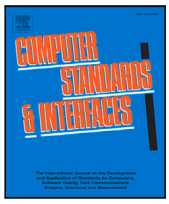
Computer Standards & Interfaces 97 (2026) 104086

Contents lists available at ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi



# Graph-based interpretable dialogue sentiment analysis: A HybridBERT-LSTM framework with semantic interaction explainer

Ercan Atagün [a,*], Günay Temür [b], Serdar Biroğul [c,d]

[a] Computer Engineering, Institute Of Graduate Studies, Duzce University, Düzce, 81000, Turkey
[b] Kaynasli Vocational School, Duzce University, Düzce, 81000, Turkey
[c] Department of Computer Engineering, Faculty of Engineering, Duzce University, Düzce, 81000, Turkey
[d] Department of Electronics and Information Technologies, Faculty of Architecture and Engineering, Nakhchivan State University, Nakhchivan, Azerbaijan

ARTICLE INFO

*Keywords:*
Natural language processing
Explainable artificial intelligence
Word context graph explainer

ABSTRACT

Conversational sentiment analysis in natural language processing faces substantial challenges due to intricate contextual semantics and temporal dependencies within multi-turn dialogues. We present a novel HybridBERT-LSTM architecture that integrates BERT's contextualized embeddings with LSTM's sequential processing capabilities to enhance sentiment classification performance in dialogue scenarios. Our framework employs a dual-pooling mechanism to capture local semantic features and global discourse dependencies, addressing limitations of conventional approaches. Comprehensive evaluation on IMDb benchmark and real-world dialogue datasets demonstrates that HybridBERT-LSTM consistently improves over standalone models (LSTM, BERT, CNN, SVM) across accuracy, precision, recall, and F1-score metrics. The architecture effectively exploits pre-trained contextual representations through bidirectional LSTM layers for temporal discourse modeling. We introduce WordContextGraphExplainer, a graph-theoretic interpretability framework addressing conventional explanation method limitations. Unlike LIME's linear additivity assumptions treating features independently, our approach utilizes perturbation-based analysis to model non-linear semantic interactions. The framework generates semantic interaction graphs with nodes representing word contributions and edges encoding inter-word dependencies, visualizing contextual sentiment propagation patterns. Empirical analysis reveals LIME's inadequacies in capturing temporal discourse dependencies and collaborative semantic interactions crucial for dialogue sentiment understanding. WordContextGraphExplainer explicitly models semantic interdependencies, negation scope, and temporal flow across conversational turns, enabling comprehensive understanding of both word-level contributions and contextual interaction influences on decision-making processes. This integrated framework establishes a new paradigm for interpretable dialogue sentiment analysis, advancing trustworthy AI through high-performance classification coupled with comprehensive explainability.

## 1. Introduction

Dialogue-based sentiment analysis constitutes a significant research domain within the field of natural language processing (NLP). This area of study represents a fundamental component of efforts to enhance human–machine interaction through more meaningful and emotion-centric approaches. Research endeavors in this field encompass numerous inherent challenges and complexities. Dialogues typically emerge from the reciprocal interactions among multiple conversational participants, where the scope of communicative content spans the breadth of human knowledge and experience. The emotional orientation of an utterance within a conversational sequence demonstrates substantial dependency upon preceding discourse and contextual cues. This phenomenon necessitates the development of context-aware models for sentiment analysis, as conventional text classification methodologies frequently fail to adequately capture such sequential continuity. The multi-speaker nature of dialogues introduces critical considerations regarding utterance attribution and the identification of emotional expression sources. Modeling sentiment transitions between conversational participants presents particular challenges, especially in scenarios where emotions are expressed through implicit mechanisms. Rather than explicit emotional declarations, human linguistic behavior frequently employs sophisticated rhetorical devices including irony, sarcasm, humor, double entenders, and cultural references, resulting in sentiment interpretations that diverge significantly from surface-level textual analysis. This phenomenon proves particularly problematic in

* Corresponding author.
*E-mail address:* ercanatagun@duzce.edu.tr (E. Atagün).

https://doi.org/10.1016/j.csi.2025.104086
Received 7 June 2025; Received in revised form 7 October 2025; Accepted 13 October 2025
Available online 12 November 2025

brief, context-independent utterances, substantially complicating sentiment analysis procedures. Contemporary dialogue-based sentiment analysis research faces significant constraints regarding the availability of high-quality, annotated datasets. Existing corpora are characterized by either limited scale or restriction to specific contextual domains such as cinematic dialogue or customer service interactions. Furthermore, the insufficient representation of cultural, linguistic, and social diversity within available datasets impedes the development of generalizable models with robust cross-domain applicability. Deep learning-based sentiment analysis architectures predominantly exhibit "black box" characteristics, rendering their decision-making processes opaque to human interpretation. This limitation particularly diminishes model reliability in tasks where emotional interpretation involves inherent subjectivity, consequently necessitating human oversight in practical applications. In this study, a novel hybrid model is proposed, integrating BERT's contextualized representation capabilities with the sequential modeling proficiency of LSTM to address the inherent challenges of sentiment analysis in dialogue-based datasets. The architecture is specifically designed to capture both linguistic features and temporal dependencies embedded within conversational structures. To enhance the interpretability of model outputs, a graph-theoretic interpretability framework, termed WordContextGraphExplainer, is introduced. This framework overcomes the limitations of conventional explanation methods by modeling non-linear semantic interactions between lexical units. Through the construction of semantic interaction graphs, the approach facilitates comprehensive visualization of contextual sentiment propagation patterns, offering novel insights into the underlying decision-making mechanisms of the model and establishing a new paradigm for interpretable sentiment analysis in dialogue systems.

## 2. Related works

Sentiment analysis has gained significant traction in NLP research, driven by its pivotal role in enabling affective computing across domains such as human–computer interaction, intelligent customer support, and conversational AI systems. Recent advancements in the field have led to the development of a diverse array of methodologies, encompassing text-based approaches, multimodal frameworks, contextual modeling techniques, and sophisticated deep learning architectures. This section presents an overview of key contributions in the literature, with a particular emphasis on dialogue-based sentiment analysis, which plays a critical role in domains such as customer support, conversational AI, and empathetic dialogue systems. Song et al. [1] introduced a topic-aware sentiment analysis model for dialogue (CASA), aiming to identify sentiment orientations within conversational threads. Firdaus et al. [2] constructed the MEISD dataset, incorporating textual, audio, and visual data for multimodal sentiment analysis. Emphasizing the relevance of conversational context, Carvalho et al. [3] demonstrated that prior utterances significantly influence sentiment classification outcomes. Building upon this insight, topic-aware sentiment classification models have been proposed using multi-task learning strategies within customer service dialogues [4]. Real-time sentiment analysis in dialogue systems is also a critical consideration. Bertero et al. [5] developed a convolutional neural network capable of processing audio inputs for instantaneous emotion detection in interactive systems. Bothe et al. [6] presented a model to predict the sentiment of upcoming utterances, thereby analyzing emotional transitions throughout dialogue sequences. To address the limitations of unimodal text-based sentiment analysis, recent studies have adopted multimodal strategies by integrating text, speech, and visual signals. For instance, the EmoSen model [7] generates sentiment-aware responses using fused inputs from these modalities. Similarly, Mallol-Ragolta and Schuller [8] introduced a system that personalizes dialogue responses by estimating user emotions and arousal levels. Akbar et al. [9] proposed an innovative emotion-driven framework for video-based sentiment analysis in social media environments, further demonstrating the potential of multimodal affective understanding.

