notably, the BERT model demonstrated superior performance, attaining a significant 95% accuracy in sentiment classification. The results demonstrate the importance of public sentiment analysis in developing an effective strategy of communication and for public health intervention, as described in the paper. Further supporting the need for deeper understanding, the study also shows the sentiment distribution within the analyzed tweet corpus, revealing a predominance of neutral sentiment, followed by positive and then negative sentiments, respectively, illustrating the multi-faceted nature of public response during such crises.

Although aforementioned studies using deep learning and attention mechanisms have shown considerable success in applying sentiment analysis, the large majority of current transformer based models (i.e., BERT and RoBERTa) that have been developed for this purpose struggle with processing the short and incomplete texts of most social media postings; since these models depend largely upon having well-defined syntactic relationships to accurately represent the input data, their performance typically suffers significantly when input is noisy or otherwise does not include all necessary dependencies. The Stacked DeBERT [9] model has introduced some form of denoising to reconstruct partially empty sentences from noisy inputs; however, it incurs extremely high levels of computational complexity through using multiple transformers. Additionally, these existing frameworks generally lack a robust, fully automated preprocessing phase capable of rectifying severe spelling and grammatical errors, often necessitating a reliance on human-annotated data, which is scarce and expensive to obtain.

Since, among previously related works, stacked DeBERT [9] appeared to be more similar to our proposed model, this provides an opportunity to address some important differences between the two studies that could be considered discriminatory foundations.

- Our proposed model takes advantage of a completely automatic correction phase with some of the SotA models that removes the need for human agents to generate complete versions of incomplete texts.
- Our proposed model utilizes only one transformer model in the sentiment classification phase, whereas two transformer models are used in stacked DeBERT, which causes more complexity.
- Our proposed model utilizes convolution and pooling layers of various sizes, an attention mechanism with special inputs in the sentiment classification phase, and a denoiser mechanism with skip connections, batch normalizations, and different structures in the denoising phase, which improves the polarity detection quality of the whole model.

3. The Proposed Model

Three general phases of our proposed model are depicted in **Figure 1**. The output of the previous phases, which may be a collection of texts or a trained model, is used as an input or specific mechanism in the subsequent phases. In the initial phase, various correction methods are automatically applied to the incomplete text data without the annotation of the complete form. The complete form of the incomplete data is obtained at the end of this phase. The second phase is the supervised training of denoiser neural networks. The denoiser network is trained in such a way that it can reconstruct the representation of incomplete texts and produce more complete representations by having access to both incomplete texts and their complete forms. In the third phase, the sentiment polarity detection of the input texts is determined.

3.1. Data Cleaning

The processes involved in the proposed model are broken down into three distinct phases after the input texts have been cleaned. A few common challenges, such as the presence of links, emojis, punctuation marks, etc., are not taken into account in the proposed model because the datasets used have already been cleaned. On the other hand, complete forms of incomplete texts created by human agents who were fluent in English do not require cleaning and can be transferred to the next step. Therefore, all samples from both used datasets have been refined by following distinct cleaning rules.

- The third or more consecutive repetitions of any word should be deleted. For example, “very very very very good”.
- The third or more consecutive repetitions of any character should be deleted. For example, “Sleeeeeeeeeep”.
- All non-ASCII characters in the text should be removed. For example, “, ¿, ®, ×, ←, †”
- All single and double quotation marks, which are null, should be removed. For example, “ “_”, ‘_’,“”,“ ”
- The space or half-space before and after the apostrophe

character should be removed if it does not interfere with the next word. For example, "dog'_s head, cat_'s eyes, can_'_t like"

- The apostrophe character for contracted expressions should be inserted in the correct place. For example, "Im, Youre, Cant,"

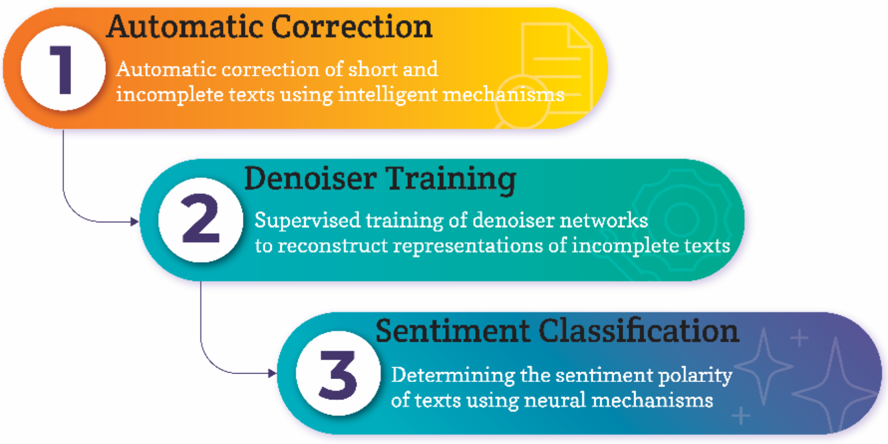


Figure 1. Three general phases of the proposed model.

3.2. Phase 1: Automatic Correction

During the phase of implementing automatic corrections, different correction mechanisms automatically rewrite incomplete textual data whose complete form has not been rewritten by human agents. In total, three correction mechanisms perform correction processes in three steps, and the aforementioned data cleaning processes (Section 3.1) are implemented at the beginning of each step to correct potential errors in the input and output streams of each mechanism. The correction procedures can be categorized into two groups: those pertaining to word spelling and those pertaining to grammatical structures. The most significant advantage of all applied correction mechanisms is that they correct each word based on its context in the text. **Figure 2** details the hierarchical cascade of the automatic correction phase, in which the Ginger-BERT, ByT5, and ELMo-LSTM modules are sequentially applied to rectify spelling, punctuation, and syntactic errors in incomplete texts.

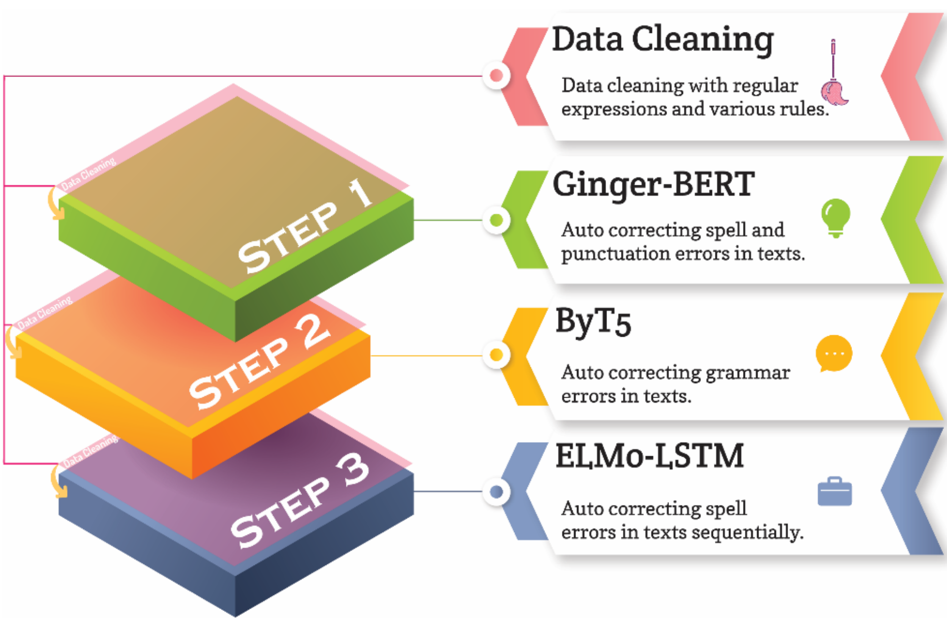


Figure 2. The three steps regarding the automatic corrections phase for incomplete texts.

In the first step of the automatic correction phase, Ginger was used to process and correct the cleaned texts. Ginger, on the other hand, can process the text at a high general level, including punctuation, but it lacks the ability to deeply understand each word in order to make changes to it. As a result, we turned to the NeuSpell tool [28] and the trained transformer-based model, BERT, which produces high-dimensional representations for each word, and can achieve a proper understanding of the word and its corresponding textual context in order to eliminate spelling errors. In the second step, we used ByT5 [29] to correct incomplete texts from a syntactic perspective. Finally, in the third step, the ELMo-LSTM model from the NeuSpell tool receives the cleaned data from the previous step to apply both spelling and grammar corrections. After checking and making the necessary changes, it produces its modified form of input textual documents, and at the end of this step, the desired constructed data is obtained, which can be regarded as the completed and corrected version of incomplete texts and applied to subsequent processing steps.