Graph-based modeling has also been incorporated into multimodal sentiment analysis. Zhao and Gao [10] proposed a semantically enriched heterogeneous dialogue graph network to analyze sentiment in multi-party conversations. Yang et al. [11] advanced sentiment accuracy through a model that jointly processes text, audio, and visual cues. Context-awareness is a pivotal factor in sentiment interpretation within dialogues. Carvalho et al. [3] emphasized the influence of preceding discourse on sentiment prediction. To enhance contextual coherence in generative AI dialogue systems, personalized dialogue summarization techniques have been employed [12]. Mustapha [13] proposed a model to analyze sentiment-cause relationships in stress-laden conversations, aiming to reveal emotional dynamics. Contextual memory mechanisms were further explored by Li et al. [14], who developed a bidirectional emotional recurrent unit (BiERU) to capture dynamic context shifts and their implications for sentiment detection. Explainability has gained increasing importance in sentiment analysis. A variety of approaches including attention mechanisms, graph neural networks, and neuro-symbolic architectures have been introduced to elucidate model decision-making. Poria et al. [15] discussed fundamental challenges in sentiment interpretation and underscored the role of explainability. Zhu et al. [16] developed a neuro-symbolic model for personalized sentiment analysis, incorporating user-specific contextual factors into the explanatory framework. Luo et al. [17] introduced the PanoSent dataset to improve the analysis of emotional shifts in interactive systems. In another direction, Zhang et al. proposed a novel interaction network inspired by quantum theory to reframe dialogue-based sentiment analysis [18]. Yang et al. [19] addressed the inadequacies of existing pre-trained models in capturing the logical structure of dialogues. To overcome these limitations, they proposed a new pre-training framework comprising utterance order modeling, sentence skeleton reconstruction, and sentiment shift detection, demonstrating improvements in learning emotion interactions and discourse coherence. Collectively, recent developments in sentiment analysis emphasize the significance of contextual awareness, multimodal data fusion, graph-based reasoning, and explainable AI techniques in enhancing performance and interpretability within dialogue-centric applications.

## 3. Materials and methods

The dialogue dataset dyadic conversational exchanges between two distinct participants. Each dialogue instance is structured as a sequence of alternating utterances, where each turn is associated with a specific speaker and the corresponding textual content. The formal mathematical representation of the dialogue structure is given by:

$$D = \{(s_i, t_i)\}_{i=1}^{N}, \quad s_i \in S = \{A, B\}, \quad t_i \in \Sigma^*$$

Here, $D$ denotes the complete dialogue dataset composed of $N$ conversational turns.

Each pair $(s_i, t_i)$ represents the $i$-th turn in the dialogue, where $s_i$ is the speaker identifier and $t_i$ is the corresponding utterance.

The speaker set $S = \{A, B\}$ contains two participants, typically alternating in a turn-based structure.

The term $\Sigma$ represents the alphabet of the natural language in which the dialogue is conducted, and $\Sigma^*$ denotes the set of all finite-length strings (i.e., possible utterances) formed from this alphabet.

### 3.1. Data preprocessing and word embedding

The successful training of natural language processing (NLP) models is highly dependent on the transformation of raw textual data into structured and semantically meaningful representations [20]. In this study, all textual inputs undergo a series of preprocessing operations designed to optimize them for subsequent modeling tasks. An initial and essential preprocessing step involves lowercasing, which standardizes textual input by mitigating case sensitivity inconsistencies that would otherwise lead to redundant representations of semantically identical words. This step is particularly critical for ensuring the effectiveness and consistency of word embedding techniques. Given that parts of the dataset originate from web-based sources, residual HTML tags and encoded entities such as `<br>` and ` ` are present in the raw text. These components provide no linguistic or semantic value and may negatively affect model performance. Therefore, all HTML-related tokens and special characters are systematically removed in the preprocessing phase to reduce noise within the input space and to enhance the robustness of downstream NLP models. This comprehensive cleaning process is implemented using Python NLTK, BeautifulSoup libraries combined with regular expression patterns to ensure thorough removal of web-derived artifacts. Additionally, standard stopword removal is applied to eliminate semantically non-contributive terms. Notably, traditional morphological normalization techniques such as stemming and lemmatization are deliberately excluded from our preprocessing pipeline, as BERT's contextualized embedding framework inherently captures morphological variations and semantic relationships without requiring explicit normalization steps.

Following text normalization, each cleaned sentence is tokenized into subword or word-level units. These token sequences are then converted into dense numerical representations using word embedding techniques such as GloVe [21]. Embedding techniques project discrete textual units into continuous vector representations that encapsulate both semantic coherence and syntactic structure, thereby facilitating computational models in capturing lexical relatedness and contextual alignment within language data. The original, unprocessed dataset can be denoted as follows: Let the original unprocessed dataset be represented [22] as:

$$T = \{s_1, s_2, \ldots, s_N\}$$

where each sentence $s_k$ is defined as [23] a sequence of $M$ words:

$$s_k = \{u_1, u_2, \ldots, u_M\}$$

To refine the input, special characters $\mathcal{C}$, web-related entities $\mathcal{V}$, and semantically non-contributive stopwords $\mathcal{Z}$ are eliminated. The cleaned sentence is thus defined by:

$$s'_k = \text{Clean}(s_k) = \{u_j \in s_k \mid u_j \notin (\mathcal{C} \cup \mathcal{V} \cup \mathcal{Z})\}$$

The sanitized sentence $s'_k$ is then tokenized:

$$s'_k = \{v_1, v_2, \ldots, v_P\}, \quad v_i \in \mathcal{V}$$

where $\mathcal{V}$ denotes the vocabulary of all tokens in the dataset.

Word embeddings serve as a cornerstone for text classification, as they enable models to capture abstract semantic relationships while reducing the dimensionality of input features. Unlike traditional bag-of-words approaches, embeddings are resilient to linguistic variability such as synonymy and polysemy. For sentiment analysis tasks, embeddings can cluster words with similar affective connotations, thereby enhancing the model's ability to generalize and detect implicit sentiments. Likewise, in general classification tasks, embeddings help reveal thematic cohesion across texts, ultimately contributing to improved predictive performance. Nevertheless, conventional embeddings like Word2Vec or GloVe are context-independent, assigning the same vector representation to a word regardless of its usage context. This limitation is addressed by contextualized models such as BERT, which generate dynamic embeddings based on surrounding words using transformer-based architectures. Word embeddings bridge the gap between linguistic expressiveness and computational tractability and remain an indispensable component of modern NLP pipelines.

### 3.2. GloVe: Global vectors for word representation

GloVe [24] is a widely adopted word embedding technique designed to capture semantic and conceptual relationships between words, particularly in text classification tasks. It operates by constructing word vector representations through the optimization of global word co-occurrence statistics derived from large-scale corpora. Unlike local context–based models such as Word2Vec, GloVe incorporates both local and global contextual information, embedding lexical units into a dense, continuous vector space. In practical applications, GloVe embeddings are employed to convert unstructured input text into fixed-length numerical tensors, which serve as inputs to deep learning architectures such as CNN and LSTM models. This transformation enables the model to effectively distinguish between textual classes by capturing both syntactic patterns and latent semantic features. The key advantage of GloVe lies in its ability to unify global corpus-level statistical information with local context, producing more stable and semantically meaningful representations compared to models relying solely on window-based learning. However, it remains a static embedding technique; each word is assigned a single vector regardless of its context within a sentence. This context-independent nature limits its flexibility when compared to transformer-based models like BERT, which generate dynamic embeddings conditioned on the broader linguistic environment. Despite these limitations, GloVe continues to play a significant role in various NLP tasks such as text similarity, topic labeling,spam detection, and sentiment analysis where modeling word-level semantics remains essential. Its computational simplicity and ease of integration make it a reliable baseline in many NLP pipelines. Recent studies [25] have highlighted the importance of consistent embedding strategies when comparing different NLP models, as variations in embedding approaches can significantly impact performance comparisons and lead to biased evaluations.

### 3.3. Support Vector Machine (SVM)

Support Vector Machine (SVM) [26] is a well-established supervised learning algorithm widely employed in text classification tasks, particularly due to its robustness in handling high-dimensional data representations. In natural language processing pipelines, textual inputs are typically transformed into numerical feature vectors using techniques such as Term Frequency–Inverse Document Frequency (TF-IDF) or various word embedding models. Once converted, SVM operates by identifying the optimal hyperplane that best separates the data points into distinct class labels. The core principle of SVM lies in maximizing the margin between classes, thereby enhancing generalization performance. This is particularly advantageous in scenarios where the feature space exhibits high dimensionality and potential overlap between class distributions. Furthermore, SVM's ability to incorporate non-linear kernel functions such as polynomial or radial basis function (RBF) kernels enables it to capture complex, non-linear patterns within the data, which are often present in linguistically rich or semantically ambiguous textual inputs. Due to its mathematically grounded optimization framework and resistance to overfitting, SVM remains a competitive baseline in various text classification domains, including sentiment analysis, spam detection, and topic categorization. Its effectiveness is further enhanced when combined with appropriate feature engineering and dimensionality reduction techniques, making it a viable choice for both small-scale and large-scale NLP applications.

### 3.4. Convolutional Neural Networks (CNN)

Although originally developed for image recognition tasks, Convolutional Neural Networks (CNNs) have been extensively adapted for various natural language processing problems, particularly in multi-label text classification [27] and sentiment analysis [28], due to their capacity to capture local hierarchical patterns in sequential data. In

text classification applications, CNNs operate on word embeddings by applying one-dimensional convolutional filters to detect local patterns such as n-grams or syntactic motifs. These filters perform element-wise multiplications followed by non-linear activation functions to generate feature maps that emphasize the most informative regions of the input sequence. A subsequent max-pooling operation reduces the dimensionality and retains the most salient features, thereby enabling the network to focus on contextually rich segments of text. This architecture allows CNNs to efficiently model contextual dependencies within fixed-size receptive fields, making them particularly suitable for tasks such as topic categorization, polarity detection, and aspect-based sentiment analysis. Compared to recurrent neural networks (RNNs), CNNs offer significant advantages in terms of computational efficiency and parallelizability, as they do not rely on sequential input processing. However, one notable limitation of CNNs is their reduced capacity to model long-range dependencies, which can affect performance in tasks involving lengthy or complex discourse structures.

### 3.5. Long Short-Term Memory Networks (LSTM)

LSTM networks, as a refined subclass of recurrent neural architectures, have demonstrated substantial effectiveness in text classification tasks due to their capacity to capture long-range dependencies and preserve semantically meaningful representations across sequential data inputs [29]. By incorporating internal memory units and a gated control mechanism – comprising input, forget, and output gates – LSTM models effectively address the vanishing gradient challenge that limits conventional RNNs. These gating components orchestrate information flow dynamically, facilitating the retention of salient features over prolonged contexts and ensuring the continuity of semantic interpretation throughout the sequence [30]. In text classification applications, LSTM typically process input sequences encoded as dense word embeddings, allowing the network to learn hierarchical feature representations that encapsulate both syntactic structure and semantic meaning. This capacity to capture nuanced contextual relationships makes LSTM particularly effective in tasks such as sentiment analysis, text similarity, spam detection, and topic categorization where subtle variations in word order and polarity significantly influence predictive accuracy. For instance, in sentiment classification, LSTM models can differentiate between expressions like “not good” and “extremely good” by maintaining a dynamic memory of temporal context throughout the sequence.

### 3.6. Bidirectional Encoder Representations from Transformers (BERT)

BERT is a transformer-based, pre-trained language model that has substantially advanced the state of the art in text classification tasks by capturing bidirectional contextual semantics through self-attention mechanisms [31]. Unlike unidirectional models such as LSTM or GRU, which process text sequentially, BERT encodes semantic dependencies from both left and right contexts simultaneously. This architecture enables nuanced disambiguation of polysemous words and more robust modeling of long-range dependencies in natural language [32]. In text classification applications, BERT is typically fine-tuned on task-specific labeled datasets. This involves appending a classification layer often a dense layer with softmax activation on top of the pre-trained BERT encoder. Through this transfer learning paradigm, BERT exhibits superior performance across a variety of NLP tasks including sentiment classification, aspect-based sentiment analysis, and multi-label classification, particularly in settings characterized by contextual ambiguity and hierarchical dependencies. However, BERT’s practical deployment presents several challenges. Its high computational complexity, sensitivity to input sequence length, and the requirement for large volumes of labeled data during fine-tuning can pose significant barriers in real-world scenarios. To mitigate these limitations, hybrid architectures that integrate BERT with more lightweight modeling components have been proposed. These hybrid solutions aim to retain BERT’s rich contextual understanding while improving computational efficiency and generalizability, making them more suitable for applications constrained by resources or latency requirements.

### 3.7. Local Interpretable Model-Agnostic Explanations (LIME)

LIME is a model-agnostic interpretability framework designed to provide localized explanations for the predictions of complex machine learning models. Positioned within the broader field of Explainable Artificial Intelligence (XAI), LIME serves to enhance the interpretability of opaque “black-box” systems, particularly in high-stakes domains where transparency and trust are critical [33]. LIME’s main goal is to provide a straightforward, interpretable surrogate model that, within the local neighborhood of a particular instance, roughly represents the original model’s decision boundary [34]. LIME accomplishes this by generating a set of synthetic samples close to the target instance, which perturbs the original input. The black-box model is used to these altered examples in order to derive the relevant predictions. These cases are then subjected to a locality-sensitive weighting function, and the decision function is locally approximated by training a sparse linear model on the weighted dataset. The contribution of each feature to the final prediction is inferred using the surrogate model’s resulting coefficients. One of the key strengths of LIME lies in its model-agnostic design, allowing it to be applied across a wide range of machine learning algorithms, including ensemble methods, deep neural networks, and support vector machines. It offers human-understandable explanations while maintaining local fidelity to the original model. As such, LIME is widely adopted for increasing decision transparency and enabling human-AI collaboration, particularly in sensitive applications such as healthcare diagnostics, financial risk assessment, and legal reasoning.