3.3. Phase 2: Denoiser Training

The proposed model architecture for training denoiser networks consists of two important parts: the first part produces a vector representation of received texts, and the second part performs the fundamental role of denoising. In the proposed model, a base version of the RoBERTa transformer model with 12 encoder layers was used to produce the vector representation of incomplete data and the completed form of those texts, and each transformer layer has an output with the shape (N_{bs}, 768, L_x). N_{bs} means the batch size, and L_x also indicates the fixed length of the input text sequence. The output related to the special token named [CLS] is used in the last layer of RoBERTa in the proposed model, that is $H_{CLS}^{12} \in \mathrm{R}^{768 \times 1}$. For simplicity, the output of H_{CLS}^{12} incomplete text is named H_{inc}, and the output of H_{CLS}^{12} complete texts is also named H_{com} in this paper. Then, the RoBERTa model is used to create short and incomplete text representations. After that, the proposed model uses a stack of 13 neural layers in the formation of autoencoder networks for denoising, so each layer is made of a different number of neurons. Each layer is fully connected to the subsequent layer, with forward connections that have trainable weights. These 13 layers of neurons could be classified into 4 groups, and each group can be considered as MultiLayer Perceptron (MLP) network. **Figure 3** shows the autoencoder-based training architecture of the denoiser networks, highlighting the dimensionality reduction and reconstruction process supervised by the Mean Squared Error (MSE) loss between incomplete and complete text representations. The first two groups are responsible for reducing the dimension of the vectors' representation of the incomplete texts to obtain more abstract representations. By placing an appropriate number of neurons in successive layers, they compress the received representations, calculate the representation of each piece of textual data in the reduced space, and extract more abstract features from their inputs.

After that, two other groups of MLP networks come into action and first receive the abstract vectors representation produced by previous layers and start the reconstruction operation. They increase the dimension of the received vectors, layer by layer, until the vectors reach dimensions equal to the RoBERTa output. Equations (1) and (2) formally define the forward propagation through the MLP blocks of the denoiser, where H_{inc} means the incomplete text representations produced by RoBERTa and $W_{l_i} \in \mathrm{R}^{d_{l_{i-1}} \times d_{l_i}}$ and $b_{l_i} \in \mathrm{R}^{1 \times d_{l_i}}$ also represent the weight and bias matrix of the i^th layer, respectively. Activation functions are also indicated by f and f, which represent the ReLU function and the TanH function, respectively.

$$\begin{aligned} O_{g_1} &= f_\alpha(W_{l_3} f_\alpha(W_{l_2}(f_\alpha(W_{l_1}(H_{inc}) + b_{l_1})) \\ &\qquad + b_{l_2}) + b_{l_3}) \\ O_{g_2} &= f_\alpha(W_{l_6} f_\alpha(W_{l_5}(f_\alpha(W_{l_4}(O_{g_1}) + b_{l_4})) \\ &\qquad + b_{l_5}) + b_{l_6}) \end{aligned} \tag{1}$$

$$\begin{aligned} O_{g_3} &= f_\alpha(W_{l_9} f_\alpha(W_{l_8}(f_\alpha(W_{l_7}(O_{g_2}) + b_{l_7})) \\ &\qquad + b_{l_8}) + b_{l_9}) \\ O_{g_4} &= f_\beta(W_{l_{13}} f_\alpha(W_{l_{12}} f_\alpha(W_{l_{11}}(f_\alpha(W_{l_{10}}(O_{g_1}) \\ &\qquad + b_{l_{10}})) + b_{l_{11}}) + b_{l_{12}}) + b_{l_{13}}) \end{aligned} \tag{2}$$

Now, having the output of the last neural layer of the denoiser (i.e., $O_{g_4} \in \mathrm{R}^{768 \times 1}$) and the output of RoBERTa corresponding to the completed text data (i.e., $H_{com} \in \mathrm{R}^{768 \times 1}$), the proposed model, for the supervised training of denoiser networks, uses the MSE measure in Equation (3). In fact, the second couple of groups learns how to reconstruct the abstract representations of the incomplete texts, which is the result of the work of the first couple of groups, to the level of the representations related to the complete textual data.

Also, to induce better generalization power and higher quality training, batch normalization is applied before entering the first neural layer, and residual connections are designed between the four groups. **Algorithm 1** shows the supervised training procedure in which an autoencoder minimizes the MSE between RoBERTa-extracted incomplete and complete text representations to restore semantic integrity.

$$\mathcal{L}\left(O_{g_4}, H_{com}\right) = \frac{1}{N_{bs}} \sum_{k=1}^{N_{bs}} \left(O_{g_4} - H_{com}\right)^2 \tag{3}$$

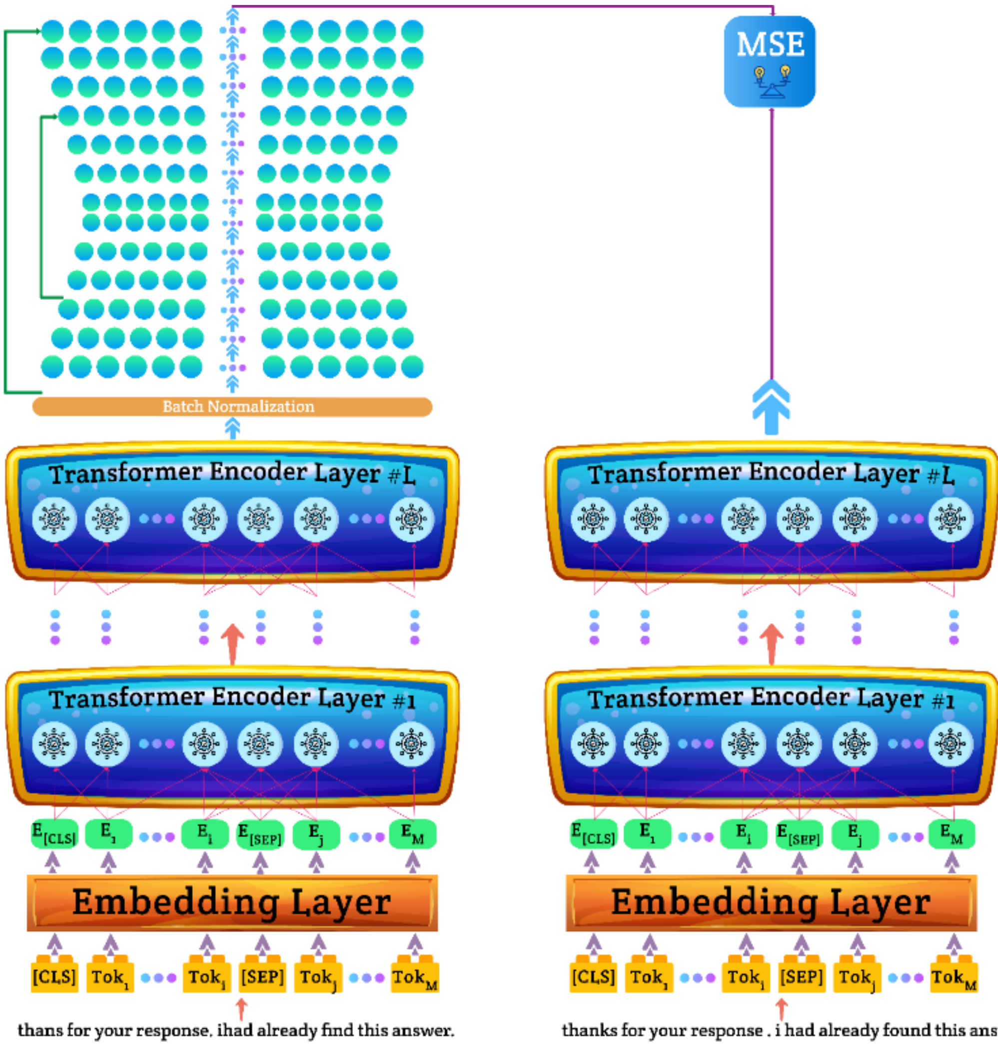


Figure 3. The denoiser network training architecture, depicting the autoencoder-based compression (dimensionality reduction) and reconstruction of incomplete text representations, supervised by the MSE loss between the predicted and complete vector representations.

Algorithm 1. Supervised Training of the Denoiser Mechanism.

Input: D_{batch}: Batch of paired texts $\left\{T_{inc}^{(i)},\ T_{com}^{(i)}\right\}_{i=1}^{N_{bs}}$, $RoBERTa(\cdot)$: Pre-trained Transformer Encoder, $f_{Enc}(\cdot)$: Encoder MLP layers (Groups 1 & 2 – Compression), $f_{Dec}(\cdot)$: Decoder MLP layers (Groups 3 & 4 – Reconstruction).
Output: Optimized parameters $\theta_{denoiser}$.

initialize weights for f_{Enc} and f_{Dec}
for each epoch **do:**
  for each batch D_{batch} **do:**
    // Step 1: Generate Representations using $RoBERTa$
    $H_{inc} \leftarrow RoBERTa\left(T_{inc}\right)$ // Representation of incomplete text

$H_{com} \leftarrow RoBERTa\,(T_{com})$ // Target representation of complete text
// Step 2: Forward Pass (Denoising Autoencoder)
$Z_{latent} \leftarrow f_{Enc}\,(H_{inc})$ // Compress to abstract features (Equation (1))
$H_{reconstructed} \leftarrow f_{Dec}\,(Z_{latent})$ // Reconstruct to original dimension (Equation (2))
// Step 3: Compute Loss and Update
$Loss_{latent} \leftarrow Loss\,(H_{reconstructed}, H_{com})$ // Mean Squared Error (Equation (3))
Update $\theta_{denoiser}$ via Backpropagation to minimize $Loss$
end for
end for
return trained model

3.4. Phase 3: Sentiment Classification

The suggested system injects the input texts into the RoBERTa model and produces rich vector representations of them by passing them via 12 consecutive transformer layers. During sentiment classification, the outputs from RoBERTa serve as inputs for two crucial mechanisms: first, the denoiser mechanism receives the output of the final layer of RoBERTa, and second, the sentiment classification mechanism receives a portion of its input from the output of the intermediate layers of RoBERTa. The output of the denoiser mechanism is regarded as crucial information for determining the emotional polarity and is regarded as an input for the sentiment identification procedure. Specifically, the denoiser mechanism is inserted between the RoBERTa model and the sentiment classifier. To determine the polarity of texts, the sentiment classification mechanism receives two crucial inputs: first, the output of the denoiser mechanism, which is the reconstructed form of incomplete text representations, and second, the output of the attention mechanism, which is produced from the output representations of both the intermediate and final layers of RoBERTa. After receiving the two inputs, they are combined using convolutional networks and different pooling layers before being passed to MLP networks to determine the sentiment polarity. In **Figure 4**, the integrated sentiment classification procedure is presented, and the fusion of the RoBERTa-based attention mechanism and the reconstructed denoiser outputs via parallel 1D CNN and pooling layers is demonstrated.

The attention mechanism has been applied to the 12 output layers of RoBERTa. As a result of this design, the significance of each intermediate and final layer output is identified, and their information is combined while being multiplied by the attention mechanism computed weight. This aggregation of representations from intermediate layers could be particularly effective and efficient when generating the representations for short texts by processing many pieces of information from distinct transformer layers. Therefore, the attention mechanism, with a proper distribution of attention weights across the representations obtained from each layer of the RoBERTa transformer, can generate a unified and aggregated representation vector containing useful information for sentiment classification. The result of this type of attention mechanism is one of the two inputs to the proposed sentiment classifier. Since in transformers such as BERT or RoBERTa, the aggregation of representations of all components of the input text is found in the output corresponding to a special token labeled [CLS], in our proposed model, by collecting all outputs of the twelve layers (L = 12) in $H_{inc}=\left\{H^1_{CLS}, H^2_{CLS}, \ldots, H^L_{CLS}\right\}$ attention weights are calculated with two learnable parameters $W_h \in \mathrm{R}^{d_{Att}\times 768}$ and $q \in \mathrm{R}^{1\times 768}$. Therefore, Equation (4) computes the aggregated attention vector (O_{Att}) by applying a SoftMax function over the transformer layers, utilizing a learnable query vector q and weight matrix W_h to determine the relative importance of each layer's output.

$$O_{Att} = W_h \left[softmax\left(q\,(H_{CLS})^T\right) H_{CLS}\right]^T \quad (4)$$

In the proposed model, the output of the denoiser networks (i.e., $O_{g_4} \in \mathrm{R}^{768\times 1}$), and the output of the attention mechanism (i.e., $O_{Att} \in \mathrm{R}^{d_{Att}\times 1}$), are separately fed into two parallel layers of the one-Dimensional Convolutional Neural Network (1D CNN) with different kernel sizes to retrieve informative local features and reduce the dimensionality. Each of these convolutional networks produces a vector with dimensions matching the network parameters from which it was produced. Two of the vectors correspond to the output of the denoiser mechanism, while the other two belong to the output of the attention mechanism. Then, two parallel and independent layers of average pooling and maximum pooling are used to manipulate and sample the initial space, as well as to process superior and discriminatory features. Consequently, each of the four outputs generated in the preceding phase is passed through these two pooling layers to generate two new vectors from each of the quadruple representations, resulting in a total of eight outputs. These eight produced vectors can be viewed as rich and summarized representations of short and incomplete input texts. In the proposed model, the eight vectors are concatenated and then normalized before being fed into the MLP networks. The ReLU activation function is used in the hidden layers of these consecutive networks, and as many neurons as the number of considered sentiment classes (i.e., K) are implanted in the output layer. After calculating the outputs of the fully connected MLP network (i.e., O_{FC}), as indicated in Equation (5), the SoftMax function () is used to generate probability distributions for each sentiment class. According to the Equation (6), the final output (i.e., $\widehat{y}$) of the model and its prediction for the emotional polarity of short and incomplete input texts is the polarity with the highest degree of probability.

$$\sigma\left(O_{FC}\right)_i = \frac{e^{O_{FC\,i}}}{\sum_{j\,=\,1}^{K} e^{O_{FC\,j}}}\ ,\ for\ j = 1, 2, \ldots, K \tag{5}$$

$$\widehat{y} = \underset{i\,\in\,1,\,2,\,\ldots,\,K}{argmax}\ \left\{\sigma\left(O_{FC}\right)_i\right\} \tag{6}$$

With the text and annotated sentiment labels from the dataset, we can train the entire integrated mechanism, including the aforementioned components and connections, in a supervised manner. In order to reduce the cross-entropy error, the derivatives of all learnable parameters were calculated and adjusted by using the backpropagation policy with respect to disparities between the model prediction (i.e., $\widehat{y}$) and the correct polarity (i.e., y). Equation (7) defines the total loss function for the sentiment classifier, minimizing the cross-entropy error between predicted and true labels while incorporating an $L2$ regularization term ($\lambda\left\|\theta\right\|^2$) to mitigate overfitting.

$$loss = -\sum_i \sum_j y_i^j log\widehat{y}_i^j + \lambda\left\|\theta\right\|^2 \tag{7}$$

Where i represents the number of the input text, j represents the number of the emotional class, λ represents the L2 regularization coefficient, and θ is the collection of all the learnable parameters addressed by the proposed model.

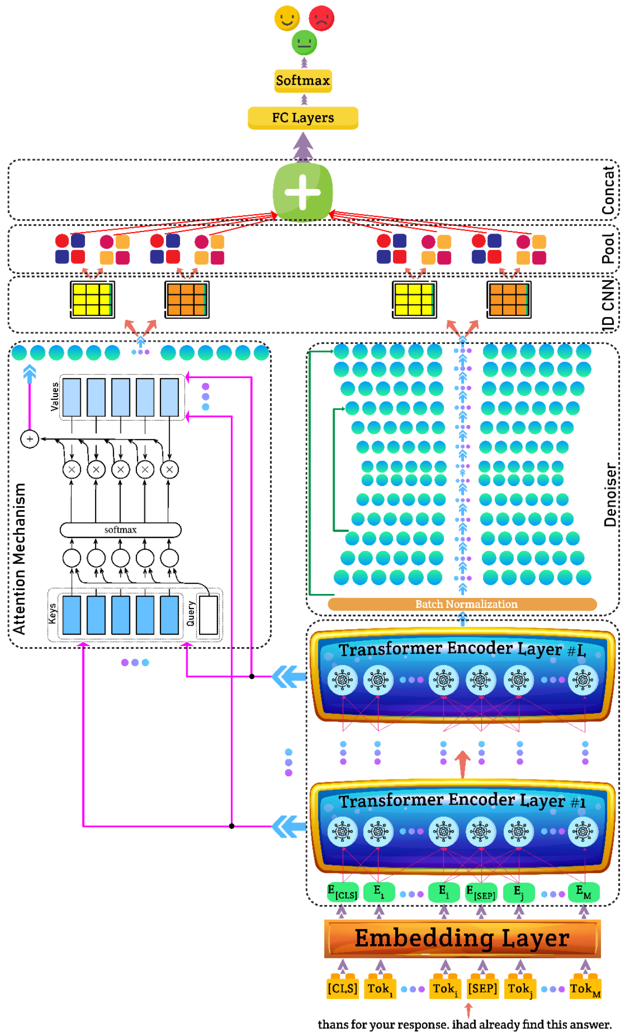


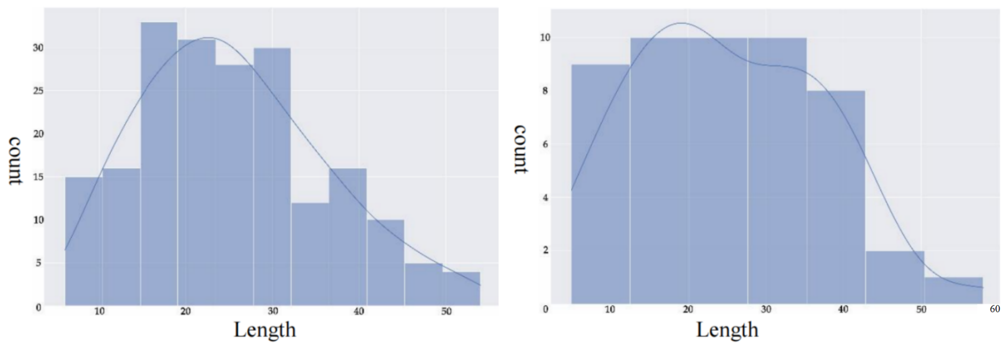
Figure 4. The sentiment classification architecture, detailing the feature fusion strategy where outputs from the RoBERTa-based attention mechanism and the denoiser network are concatenated and processed via parallel 1D CNN and pooling layers to determine polarity.