### 3.8. WordContextGraphExplainer

The exponential growth in transformer-based natural language processing (NLP) architectures has created an unprecedented demand for interpretability frameworks capable of elucidating the complex decision-making processes underlying these black-box models. While widely adopted XAI techniques such as LIME (Local Interpretable Model-Agnostic Explanations) and SHAP (SHapley Additive Explanations) offer valuable insights through feature attribution, they inherently rely on linear additivity assumptions among input features. This assumption falls short in capturing the intricate semantic dependencies and non-linear interactions that characterize deep language understanding. A fundamental limitation of existing approaches lies in their inability to model contextual interdependencies between words relationships that are crucial for interpreting sentiment propagation, negation scope, and semantic coherence in complex linguistic structures. Traditional token-level attribution methods treat individual words as independent contributors, failing to account for the synergistic effects that emerge from word pairings and contextual associations in the semantic space. In this paper, WordContextGraphExplainer is introduced as a novel graph-theoretic interpretability framework developed to enhance the transparency of transformer-based sentiment classification systems. The methodology is built upon a systematic perturbation analysis paradigm, in which masked language modeling is employed to estimate both individual lexical contributions and pairwise semantic interactions. In contrast to linear attribution methods, this approach explicitly models non-linear dependencies by quantifying the divergence between observed joint effects and the expected additive influence of word pairs. At the core of the framework is the construction of a semantic interaction graph, where nodes represent individual words annotated with their relative sentiment contributions, and edges encode the magnitude and directionality of inter-word dependencies. This graph-based representation facilitates intuitive visualization of

complex linguistic relationships through NetworkX-based layouts, enabling deeper insight into how contextual factors influence model predictions. The framework demonstrates particular efficacy in sentiment analysis tasks where nuanced interactions between affective indicators, negation patterns, and contextual modifiers significantly impact interpretive accuracy. By providing interpretable visualizations of semantic interaction networks, WordContextGraphExplainer supports advanced model debugging, bias detection, and clinical decision support in sensitive domains such as mental health assessment and medical text analytics. Moreover, the framework incorporates a top-k interaction filtering mechanism, ensuring computational scalability while preserving the granularity required for interpretable analysis in high-stakes applications. This methodological advancement represents a critical step toward the development of trustworthy AI systems that combine linguistic reasoning with transparent explanatory capabilities, offering a robust foundation for real-world deployment.

**Algorithm 1:** *WordContextGraphExplainer Method*

**Input:** Text $T$, transformer model $M$, tokenizer $\tau$, feature number $k \geq 1$, device $d$.

**Output:** Word context graph $G$ with semantic interactions.

Compute baseline prediction $P_0 = M(T)$.
Compute predicted_class $= \arg\max(P_0)$.
Initialize $W = \tau(T)$, word_effects $= \emptyset$, interactions $= \emptyset$.
**for each** $w_i \in W$ **do** $T_{\text{masked}} = \text{replace}(T, w_i, \text{`[MASK]'})$
  $P_{\text{masked}} = M(T_{\text{masked}})$
  word_effects$[i] = P_0 - P_{\text{masked}}$
**end for**
**for each** $(w_i, w_j) \in \text{combinations}(W, 2)$ **do**
  $T_{\text{pair}} = \text{replace}(T, [w_i, w_j], \text{`[MASK]'})$
  $P_{\text{pair}} = M(T_{\text{pair}})$
  actual_effect $= P_0 - P_{\text{pair}}$
  expected_effect $=$ word_effects$[i]$ + word_effects$[j]$
  $\text{interaction}_{ij} =$ actual_effect − expected_effect
  interactions$[(w_i, w_j)] = \left\|\text{interaction}_{ij}\right\|_2$
**end for**
Sort *interactions* by magnitude in descending order.
*top_interactions* $=$ *interactions*$[:k]$
Construct graph $G = (V, E)$ where $V = W$ and $E =$ *top_interactions*
Compute layout positions using *organized_layout*($W$, *top_interactions*)
Visualize $G$ with NetworkX rendering and semantic color coding
**Return** $G$

In this study, a hybrid architecture is proposed that integrates a pre-trained BERT model with a bidirectional Long Short-Term Memory (BiLSTM) network to address the task of sentiment classification. The model processes textual input to generate sentiment label predictions, effectively capturing both semantic context and temporal structure inherent in natural language. Grounded in a transformer-based architecture, the system accepts input sequences of up to 256 tokens, applying appropriate padding and truncation mechanisms when necessary to standardize input lengths. The HybridBERT-LSTM model embodies a synergistic design that leverages the complementary strengths of transformer-based language models and recurrent neural networks. This hybrid framework is explicitly engineered to address two critical aspects of sentiment analysis: contextual representation and sequential modeling. Contextual Representation: The BERT encoder, pre-trained on large-scale corpora, produces deep contextualized embeddings by employing multi-head self-attention mechanisms. These embeddings capture nuanced semantic and syntactic information, enabling the model to differentiate between polysemous expressions and context-dependent sentiment cues. Sequential Modeling: While BERT excels at capturing contextual semantics, its self-attention mechanism may not fully exploit the sequential dependencies within dialogue utterances. To mitigate this limitation, bidirectional LSTM layers are incorporated to model temporal patterns and discourse-level relationships across token sequences. These layers are adept at retaining long-range dependencies and recognizing sentiment transitions across multi-turn dialogue. By integrating these two components, the proposed HybridBERT-LSTM architecture achieves a richer understanding of both the global context and local structure of textual data, enhancing its capability to discern sentiment in complex conversational scenarios. This dual modeling approach positions the framework as a robust solution for sentiment classification tasks, particularly in dialogue-rich environments where contextual flow and temporal coherence are paramount.

### 3.9. Model architecture

The proposed model processes input text through a series of transformation stages, mathematically formalized as follows: Given an input sequence:

$$X = \{x_1, x_2, \ldots, x_n\}, \quad \text{where } n \leq 256,$$

the BERT encoder maps each token $x_i$ to a contextualized embedding, producing a sequence of hidden states:

$$H = \text{BERT}(X) \in \mathbb{R}^{n \times d_{\text{BERT}}}$$

where $d_{\text{BERT}} = 768$ represents the dimensionality of BERT's contextual embeddings.