4. Evaluation Method

The proposed model has been evaluated using two popular datasets, Sentiment 140[9] and ACL 14[8], both of which contain English-language tweets and are known for detecting the polarity of sentiment at the document and aspect levels. For each short and incomplete textual data in Sentiment 140, the complete form and sentiment polarity are annotated, but in ACL 14, only the sentiment polarity is annotated. **Table 2** demonstrates the number of instances of each emotional polarity in the test and train sections, whereas in **Figures 5** and **6**, the probability density distribution of text string lengths for Sentiment 140 and ACL 14 datasets is visualized.

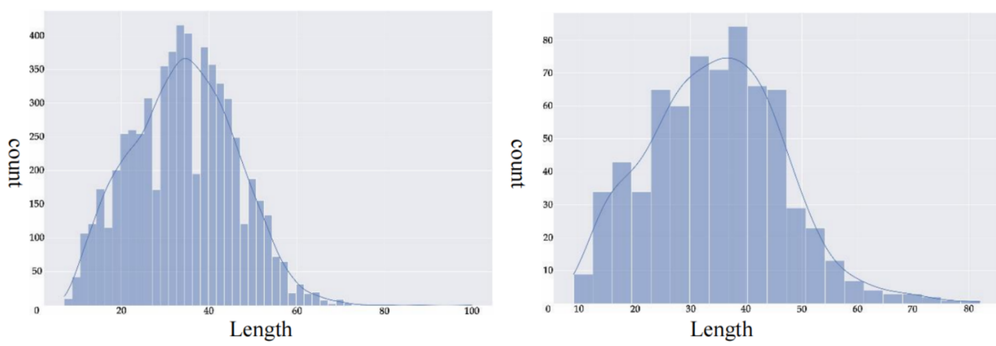
Table 2. Number of instances in Sentiment 140 and ACL 14 datasets by their sentiment polarities.

Polarity	Train Set		Test Set		Total	
	ACL 14	Sentiment 140	ACL 14	Sentiment 140	ACL 14	Sentiment 140
Positive	1561	100	173	25	1734	125
Neutral	3127	-	346	-	3473	-
Negative	1560	100	173	25	1733	125
Total	6248	200	692	50	6940	250



(**a**) (**b**)

Figure 5. Frequency of text string length in the Sentiment 140 dataset: (**a**) train set; (**b**) test set.



(**a**) (**b**)

Figure 6. Frequency of text string length in the ACL 14 dataset: (**a**) train set; (**b**) test set.

All components and elements of the proposed model have been implemented and executed using the Python programming language, and various neural network modules have been constructed and utilized with the Pytorch library. In addition, all transformers and their pre-trained weights have been compiled with the assistance of the Transformers library and the HuggingFace platform. Using the Nvidia Tesla K80 GPU, all units of the automatic correction phase were completed in the Google Colab environment. Using an Nvidia Tesla P100 GPU, training and testing of the denoiser processes and detecting the sentiment polarity were also performed in the Kaggle environment.

To facilitate reproducibility and ensure efficient convergence, the training phases were executed with carefully selected hyperparameters. The denoiser mechanism was trained using a batch size of 8, a learning rate of 2×10^{-5}, and a weight decay of 1×10^{-5}. Given the disparities in dataset sizes, the training duration was established at 2000 epochs for the Sentiment 140 dataset and 200 epochs for the ACL 14 dataset. Following this, the sentiment classification phase involved training the model for 14 epochs, maintaining a learning rate of 2×10^{-5} and a weight decay of 1×10^{-5}. Furthermore, the batch size was modified for this classification phase, set to 2 for the Sentiment 140 dataset and 16 for the ACL 14 dataset.

In the first phase of the proposed model, which is automatic correction, the assessment criterion is the similarity between the output texts of this phase and their human-annotated complete form. To calculate the similarity scores, we first fed both the text produced by the model and the text annotated by human agents into the T5 model to generate rich representations of the texts. Then, the outputs of the encoder portion of the T5 transformer[30] (which correspond to the produced text (i.e., R_{gen}) and the annotated text (i.e., R_{ann}), are acquired, and we calculate the cosine similarity measure (i.e., $\cos\left(R_{gen}, R_{ann}\right)$) to obtain the semantic similarity score of two texts.

In the second phase of the proposed model, which is the training of the denoiser mechanism, the evaluation criterion is the MSE value according to Equation (3).

The accuracy, precision, recall, and F1 criteria are employed in the third phase of the proposed model, which is the sentiment classification. These criteria are applied as formulated in the referenced study[31], respectively. For general analysis, weighted and macro versions of the aforementioned criteria are calculated as in this work[31].

5. Experimental Results

The proposed model is compared to those of the following baseline models, each of which has demonstrated good and high-quality performance on all of the intended datasets in recent years.

- **Sem-Hash**: Embedding vectors are constructed for each character rather than each word, and classifiers such as MLP[32], Support Vector Machines (SVM)[32], and Random Forests (RF)[32] are used for classification.
- **BERT-MLP**: BERT for generating representations and MLP classification are utilized[10].
- **BERT-Att**: BERT and an attention mechanism for generating and enriching representations, and MLP classification, are utilized[25].
- **RoBERTa-MLP**: RoBERTa for generating representations and MLP classification are utilized[12].
- **RoBERTa-Att**: RoBERTa and an attention mechanism for generating and enriching representations, and MLP classification are utilized[25].
- **S-DeBERT**: First BERT for generating representations, the denoiser mechanism for denoising, second BERT for generating representations of the denoiser outputs, and MLP classification are utilized[9].
- **LSTM**: LSTM for generating representations and MLP classification are utilized[22].
- **Bi-LSTM**: Bidirectional LSTM for generating representations and MLP classification are utilized[33].
- **IAN**: Two LSTM layers and an attention mechanism for generating and enriching representations, and MLP classification, are utilized[34].
- **T-MGAN**: A Transformer Encoder for word-level representations and a Tree Transformer Encoder for phrase-level context representations are utilized. A multi-grained attention network with a dual-pooling method is then employed for interactive feature extraction, followed by SoftMax classification[35].
- **SK-GCN**: BERT for generating representations, Graph Convolution Network (GCN) for modeling syntax and common sense, and MLP classification are utilized[20].
- **BERT-GNN**: BERT for generating representations, the Relational Graph Attention Network (RGAT) for encoding the reshaped and pruned dependency tree, Multi-Head Attention (MHA) for highlighting the important dependency relations, and MLP classification are utilized[36].
- **Llama 3.2**: The Llama 3.2, an auto-regressive language model with 1.23 billion parameters that uses an optimized transformer architecture, is utilized. This instruction-tuned text-only model is fine-tuned to perform sentiment classification on short and incomplete texts[37].

In two distinct scenarios, the performance of the proposed model is measured and compared to baseline models: in the first scenario, the complete form of the data is pre-annotated by human agents; in the second scenario, the complete form of the incomplete data is generated automatically by applying automatic correction policies. The primary objective for analyzing experimental results under two separate scenarios is to assess the suggested system in both contexts: datasets that are annotated and rectified by human operators (such as Sentiment 140) and datasets devoid of human-annotated data for incomplete texts (like ACL 14), prompting the automated correction stage to produce the complete version of incomplete texts.

5.1. First Scenario

As noted earlier, the presence of human agents in the repair operation of incomplete texts is considered in the first scenario; consequently, every finding in this scenario is derived from a human-annotated dataset (which is sentiment 140). **Figure 7** depicts the optimal training and convergence of denoiser networks, revealing a decreasing trend in loss for the proposed model throughout the first scenario. **Table 3** displays the evaluation of the proposed model sentiment classification for the first scenario of the Sentiment 140 dataset. Considering that Sentiment 140 may be seen as a small dataset for complex models such as transformers, we adopt a specific learning rate and L2 regularization approach during the training stages of denoising and classification to prevent overfitting.