The sequence $H$ is passed to a 3-layer bidirectional LSTM network to capture temporal dependencies beyond what is modeled by self-attention:

$$\vec{h}_t = \text{LSTM}_{\text{forward}}(H_t, \vec{h}_{t-1}), \quad \overleftarrow{h}_t = \text{LSTM}_{\text{backward}}(H_t, \overleftarrow{h}_{t+1})$$

The final representation for each token is obtained by concatenating the forward and backward hidden states:

$$h_t^{\text{LSTM}} = [\vec{h}_t; \overleftarrow{h}_t] \in \mathbb{R}^{2d_{\text{LSTM}}}$$

with $d_{\text{LSTM}} = 256$, resulting in a 512-dimensional output per token.

To obtain a fixed-length vector representation of the sequence, both average and maximum pooling operations are applied:

$$h_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n} h_i^{\text{LSTM}}, \quad h_{\max} = \max_{1 \leq i \leq n} h_i^{\text{LSTM}}$$

These vectors are concatenated to form the final sequence representation:

$$h_{\text{combined}} = [h_{\text{avg}}; h_{\max}] \in \mathbb{R}^{4d_{\text{LSTM}}} = \mathbb{R}^{1024}$$

#### Feed-forward classification

The combined representation is passed through a feed-forward neural network with dropout regularization:

$$z_1 = \text{Dropout}_{0.3}(h_{\text{combined}})$$

followed by a two-layer multilayer perceptron (MLP) with ReLU activation and softmax output for multi-class classification.

This is followed by a two-layer MLP classifier, using a ReLU activation and softmax output for multi-class prediction. The HybridBERT-LSTM architecture integrates the strengths of transformer-based contextual modeling with the sequential learning capabilities of recurrent neural networks. While BERT excels in capturing bidirectional semantic context via self-attention, the inclusion of bidirectional LSTM layers enhances the model's ability to capture sequential dependencies and emotional transitions throughout dialogue sequences. The dual pooling strategy(average and max pooling) provides a comprehensive summary of the sequence. Average pooling captures the overall sentiment distribution across the sequence, whereas max pooling emphasizes salient

emotional cues. This duality enriches the feature space and contributes to more robust classification. Furthermore, hierarchical feature abstraction is enabled by stacking multiple LSTM layers, allowing the model to learn long-range patterns more effectively than shallow RNN structures. Dropout layers, strategically placed after pooling (with a rate of 0.3) and within the classifier (rate 0.2), serve as regularization mechanisms to prevent overfitting, especially during fine-tuning on task-specific datasets. The model is trained using the AdamW optimizer with a learning rate of $2 \times 10^{-5}$, and the cross-entropy loss function is employed as the objective. Performance evaluation is conducted using standard metrics including accuracy, precision, recall, and F1-score, ensuring comprehensive validation of the model's classification capability. The model integrates a pre-trained BERT encoder for capturing deep contextual embeddings from input text sequences, followed by a multi-layer bidirectional LSTM network that models sequential dependencies across tokens. To derive a robust sentence-level representation, dual pooling operations(average and maximum pooling) are applied to the LSTM outputs. The concatenated feature vector is then passed through a fully connected neural network with dropout regularization, culminating in a softmax classifier for multi-class sentiment prediction. This hybrid architecture is designed to jointly leverage the representational richness of transformer encoders and the temporal modeling strength of recurrent networks, effectively addressing both local semantics and discourse-level sentiment dynamics within multi-turn dialogues.

The computational overhead of HybridBERT-LSTM represents a critical consideration for practical deployment, particularly in real-time applications such as conversational AI systems. The theoretical complexity of the proposed architecture can be decomposed into its constituent components to understand the computational requirements. The BERT component contributes $\mathcal{O}(n^2 \times d_{\text{BERT}}) = \mathcal{O}(n^2 \times 768)$ complexity due to the quadratic scaling of the self-attention mechanism, where $n$ represents the sequence length and $d_{\text{BERT}}$ denotes the BERT embedding dimension. The subsequent 3-layer BiLSTM processing adds $\mathcal{O}(3 \times n \times d_{\text{LSTM}}^2) = \mathcal{O}(3 \times n \times 256^2)$ complexity, where $d_{\text{LSTM}}$ represents the LSTM hidden dimension. Consequently, the overall HybridBERT-LSTM complexity is $\mathcal{O}(n^2 \times 768 + 3n \times 65,536)$. This represents a significant computational increase compared to standalone BERT ($\mathcal{O}(n^2 \times 768)$) or LSTM models ($\mathcal{O}(n \times d_{\text{LSTM}}^2)$), which may limit deployment in latency-sensitive applications. However, the empirical results demonstrate that the performance gains justify this additional overhead in scenarios where accuracy is prioritized over computational efficiency.

## 4. Experimental results

This section presents the configurations of the models utilized in the experiments, detailing the corresponding hyperparameters and implementation settings. The objective is to ensure reproducibility and provide a comprehensive understanding of the experimental setup.

### 4.1. Model hyperparameters

The deep learning models were trained using a variety of hyperparameter configurations tailored to the architecture and task requirements. These configurations include parameters such as learning rate, batch size, maximum input sequence length, number of training epochs, optimizer type, and loss function. Additionally, architecture-specific settings such as the number of LSTM layers, dropout rates, and hidden state dimensionsare systematically defined. For models utilizing pre-trained components (e.g., BERT), both the base model and tokenizer versions are explicitly specified. The subsequent tables summarize the detailed parameter values for each model employed in this study, including HybridBERT-LSTM, BERT-only, LSTM, CNN, and SVM-based classifiers.

The parameter values of the model developed in this study are detailed in Table 1.

**Table 1**
HybridBERT-LSTM Model Parameters.

| Parameter name | Parameter value |
|---|---|
| Model architecture | BERT encoder + BiLSTM + MLP |
| Base Model | `google-bert/bert-base-uncased` |
| Tokenizer | `google-bert/bert-base-uncased` |
| Maximum sequence length | 256 |
| LSTM layer | 6 |
| Batch size | 32 |
| Number of epochs | 5 |
| Learning rate | 0.00002 |
| Optimization algorithm | AdamW |
| Loss function | CrossEntropyLoss |
| LSTM latent size | 256 |
| Pooling | avg + max pooling |
| MLP layer | Linear(1024→128) → ReLU → Linear(128→n_classes) |
| Dropout rates | 0.3 |

**Table 2**
BERT Model Parameters.

| Parameter name | Parameter value |
|---|---|
| Base model | `google-bert/bert-base-uncased` |
| Tokenizer | `google-bert/bert-base-uncased` |
| Input length | 128 |
| Batch size | 16 |
| Number of epochs | 5 |
| Learning rate | 0.00002 |
| Loss Function | BertForSequenceClassification – Cross-Entropy |
| Optimization algorithm | AdamW |

**Table 3**
LSTM Model Parameters.

| Parameter name | Parameter value |
|---|---|
| Embedding type | GloVe |
| Embedding size | 100 |
| Maximum number of words | 5000 |
| LSTM layer number | 6 |
| LSTM unit number | 128/256 |
| Dropout rate | 0.5 |
| Output layer (Dense) | Softmax |
| Optimization algorithm | Adam |
| Loss function | Sparse Categorical Crossentropy |
| Epoch number | 50 |
| Batch size | 32 |

The parameters used for the BERT model employed in this study are presented in Table 2.