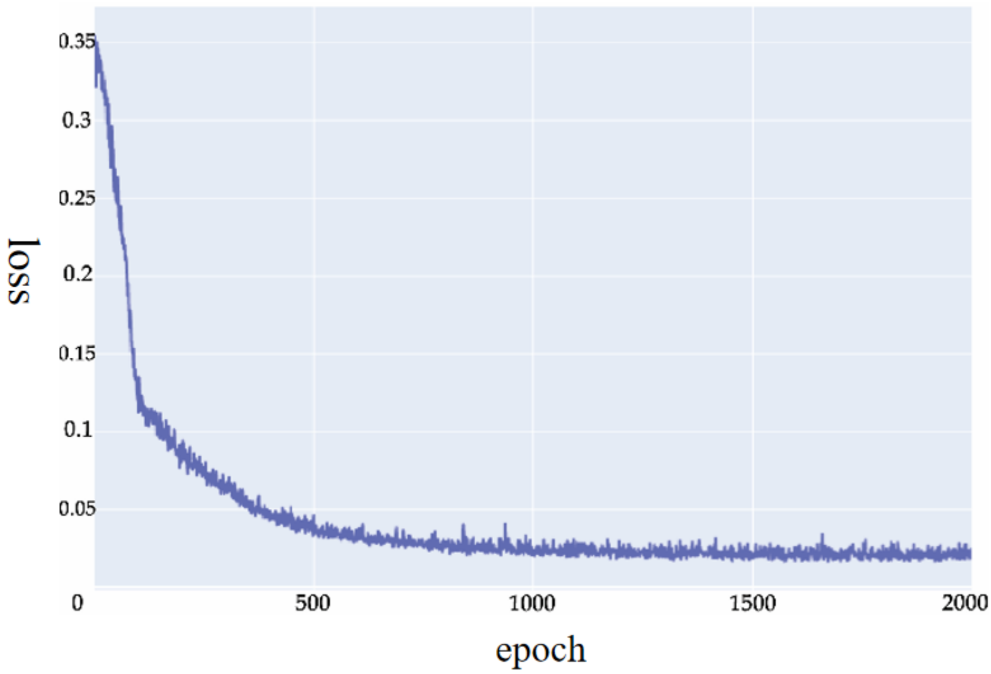


Figure 7. Convergence of denoiser networks for the first scenario.

Table 3. Evaluation of the proposed sentiment classification for the first scenario of the Sentiment 140.

		Precision (%)	Recall (%)	F1 (%)	# Instances
Polarity	negative	91.67	88.00	89.80	25
	Positive	88.46	92.00	90.20	25
Average	macro	90.06	90.00	89.99	50
	weighted	90.06	90.00	89.99	50

On the Sentiment 140 dataset, the comparison of the sentiment classification between several models for the first scenario is reported in **Table 4**. In comparison to baseline models, the proposed model achieved the highest accuracy rate among other models, and for the accuracy criterion, it increased by 10%, by obtaining an accuracy rate of 90%. The design of denoiser networks, the application of the attention mechanism on the intermediate layers of the transformer, and the proper integration of outputs with convolutional networks and pooling layers are the most significant reasons for the superiority of the proposed model over other baseline models. The S-DeBERT model is a competitive rival because it is the only baseline that also employs a denoiser process. An examination of the normalized confusion matrix (**Table 5**) reveals a nuanced comparison:

- For negative polarity, the models show comparable performance, with both the proposed model and S-DeBERT achieving an 88% correct prediction rate (0.88).
- For positive polarity, however, the proposed model demonstrates clear superiority, achieving a 92% cor-

rect prediction rate (0.92) compared to S-DeBERT's 72% (0.72). This translates to a significantly lower false-negative rate for positive texts (0.08 vs. 0.28).

This distinction confirms that the architecture's enhancements, namely the attention mechanism over intermediate layers and the specific design of the denoiser networks and their integration with convolutional layers, provide a substantial performance gain, particularly in classifying positive sentiment.

Table 4. Comparison of the sentiment classification between several models for the first scenario.

Model	Criteria	
	Accuracy (%)	# Instances
Sem-Hash	72	50
BERT-MLP [10]	72	50
BERT-Att [25]	76	50
RoBERTa-MLP [12]	80	50
RoBERTa-Att [25]	84	50
S-DeBERT [9]	80	50
Proposed Model	90	50

Table 5. Normalized confusion matrix of sentiment classification results of the proposed model and S-DeBERT.

Real Class	Predicted Class	Model	
		Proposed Model	S-DeBERT [9]
Positive	Positive	0.92	0.72
	Negative	0.08	0.28
Negative	Positive	0.12	0.12
	Negative	0.88	0.88

The suggested system comprises multiple elements that enhance its performance over baseline models. Nonetheless, it is beneficial to evaluate the impact of each element on sentiment classification performance. Hence, RoBERTa, coupled with MLP (RoBERTa-MLP[12]), intermediate attention mechanism (RoBERTa-Att[25]), the denoiser mechanism (RoBERTa-Denoiser), and ultimately all components, including convolution layers (viewed as the entire proposed model), are evaluated on the Sentiment 140 dataset. The results are demonstrated in **Table 6**.

Table 6. Comparison of the sentiment classification between the proposed model and models based on RoBERTa.

Model	Criteria		
	Accuracy (%)	F1 (%)	Testing Time (Millisecond)
RoBERTa-MLP [12]	80	79.87	362.18
RoBERTa-Att [25]	84	83.97	378.23
RoBERTa-Denoiser	86	85.99	744.55
Proposed Model	90	89.99	863.66

5.2. Second Scenario

As previously stated, the second scenario involves the exclusion of human agents from the process of correcting incomplete texts. In this scenario, the automatic correction phase is implemented on both datasets, namely sentiment 140 and ACL 14. Corrector models and strategies are utilized to generate the complete version of the incomplete texts. **Tables 7** and **8** display examples of automatically corrected texts for the two datasets, demonstrating the quality and precision of the correcting mechanisms in terms of fixing spelling, grammar, and punctuation.

Table 7. Incomplete and their automatically corrected texts for the sentiment 140 dataset in the proposed model.

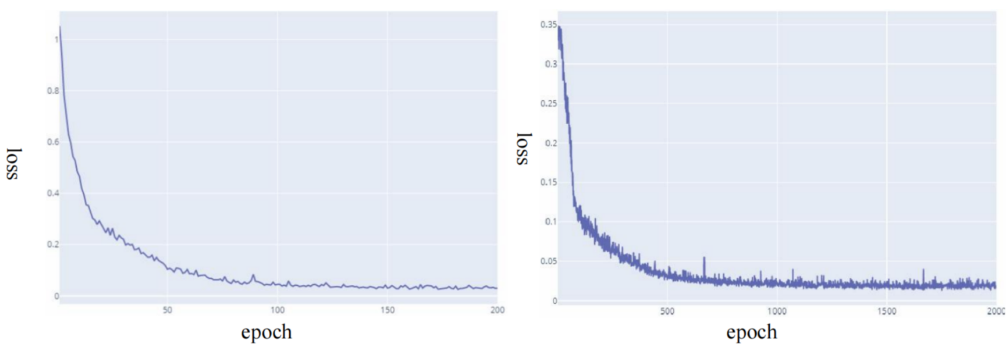
#	Incomplete Text	Complete Text
1	i baked u a cake but i ated it.	i baked you a cake, but i ate it.
2	thans for your response. ihad already find this answer.	thanks for your response. i had already found this answer.
3	will u ask donnie if he got the gift from my 9 yr old lastnight?.	will you ask donnie if he got the gift from my 9 year old last night ?.
4	me too i is poor.	me too, i am poor.
5	no he's still miss'n.	no, he's still missing.
6	i been out of range all day i'm back now and hopefully for good.	i have been out of range all day. i'm back now and hopefully for good.

Table 8. Incomplete and their automatically corrected texts for the ACL 14 dataset in proposed model.

#	Incomplete Text	Complete Text
1	i ca n't see it! i do n't know if you took it off though . . yeah ipod touch. in love.	i can not see it! i do not know if you took it off though yeah ipod touch. in love.
2	ipaymytaxesbecause with my luck i would n't wind up in a country club type prison like martha stewart or bernie madoff.	i pay my taxes because with my luck, i would not wind up in a country club type prison like martha stewart or bernie madoff.
3	i ca n't live without you, ca n't breathe without you. i dream about you, honestly tell me that it's over. over, lindsay lohan.	i can not live without you, can not breathe without you. i dream about you honestly tell me that it's over. over, lindsay lohan.
4	su techwraith dareyoudevil if bill gates wanted he could say, buddhists must die and hires a million police officers to kill them.	so technology dares your devil, if bill gates wanted, he could say buddhists must die and hire a million police officers to kill them.
5	honestly, i don like it. windows 7 beats vista by 234 in first week sales.	honestly, i do not like it. windows 7 beats vista by 234 in the first week sales.
6	i love britney spears don hate me.	i love britney spears and do not hate me.

Figure 8 depicts the decreasing trend of model loss in the stage of denoiser networks training for both datasets in the second scenario, demonstrating proper training and optimal convergence of denoiser networks by using auto-corrected texts. The denoiser network training convergence, depicted in **Figure 8**, demonstrates proper training with auto-corrected texts for both datasets in the second scenario. The significantly larger size of the ACL 14 training set (6248 instances) compared to the Sentiment 140 training set (200 instances) is the primary factor explaining the difference in convergence. Specifically, the denoiser networks for ACL 14 converged much faster and more smoothly (**Figure 8a**) than those for Sentiment140 (**Figure 8b**), due to the greater number of training samples, which facilitates more efficient gradient updates and a more generalized learning process.

For the second scenario, the evaluation of the proposed model classification of Sentiment 140 and ACL 14 datasets is shown in **Tables 9** and **10**, respectively.



(a) **(b)**

Figure 8. Decreasing trend of model loss in the stage of denoiser networks training for: (**a**) ACL 14; (**b**) Sentiment 140.

Table 9. Evaluation of the proposed sentiment classification for the second scenario of the Sentiment 140.

		Precision (%)	Recall (%)	F1 (%)	# Instances
Polarity	negative	85.71	96.00	90.57	25
	Positive	95.45	84.00	89.36	25
Average	macro	90.58	90.00	89.96	50
	weighted	90.58	90.00	89.96	50

Table 10. Evaluation of the proposed sentiment classification for the second scenario of the ACL 14.

		Precision (%)	Recall (%)	F1 (%)	# Instances
Polarity	negative	79.11	72.25	75.53	173
	neutral	77.81	86.13	81.76	346
Average	Positive	78.81	68.79	73.46	173
	macro	78.58	75.72	76.91	692
	weighted	78.38	78.32	78.12	692

The proposed model outperformed the baseline SA models in the second scenario in terms of accuracy and F1, with rates of 78.32% and 76.91%, respectively, and growth rates of 1.73% and 2.24%, respectively. The comparison of the sentiment classification between several models for the second scenario for the ACL 14 dataset is given in **Table 11**. While it is true that the baseline models are selected from the best and most well-known research and use a wide variety of architectures and ideas (such as syntactic knowledge, transformers, recurrent networks, etc.), the proposed model was able to obtain a suitable competitive edge with the help of an effective autocorrection ability and a combination of denoiser networks and attention mechanisms. In addition, the confusion matrix of the proposed sentiment classification results for the second scenario in the ACL 14 dataset (**Figure 9**) reveals the proper number of correct model predictions; this number is greater for neutral polarities than other polarities. This could be explained by the fact that there are more texts with neutral polarity than other polarities.

In addition, for more comprehensive evaluations, we compare the results of two scenarios to assess the efficiency of the models' outputs across all three phases. **Figure 10** illustrates the distribution of cosine similarity scores between automatically corrected texts and human-annotated references. By examining it, one can recognize the high similarity of the automatic correction outputs to the human agent generated texts and discover that there is an average of 90% similarity between the texts of these two groups, indicating that the outputs of the automatic correction phase of the proposed model are of sufficient quality.

Table 11. Comparison of the sentiment classification between several models for the second scenario on ACL 14 dataset.

Model	Criteria		
	Accuracy (%)	**F1 (%-Macro)**	**# Instances**
LSTM [22]	69.22	66.52	692
Bi-LSTM [33]	70.80	69.00	692
IAN [34]	71.82	69.11	692
BERT-MLP [10]	72.46	71.04	692
BERT-Att [25]	73.35	71.88	692
RoBERTa-MLP [12]	72.83	72.00	692
RoBERTa-Att [25]	71.39	71.00	692
T-MGAN [35]	71.23	70.63	692
SK-GCN [20]	75	73.01	692
BERT-GNN [36]	76.59	74.67	692
Llama 3.2 [37]	71.36	69.60	692
Proposed Model	78.32	76.91	692

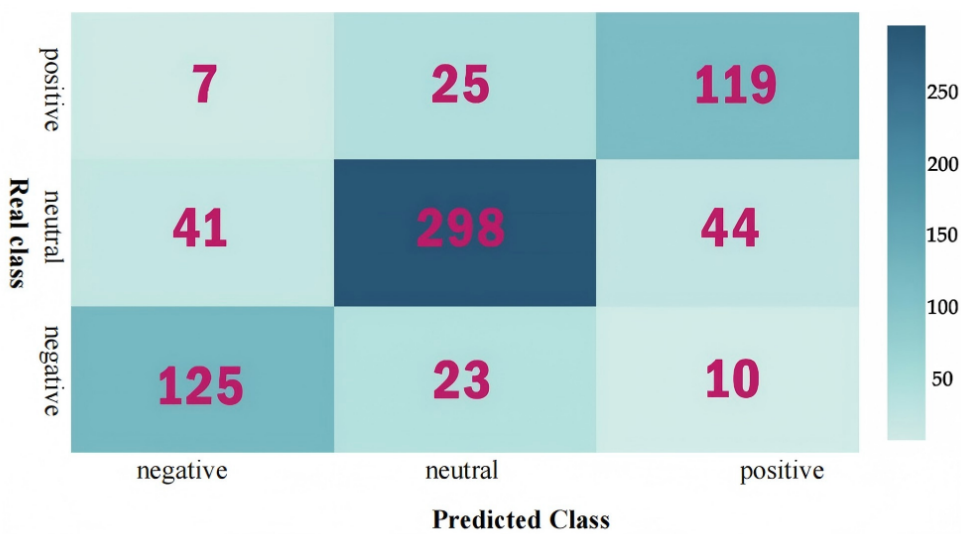
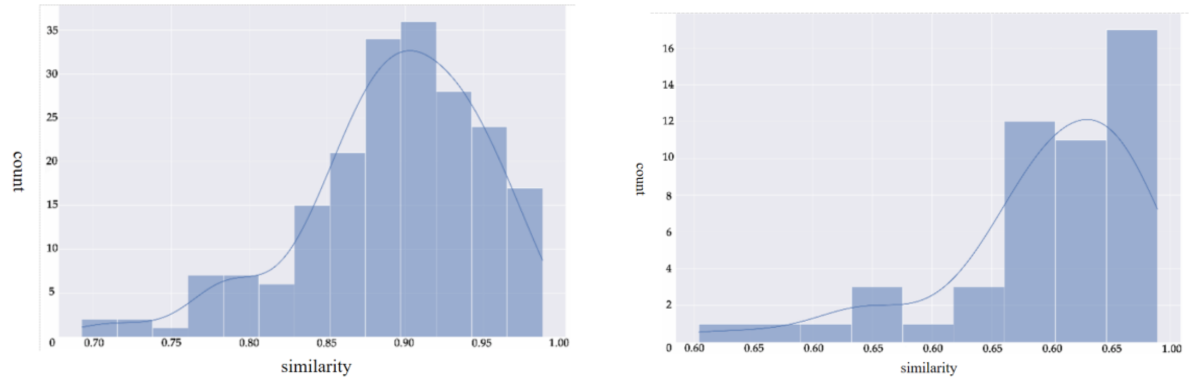


Figure 9. Confusion matrix for the ACL 14 dataset (Scenario 2), comparing the proposed model's predicted labels against ground-truth sentiments (Positive, Neutral, Negative) to highlight classification accuracy across different polarities.



(**a**) (**b**)

Figure 10. The distribution of the similarity measure between autocorrected and human generated texts for the Sentiment 140: (**a**) train set; (**b**) test set.

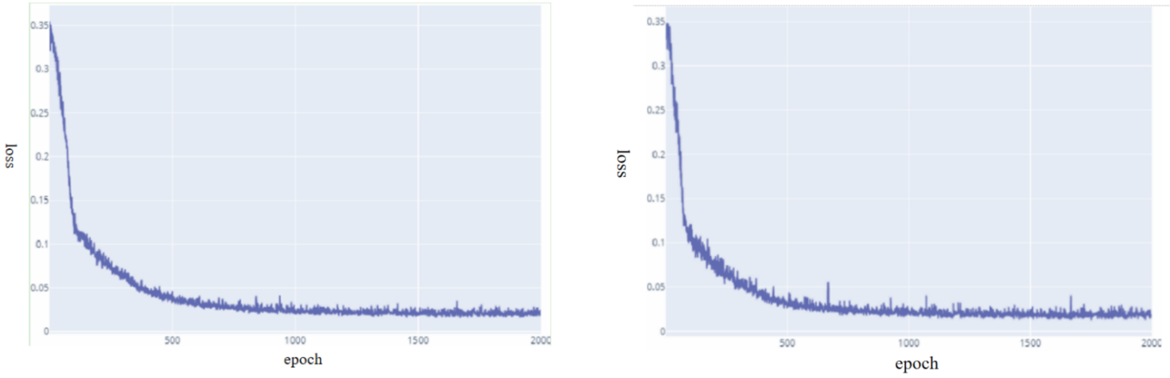
Table 12 shows examples of incomplete texts fixed by human agents and correcting mechanisms for the Sentiment 140 dataset: By analyzing these texts, we are able to comprehend three crucial aspects of automatic correction procedure in the proposed model: First, the correction processes could detect and correct incomplete words and faulty punctuation just as effectively as human agents (such as «Luv» and «its» in texts no. 1 and 3 from **Table 12**). Second, the correcting mechanisms, unlike human agents, are error-free, and if they detect the incorrect spelling of incomplete words, they will definitely correct them correctly. For instance, in **Table 12**, the word «following» in text no.1, was incorrectly corrected by human agents, but the proposed correction mechanisms produced the correct version. Thirdly, human agents may paraphrase instead of correcting (such as text no. 2 from **Table 12**), which can be troublesome in the process of training denoiser networks, as the correcting mechanisms simply correct the texts and do not make any customizations.