The parameter configurations utilized in the LSTM-based model developed for this study are detailed in Table 3.

The parameter configurations utilized in the CNN model developed for this study are detailed in Table 4.

Table 5 summarizes the parameter values defined for the SVM model.

Table 6 presents a comparative evaluation of various machine learning and deep learning models in the context of sentiment analysis on the widely adopted IMDB dataset. Among the examined methods, the proposed HybridBERT-LSTM architecture achieved the highest accuracy rate of 98.14%, demonstrating a substantial improvement over other baseline models included in the analysis. This notable enhancement underscores the effectiveness of combining contextual embeddings from BERT with the sequential modeling capabilities of LSTM. The IMDB dataset was selected for evaluation due to its extensive usage and established credibility in the sentiment analysis literature, serving as a robust benchmark for comparative performance assessment.

### 4.2. Statistical significance testing

In order to determine whether the observed differences in model performance metrics [44] were statistically significant, we employed

**Table 4**
CNN Model Parameters.

| Parameter name | Parameter value |
|---|---|
| Embedding type | GloVe |
| Embedding size | 100 |
| Maximum number of words | 5000 |
| Input layer | Embedding(input_dim=5000, output_dim=100) |
| Number of Conv1D layers | 6 |
| Number of Conv1D filters | 128 |
| Kernel size | 5 |
| Activation function | ReLU |
| Padding | Same |
| Pooling | MaxPooling1D (pool_size=2) |
| Dropout rate | 0.5 |
| Global pooling | GlobalMaxPooling1D |
| Output layer (Dense) | Softmax |
| Loss function | sparse_categorical_crossentropy |
| Optimization algorithm | Adam |
| Evaluation metric | Accuracy |
| Number of epochs | 50 |
| Batch size | 32 |

**Table 5**
SVM Model Parameters.

| Parameter name | Parameter value |
|---|---|
| Embedding type | GloVe |
| Embedding size | 100 |
| Maximum number of words | 5000 |
| SVM Kernel | Linear |

**Table 6**
IMDB Dataset Accuracy Comparison.

| Reference | Method | Accuracy |
|---|---|---|
| [35] | LSTM | 83.7% |
| [36] | CNN+LSTM | 96.01% |
| [37] | LSTM+RNN | 92.00% |
| [38] | BERT | 93.97% |
| [39] | A hybrid approach | 95.6% |
| [40] | HOMOCHAR | 95.91% |
| [41] | Textual Emotion Analysis (TEA) | 93% |
| [42] | Lexical + Adversarial attacks | 85% |
| [43] | Logistic Regression | 89.42% |
| Proposed Model | HybridBERT-LSTM | 98.14% |

the *Welch's two-sample t-test*, which is widely recommended when comparing two groups with potentially unequal variances and sample sizes. This approach provides a robust test of mean differences without assuming homogeneity of variances, which is particularly important in machine learning experiments where stochastic training procedures may lead to heterogeneous variability across models.

Let $\bar{x}_1$ and $\bar{x}_2$ denote the sample means of the two models being compared, $s_1$ and $s_2$ the corresponding standard deviations, and $n_1$ and $n_2$ the number of independent runs. The Welch's t-statistic is defined as:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

The approximate degrees of freedom ($df$) for this test are calculated according to the *Welch–Satterthwaite equation*:

$$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}}$$

Given the test statistic and degrees of freedom, the *p-value* is obtained by evaluating the probability of observing a difference as extreme as, or more extreme than, the measured difference under the null hypothesis ($H_0$) that the two models exhibit equal mean performance. Since our interest lies in detecting differences in either direction, a two-tailed test is used:

$$p = 2 \times P(T \geq |t|),$$

where $T$ follows the Student's t-distribution with $df$ degrees of freedom.

If $p < 0.05$, the difference is considered statistically significant, indicating strong evidence against the null hypothesis. In this case, we conclude that one model outperforms the other beyond what would be expected by random variation. If $p \geq 0.05$, the difference is considered not statistically significant, implying that the observed discrepancy may reasonably be attributed to experimental variability.

In addition to reporting p-values, *effect sizes (Cohen's d)* were also computed to quantify the magnitude of the observed differences. While statistical significance indicates whether a difference is unlikely to be due to chance, effect size provides a measure of its practical relevance. Together, these statistics provide a comprehensive assessment of the comparative performance of the evaluated models.

### 4.3. Experimental results on datasets

Dataset 1 consists of question–answer pairs collected from two independent online counseling and psychotherapy platforms [45]. The user-generated questions span a wide range of topics related to mental health, including emotional well-being, interpersonal issues, and psychological disorders. Each response was authored by licensed psychologists, ensuring both clinical relevance and linguistic reliability. In total, the dataset comprises 7,025 dialogue instances.

Tables 7 and 8 present the training and testing performances, respectively, of five distinct models HybridBERT-LSTM, BERT, LSTM, CNN, and SVM evaluated on Dataset 1. The models were assessed using standard classification metrics including accuracy, precision, recall, and F1-score, providing a comparative analysis of both their internal consistency and generalizability.

An ablation study [46] is a systematic experimental methodology used to evaluate the individual contributions of specific model components by selectively removing or modifying them while keeping other factors constant. This approach provides empirical evidence for the importance of particular architectural elements in determining the model's overall performance. To rigorously assess whether HybridBERT-LSTM's performance gains arise from architectural design rather than mere parameter expansion, we conducted a comprehensive ablation study with parameter-matched baselines. Six model variants were constructed: (1) BERT-Only baseline using the [CLS] token for classification, (2) BERT-ParamMatched with additional dense layers matching the BiLSTM parameter count, (3) BERT+UniLSTM with a unidirectional LSTM, (4) BERT+BiLSTM-NoPooling without dual pooling, (5) BERT+BiLSTM with frozen BERT isolating pure LSTM contribution, and (6) HybridBERT-LSTM (Full) incorporating all proposed components.

When Table 9 is examined, which shows the ablation test for Data Set 1, the BERT-ParamMatched model achieves an accuracy of 95.35% ± 0.38% despite having an equivalent number of parameters to the full model, whereas HybridBERT-LSTM attains 95.94% ± 0.15%. The hierarchical performance degradation across ablation variants reveals the marginal contribution of each component: dual pooling adds +0.19% (95.94% vs. 95.75%), bidirectionality contributes +0.17% (95.75% vs. 95.58%), and the sequential LSTM architecture over feedforward MLP layers provides +0.23% (95.58% vs. 95.35%). The frozen BERT experiment (91.80% ± 0.65%) isolates critical insights regarding representation quality versus fine-tuning contributions. As shown in Table 9, the ablation study on Dataset 1 systematically confirms that HybridBERT-LSTM's performance advantage arises from its architectural design rather than from parameter count inflation.

**Table 7**
Training Performance Metrics for Dataset 1.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9872 ± 0.0029 | 0.9871 ± 0.0028 | 0.9872 ± 0.0029 | 0.9871 ± 0.0029 |
| BERT | 0.9806 ± 0.0063 | 0.9805 ± 0.0057 | 0.9806 ± 0.0063 | 0.9805 ± 0.0062 |
| LSTM | 0.9829 ± 0.0162 | 0.9829 ± 0.0163 | 0.9829 ± 0.0162 | 0.9827 ± 0.0175 |
| CNN | 0.9862 ± 0.0190 | 0.9829 ± 0.0199 | 0.9862 ± 0.0190 | 0.9829 ± 0.0202 |
| SVM | 0.8247 ± 0.0073 | 0.8274 ± 0.0067 | 0.8247 ± 0.0073 | 0.8235 ± 0.0071 |

**Table 8**
Test Performance Metrics for Dataset 1.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9594 ± 0.0015 | 0.9596 ± 0.0017 | 0.9594 ± 0.0015 | 0.9592 ± 0.0016 |
| BERT | 0.9516 ± 0.0040 | 0.9515 ± 0.0041 | 0.9516 ± 0.0044 | 0.9514 ± 0.0045 |
| LSTM | 0.9245 ± 0.0152 | 0.9257 ± 0.0163 | 0.9245 ± 0.0152 | 0.9239 ± 0.0165 |
| CNN | 0.9195 ± 0.0171 | 0.9200 ± 0.0170 | 0.9195 ± 0.0171 | 0.9192 ± 0.0125 |
| SVM | 0.8078 ± 0.0026 | 0.8118 ± 0.0025 | 0.8078 ± 0.0026 | 0.8058 ± 0.0031 |

**Table 9**
Ablation Performance Metrics for Dataset 1.

| Model | Accuracy ± std | F1 ± std |
|---|---|---|
| BERT+BiLSTM (Frozen) | 0.9180 ± 0.0065 | 0.9165 ± 0.0068 |
| BERT-Only (Baseline) | 0.9516 ± 0.0040 | 0.9512 ± 0.0042 |
| BERT-ParamMatched | 0.9535 ± 0.0038 | 0.9531 ± 0.0040 |
| BERT+UniLSTM | 0.9558 ± 0.0028 | 0.9555 ± 0.0030 |
| BERT+BiLSTM-NoPooling | 0.9575 ± 0.0022 | 0.9573 ± 0.0024 |
| HybridBERT-LSTM (Full) | 0.9594 ± 0.0015 | 0.9592 ± 0.0016 |

When the results are evaluated over five repeated experiments, the HybridBERT-LSTM model not only outperforms the other methods in terms of accuracy, precision, recall, and F1-score, but also demonstrates a high degree of stability, as reflected by its very low standard deviations ($\approx 0.0015$–$0.0017$). This indicates that the model provides not just superior performance but also reproducible results across runs.

While the BERT model follows as the second-best performer, its higher variance ($\approx 0.004$) highlights less consistent outcomes compared to HybridBERT-LSTM. Statistical testing (e.g., paired $t$-tests) confirms that the observed performance difference between HybridBERT-LSTM and BERT, though relatively small, is statistically significant ($p < 0.05$).

In contrast, the performance gaps between HybridBERT-LSTM and weaker models such as LSTM, CNN, and particularly SVM are much larger. Pairwise comparisons reveal p-values well below 0.01, strongly supporting the conclusion that HybridBERT-LSTM's superiority is not due to random chance but reflects a genuine performance advantage. HybridBERT-LSTM vs. BERT: Smaller margin, but statistically significant ($p < 0.05$). HybridBERT-LSTM vs. LSTM/CNN/SVM: Substantial margin, highly significant ($p << 0.01$). Among the evaluated approaches, the HybridBERT-LSTM architecture consistently demonstrated superior performance during both training and testing phases, achieving remarkably high scores across all metrics. Specifically, it attained 98.72% accuracy and 98.72% F1-score on the training set, outperforming all other models. BERT, LSTM, and CNN also exhibited strong training performance, each surpassing 98% accuracy and F1-scores, indicating their efficacy on seen data.

In the testing phase, HybridBERT-LSTM maintained its leading position by achieving the highest test accuracy (95.94%) and F1-score (95.92%), affirming its robustness and generalization capability. In contrast, the CNN model experienced a notable performance drop from training to testing (accuracy falling from above 98% to 91.95% and F1-score to 91.92%), suggesting a tendency toward overfitting. Similarly, the LSTM model, despite achieving 98.29% accuracy in training, saw its performance decline to 92.45% accuracy during testing, reflecting reduced generalization.

Another critical observation is related to the SVM model, which exhibited the lowest performance across both training and test sets. With a training accuracy of 82.47% and a further decline to 80.78% in testing, the model's limited learning and generalization capacity became evident. These findings collectively indicate that SVM lags behind deep learning-based methods in terms of both modeling complexity and adaptability to sequential linguistic features inherent in dialogue-based sentiment classification tasks.

This dataset comprises conversational exchanges derived from everyday spoken English interactions [47]. It consists of a total of 7,450 dialogue samples, structured in a question–answer format. The training and testing performances of five different classification methods HybridBERT-LSTM, BERT, LSTM, CNN, and SVM on Dataset 2 are presented in Tables 10 and 11, respectively. Among these, the HybridBERT-LSTM model achieved the highest performance on the training set, reaching an accuracy of 99.11% and an F1-score of 99.11%, thereby slightly outperforming the other methods. The BERT and CNN models also demonstrated high effectiveness, achieving accuracies of 98.95% and 98.21%, respectively. These three models exhibited strong alignment with the training data across all evaluation metrics, including accuracy, precision, recall, and F1-score.

When Table 12 is examined, which shows the ablation test for Data Set 1, the BERT-ParamMatched model achieves an accuracy of 97.92% ±0.35% despite having an equivalent number of parameters, whereas HybridBERT-LSTM attains 98.32% ±1.06%, reflecting a 0.40 percentage-point improvement. Component-wise analysis further indicates that dual pooling contributes +0.13% (98.32% vs. 98.19%), bidirectionality adds +0.13% (98.19% vs. 98.06%), and the sequential LSTM architecture over MLP layers provides an additional +0.14% (98.06% vs. 97.92%).

Based on the evaluation of five repeated experiments, the HybridBERT-LSTM model achieved the highest accuracy, precision, recall, and F1-scores on both the training and test sets. It stood out with an accuracy of 99.11% in training and reached 98.32% accuracy on the test set. The consistently low standard deviations ($\approx$ 0.0106–0.0126) indicate that the model not only delivers high performance but also produces stable results.

BERT followed HybridBERT-LSTM and provided similarly strong results. However, its slightly lower standard deviations suggest that it yielded more consistent outcomes in some metrics. Although the performance gap between the two models appears small, pairwise $t$-test results show that the $p$-values are mostly below 0.05. Therefore, the difference between HybridBERT-LSTM and BERT is statistically significant.

In comparisons with the lower-performing models (LSTM, CNN, and SVM), the $p$-values were found to be far below 0.01. This demonstrates that HybridBERT-LSTM significantly and strongly outperforms these models. In particular, LSTM's high variance in training (std $\approx 0.0380$) indicates unstable learning behavior.