Table 12. Incomplete texts fixed by human agents and proposed correction mechanisms for the Sentiment 140 dataset.

#	Incomplete Text	Corrected by Human Agents	Corrected by Proposed Model
1	Much Luv to Mary and John who's now following me on twitter!!	Sending much love to Mary and John, who are now both folling me on Twitter.	much love to mary and john, who's now following me on twitter.
2	Me too I is poor	I am poor too.	me too, i am poor.
3	ugh. cant sleep. its 1:30 am.	I can't sleep and it's 1:30 am.	ugh. can not sleep. it's 1:30 am.
4	now ur leaving me....	Now you're leaving me!	now you are leaving me.

Although the overall impact of the automatic correction stage is clear from the high similarity values (shown in **Figure 10**), a determination of the unique contribution of each tool was necessary due to their distinct, non-redundant roles in the hierarchical refinement process. The tools have different quantitative impacts as well, with Ginger and NeuSpell being important for resolving discrete spelling and punctuation errors at the token level (e.g., "luv" → "love", shown in **Table 12**, row 1), ByT5 providing a unique ability to restore fragmented syntax (e.g., restoring the sentence structure in **Table 12**, row 4), and ELMo-LSTM being important for maintaining the flow of sequential information. Removing any one component leaves residual noise—whether it be morphological, syntactical, or semantically—that severely hinders the ability of the subsequent denoiser network to converge, thus confirming the necessity for the combined use of all three mechanisms in order to produce the correction quality observed.

Figure 11 depicts the decreasing trend of the model loss during the denoiser network training phase for the Sentiment 140 dataset in the two aforementioned scenarios. Training with texts generated by proposed correction processes has been somewhat more variable. The explanation for this difference can be observed in the potential output breakdowns.



(a) **(b)**

Figure 11. The model loss during the denoiser network training phase for the Sentiment 140: (**a**) first scenario; (**b**) second scenario.

As you can see in **Table 13**, the performance of proposed sentiment polarity identification phase is comparable in two scenarios of the Sentiment 140 dataset according to the F1 criterion. In fact, by automatically producing the complete form of incomplete texts using the proposed correction mechanisms, we were able to acquire results with a quality approximately equivalent to the generated texts by human agents.

Table 13. Evaluation of the proposed sentiment classification on the first and second scenarios of the Sentiment 140.

Criteria	Polarity	Scenario		# Instances
		First	Second	
Accuracy (%)	Positive	91.67	85.71	25
	Negative	88.46	95.45	25
Recall (%)	Positive	88.00	96.00	25
	Negative	92.00	84.00	25
F1 (%)	Positive	89.80	90.57	25
	Negative	90.20	89.36	25

5.3. Error Analysis

The failures of this proposed approach were examined by comparing and contrasting the confusion matrices along with the classification inaccuracies from each of these two datasets. The models demonstrated a greater number of accurate classifications for neutral sentiment in comparison to either positive or negative sentiment in the ACL 14 dataset; a result which is consistent with the composition of the ACL 14 data set, where there is a high number of neutral instances. In addition, although the model showed an impressive accuracy rating of 92% when predicting positive text in the Sentiment 140 data set (which significantly decreased the false-negative rate from 0.28 in the S-DeBERT baseline to 0.08), it had a 0.12 error rate when classifying negative text as positive. These results indicate the proposed approach performs well in generating and recognizing positive sentiment-based feature patterns using the attention and denoising mechanisms, but may find it much more difficult to differentiate negative polarity due to the difficulty in determining specific negative linguistic structure (such as ambiguity or sarcasm) in short and incomplete texts.

5.4. Theoretical Implications

In addition to assessing the empirical performance of the proposed model, it provides theoretical validation to NLP for dealing with short and incomplete texts through three major implications. The first implication, the success of the proposed denoising process (i.e., operating directly on the RoBERTa embedding layer instead of working with the surface text), supports the idea that semantic integrity can be restored in the latent space using vector-based reconstruction as an effective replacement for purely symbolic correction methods. The second implication is based upon the results of the improved sentiment detection produced by applying attention mechanisms at each of the intermediate layers of the transformer, supporting the theory of hierarchical information encoding and indicating that important local syntactic features required to analyze short texts are frequently contained within the mid-layers rather than being located in the higher-level abstractions represented in the top layers. Finally, the very high cosine similarities seen between the model's corrected texts generated through automated corrections and their corresponding human-annotated baseline texts support the notion that cascading automated correction processes may serve as a cost-effective replacement for time-consuming human-curation processes, thereby providing a sound way of developing solutions to scale for the analysis of large volumes of low-resourced or noisy social media datasets.

6. Conclusions

This paper presents a system based on transformers and neural networks to identify the polarity of user sentiment in the Twitter social network for short and incomplete data across three distinct phases, with the goal of analyzing the feelings expressed in the textual feedback of users. The proposed automatic correction phase experimental results show that the texts generated by the correction mechanisms are roughly 90% similar to the annotations of human agents. Furthermore, the experimental results for the training of the denoiser networks phase of the proposed model demonstrate the favorable convergence of the system, both for the Sentiment 140 dataset with fewer training samples and for the ACL 14 dataset with more training samples, and thus, it is a trustworthy mechanism for exploiting emotion polarity detection. Finally, the results for the third phase of our proposed model, using the outputs of all previous phases, demonstrated the model's overall performance in identifying the polarity of sentiment, also demonstrated the superior effectiveness of the proposed model over the SotA models, achieving a rate of 89.96% in the Sentiment 140 dataset and 76.91% in ACL 14 dataset according to the F1 criterion.

Syntax knowledge and the importance of POS tags, such as adjectives and adverbs, can have a positive effect on the quality of SA; therefore, their inclusion in the various phases of the proposed model can be considered as future work for a deeper and more thorough understanding of the input text. On the other hand, with the presence of sophisticated mechanisms for automatic text correction, syntax and related information can be applied to the corrected texts with more accuracy.

A potential direction for future work is to apply the proposed model to a real-world application, such as a recommender system, so that users could receive suggestions based on their personal preferences and the knowledge hidden in the sentiments of shared textual feedback. The modular design of this proposed model will provide great opportunities to generalize and invites further research in adapting this structure to additional languages and subject areas through integration with cross-lingual transformers or through applying transfer learning for various specialized technical domains, which will enable a denoising-based reconstruction of latent representations within low-resourced domains to diminish an extensive dependency upon human annotators.

Author Contributions

Conceptualization, R.N.G.; methodology, R.N.G.; software, R.N.G.; validation, N.T.; data curation, R.N.G.; writing—original draft preparation, R.N.G.; writing—review and editing, N.T.; supervision, N.T. Both authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding. All work was self-funded by the authors.

Institutional Review Board Statement

Not applicable. This study does not involve human or animal participation. It utilizes publicly available datasets for computational analysis.

Informed Consent Statement

Not applicable.

Data Availability Statement

The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request. The source code used to support the findings of this study is available from the corresponding author upon reasonable request.

Notes:

- **Ginger Software API**: Used for text preprocessing and grammar correction (www.gingersoftware.com/content/ginger-api).
- **Hugging Face**: Repository for the pre-trained transformer models (e.g., BERT/RoBERTa) used in this work (huggingface.co).
- **Google Colab**: The cloud computing platform used to execute the experiments (colab.research.google.com).
- **Kaggle**: Utilized as a high-performance computational platform and GPU resource for model training (www.kaggle.com).

Acknowledgments

The authors express their profound gratitude to Dr. Chitra Dadkhah for her invaluable mentorship and expert guidance throughout this research. Her generous commitment of time, insightful direction, and meticulous supervision were instrumental in shaping the quality of this work. Her constructive feedback and consistent encouragement provided a critical foundation for the project.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Zheng, C., 2025. A Comprehensive Review of Probabilistic and Statistical Methods in Social Network Sentiment Analysis. Advances in Engineering Innovation. 16(3), 38–43. DOI: https://doi.org/10.54254/2977-3903/2025.21918

[2] Tohidi, N., Beheshti, M., 2024. Enhanced Explanations in Recommendation Systems. In Proceedings of the 2024 IEEE International Symposium on Systems Engineering (ISSE), Perugia, Italy, 16–18 October 2024; pp. 1–5. DOI: https://doi.org/10.1109/ISSE63315.2024.10741116

[3] Ghezelji, M., Dadkhah, C., Tohidi, N., et al., 2022. Personality-Based Matrix Factorization for Personalization in Recommender Systems. International Journal of Information & Commununication Technology Research. 14(1), 48–55. DOI: https://doi.org/10.52547/itrc.14.1.48

[4] Liu, B., 2020. Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press: Cambridge, UK.

[5] Shaik, T., Tao, X., Dann, C., et al., 2023. Sentiment Analysis and Opinion Mining on Educational Data: A Survey. Natural Language Processing Journal. 2, 100003. DOI: https://doi.org/10.1016/J.NLP.2022.100003

[6] Ganji, R.N., Dadkhah, C., Tohidi, N., 2023. Improving Sentiment Classification for Hotel Recommender System through Deep Learning and Data Balancing. Computación y Sistemas. 27(3), 811–825. DOI: https://doi.org/10.13053/cys-27-3-4655

[7] Tang, J., Chang, Y., Liu, H., 2014. Mining Social Media with Social Theories: A Survey. ACM SIGKDD Explorations Newsletter. 15(2), 20–29. DOI: https://doi.org/10.1145/2641190.2641195

[8] Dong, L., Wei, F., Tan, C., et al., 2014. Adaptive Recursive Neural Network for Target-Dependent Twitter Sentiment Classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June 2014; pp. 49–54. DOI: https://doi.org/http://dx.doi.org/10.3115/v1/P14-2009

[9] Cunha Sergio, G., Lee, M., 2021. Stacked DeBERT: All Attention in Incomplete Data for Text Classification. Neural Networks. 136, 87–96. DOI: https://doi.org/10.1016/j.neunet.2020.12.018

[10] Vaswani, A., Shazeer, N., Parmar, N., et al., 2017. Attention Is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–7 December 2017; pp. 6000–6010. Available from: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[11] Devlin, J., Chang, M.-W., Lee, K., et al., 2019. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. DOI: https://doi.org/10.18653/v1/N19-1423

[12] Liu, Y., Ott, M., Goyal, N., et al., 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint. arXiv:1907.11692. DOI: https://doi.or

g/10.48550/arxiv.1907.11692
[13] Shetty, S.M., Pushpa, D., 2024. An Overview of the Literature on Sentiment Analysis Methods for Online and Social Media Platforms. In Proceedings of the 12th International Conference on Intelligent Systems and Embedded Design (ISED 2024), Rourkela, India, 20–22 December 2024; pp. 1–5. DOI: https://doi.org/10.1109/ISED63599.2024.10956498
[14] Alnasrawi, A.M., Alzubaidi, A.M.N., Al-Moadhen, A.A., 2024. Improving Sentiment Analysis Using Text Network Features within Different Machine Learning Algorithms. Bulletin of Electrical Engineering and Informatics. 13(1), 405–412. DOI: https://doi.org/10.11591/EEI.V13I1.5576
[15] Liu, Y., He, H., Han, T., et al., 2025. Understanding LLMs: A Comprehensive Overview from Training to Inference. Neurocomputing. 620, 129190. DOI: https://doi.org/10.1016/J.NEUCOM.2024.129190
[16] Zhang, H., Shafiq, M.O., 2024. Survey of Transformers and Towards Ensemble Learning Using Transformers for Natural Language Processing. Journal of Big Data. 11(1), 25. DOI: https://doi.org/10.1186/s40537-023-00842-0
[17] Shuang, K., Gu, M., Li, R., et al., 2021. Interactive POS-Aware Network for Aspect-Level Sentiment Classification. Neurocomputing. 420, 181–96. DOI: https://doi.org/10.1016/j.neucom.2020.08.013
[18] Liu, N., Shen, B., 2020. ReMemNN: A Novel Memory Neural Network for Powerful Interaction in Aspect-Based Sentiment Analysis. Neurocomputing. 395, 66–77. DOI: https://doi.org/10.1016/j.neucom.2020.02.018
[19] Naseem, U., Razzak, I., Musial, K., et al., 2020. Transformer Based Deep Intelligent Contextual Embedding for Twitter Sentiment Analysis. Future Generation Computer Systems. 113, 58–69. DOI: https://doi.org/10.1016/j.future.2020.06.050
[20] Zhou, J., Huang, J.X., Hu, Q.V., et al., 2020. SK-GCN: Modeling Syntax and Knowledge via Graph Convolutional Network for Aspect-Level Sentiment Classification. Knowledge-Based Systems. 205, 106292. DOI: https://doi.org/10.1016/j.knosys.2020.106292
[21] Liao, W., Zeng, B., Yin, X., et al., 2020. An Improved Aspect-Category Sentiment Analysis Model for Text Sentiment Analysis Based on RoBERTa. Applied Intelligence. 51, 3522–3533. DOI: https://doi.org/10.1007/S10489-020-01964-1
[22] Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. Neural Computation. 9(8), 1735–1780. DOI: https://doi.org/10.1162/NECO.1997.9.8.1735
[23] Chung, J., Gulcehre, C., Cho, K., et al., 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. arXiv preprint. arXiv:1412.3555. DOI: https://doi.org/10.48550/arXiv.1412.3555
[24] Basiri, M.E., Nemati, S., Abdar, M., et al., 2021. ABCDM: An Attention-Based Bidirectional CNN-RNN Deep Model for Sentiment Analysis. Future Generation Computer Systems. 115, 279–94. DOI: https://doi.org/10.1016/j.future.2020.08.005
[25] Song, Y., Wang, J., Liang, Z., et al., 2020. Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference. arXiv preprint. arXiv:2002.04815. DOI: https://doi.org/10.48550/arxiv.2002.04815
[26] Mutinda, J., Mwangi, W., Okeyo, G., 2023. Sentiment Analysis of Text Reviews Using Lexicon-Enhanced Bert Embedding (LeBERT) Model with Convolutional Neural Network. Applied Sciences. 13(3), 1445. DOI: https://doi.org/10.3390/APP13031445
[27] Mirugwe, A., Ashaba, C., Namale, A., et al., 2024. Sentiment Analysis of Social Media Data on Ebola Outbreak Using Deep Learning Classifiers. Life. 14(6), 708. DOI: https://doi.org/10.3390/LIFE14060708
[28] Jayanthi, S.M., Pruthi, D., Neubig, G., 2020. NeuSpell: A Neural Spelling Correction Toolkit. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, online, 16–20 November 2020; pp. 158–164. DOI: https://doi.org/10.18653/V1/2020.EMNLP-DEMOS.21
[29] Xue, L., Barua, A., Constant, N., et al., 2022. ByT5: Towards a Token-Free Future with Pre-Trained Byte-to-Byte Models. Transactions of the Association for Computational Linguistics. 10, 291–306. DOI: https://doi.org/10.1162/tacl_a_00461
[30] Raffel, C., Shazeer, N., Roberts, A., et al., 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research. 21(140), 1–67. Available from: https://jmlr.org/papers/v21/20-074.html
[31] Sokolova, M., Lapalme, G., 2009. A Systematic Analysis of Performance Measures for Classification Tasks. Information Processing & Management. 45(4), 427–437. DOI: https://doi.org/10.1016/J.IPM.2009.03.002
[32] Shridhar, K., Dash, A., Sahu, A., et al., 2019. Subword Semantic Hashing for Intent Classification on Small Datasets. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–6. DOI: https://doi.org/10.1109/IJCNN.2019.8852420
[33] Tang, D., Qin, B., Feng, X., et al., 2016. Effective LSTMs for Target-Dependent Sentiment Classification. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016; pp. 3298–3307. DOI: https://aclanthology.org/C16-1311/
[34] Ma, D., Li, S., Zhang, X., et al., 2017. Interactive Attention Networks for Aspect-Level Sentiment Analysis. In Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI'17), Mel-

bourne, Australia; pp. 4068–4074. Available from: https://dl.acm.org/doi/10.5555/3171837.3171854

[35] Sun, J., Han, P., Cheng, Z., et al., 2020. Transformer Based Multi-Grained Attention Network for Aspect-Based Sentiment Analysis. IEEE Access. 8, 211152–211163. DOI: https://doi.org/10.1109/ACCESS.2020.3039470

[36] Zhao, G., Luo, Y., Chen, Q., et al., 2023. Aspect-Based Sentiment Analysis via Multitask Learning for Online Reviews. Knowledge-Based Systems. 264, 110326. DOI: https://doi.org/10.1016/J.KNOSYS.2023.110326

[37] Grattafiori, A., Dubey, A., Jauhri, A., et al., 2024. The Llama 3 Herd of Models. arXiv preprint. arXiv:2407.21783. DOI: https://doi.org/10.48550/arXiv.2407.21783