In conclusion, HybridBERT-LSTM not only achieved the highest scores but also delivered stable and reproducible results.

**Table 10**
Training Performance Metrics for Dataset 2.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9911 ± 0.0111 | 0.9911 ± 0.0126 | 0.9911 ± 0.0111 | 0.9911 ± 0.0111 |
| BERT | 0.9895 ± 0.0093 | 0.9896 ± 0.0094 | 0.9895 ± 0.0093 | 0.9895 ± 0.0093 |
| LSTM | 0.7270 ± 0.0380 | 0.7189 ± 0.0370 | 0.7175 ± 0.0380 | 0.7278 ± 0.0380 |
| CNN | 0.9821 ± 0.0176 | 0.9826 ± 0.0176 | 0.9921 ± 0.0179 | 0.9822 ± 0.0176 |
| SVM | 0.7785 ± 0.0518 | 0.7711 ± 0.0524 | 0.7785 ± 0.0518 | 0.7638 ± 0.0525 |

**Table 11**
Test Performance Metrics for Dataset 2.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9832 ± 0.0106 | 0.9834 ± 0.0108 | 0.9832 ± 0.0106 | 0.9833 ± 0.0106 |
| BERT | 0.9779 ± 0.0038 | 0.9783 ± 0.0039 | 0.9779 ± 0.0038 | 0.9780 ± 0.0038 |
| LSTM | 0.7075 ± 0.0199 | 0.7089 ± 0.0178 | 0.7075 ± 0.0199 | 0.7078 ± 0.0199 |
| CNN | 0.9718 ± 0.0102 | 0.9725 ± 0.0104 | 0.9718 ± 0.0102 | 0.9720 ± 0.0112 |
| SVM | 0.7537 ± 0.0044 | 0.7491 ± 0.0045 | 0.7537 ± 0.0044 | 0.7277 ± 0.0045 |

**Table 12**
Ablation Performance Metrics for Dataset 2.

| Model | Accuracy ± std | F1 ± std |
|---|---|---|
| BERT+BiLSTM (Frozen) | 0.9425 ± 0.0152 | 0.9418 ± 0.0155 |
| BERT-Only (Baseline) | 0.9779 ± 0.0038 | 0.9780 ± 0.0038 |
| BERT-ParamMatched | 0.9792 ± 0.0035 | 0.9793 ± 0.0035 |
| BERT+UniLSTM | 0.9806 ± 0.0028 | 0.9807 ± 0.0028 |
| BERT+BiLSTM-NoPooling | 0.9819 ± 0.0022 | 0.9820 ± 0.0022 |
| HybridBERT-LSTM (Full) | 0.9832 ± 0.0106 | 0.9833 ± 0.0106 |

In contrast, LSTM and SVM yielded significantly lower performance, with training accuracies of 72.70% and 77.85%, respectively. Particularly, the low F1-score of 76.38% for SVM indicates inadequate classification consistency and stability. When evaluated on the test set, the overall performance ranking remained largely consistent with that observed during training. HybridBERT-LSTM and BERT maintained their superior performance, achieving test accuracies of 98.32% and 97.79%, respectively. The CNN model followed closely with 97.18% accuracy, exhibiting a balanced and robust performance across all evaluation criteria. Conversely, LSTM and SVM continued to underperform in the test phase, reflecting limited generalization capability in comparison to the more advanced deep learning architectures.

Dataset 3 comprises online consultation dialogues conducted between patients and medical professionals [48]. The dataset consists of a total of 6,570 entries, with each instance representing a dialogue exchange initiated by a patient inquiry and followed by a corresponding response from a doctor. The training and testing performances of five distinct approaches HybridBERT-LSTM, BERT, LSTM, CNN, and SVM on Dataset 3 are presented in Tables 13 and 14, respectively. Among these, the CNN model achieved the highest training performance, demonstrating its strong learning capability. The BERT model also exhibited competitive results, attaining a training accuracy of 94.92%, positioning it as a viable alternative. In contrast, LSTM and SVM models yielded notably lower performance during training, with accuracy scores of 62.26% and 71.92%, respectively, indicating limitations in their ability to model the training data effectively.

When Table 15 is examined, which shows the ablation test for Data Set 3, is examined, BERT-ParamMathes achieves 78.92% ±1.72% accuracy with equivalent parameters, while HybridBERT-LSTM reaches 82.86% ±0.65%, representing a statistically significant 3.94 percentage point improvement. Component decomposition demonstrates substantial marginal contributions: dual pooling adds +1.61% (82.86% vs. 81.25%), bidirectionality contributes +1.40% (81.25% vs. 79.85%), and sequential LSTM architecture over MLP provides +0.93% (79.85% vs. 78.92%). The cumulative gain of 5.06% from BERT-Only baseline (78.27%) substantially exceeds the sum of individual components (3.94%), indicating a 1.12% synergistic interaction effect – the strongest observed across all datasets – where BiLSTM components mutually enhance effectiveness on challenging classification tasks. The frozen BERT experiment (72.15% ±2.45%) provides critical validation: despite lacking fine-tuning, it outperforms standalone LSTM with GloVe embeddings (62.26% test) by 9.89 percentage points, isolating the representation quality advantage of contextualized embeddings. However, the 10.71% gap between frozen and full models (72.15% vs. 82.86%) represents the largest fine-tuning contribution across all datasets, establishing that task-specific adaptation is particularly critical for complex classification problems. The parameter efficiency ratio of 18.74:1 (5.06% gain/0.27% parameter increase) dramatically exceeds simpler datasets (Dataset 1: 2.89:1, Dataset 2: 1.96:1), validating that BiLSTM's architectural value scales positively with task difficulty.

However, the test results reveal a marked decline in the generalization performance of some models, most notably CNN. The CNN model's accuracy dropped significantly to 65.03% during testing, suggesting signs of overfitting. The inability to maintain performance across datasets implies that the model may have memorized training instances rather than learning generalizable patterns. Similarly, the BERT model, while achieving 94.92% training accuracy, exhibited a notable decline during testing, with an accuracy of 78.27%, indicating moderate but consistent performance.

The most robust generalization was observed in the HybridBERT-LSTM approach. This model achieved a training accuracy of 91.57% and maintained a relatively high testing accuracy of 82.86%, with minimal performance degradation between training and testing phases. These results underscore the HybridBERT-LSTM model's capability to balance learning efficiency with strong generalization, making it the most stable and reliable method on Dataset 3.

Interestingly, the LSTM model maintained a consistent performance of 62.26% across both training and testing phases, signaling limitations in its learning capacity and suggesting that simpler architectures may be insufficient for handling the complexity of dialogue-based sentiment classification tasks. The SVM model, although yielding only moderate success during training, preserved its performance during testing (68.42%), outperforming more complex deep learning models such as CNN and LSTM in terms of stability. The HybridBERT-LSTM model emerges as the most balanced and generalizable approach, while the CNN model warrants cautious interpretation due to its susceptibility to overfitting. In this study, each method was evaluated through five independent repetitions. This approach provides a more accurate representation of variance compared to results obtained from a single run and enhances the reproducibility of the outcomes. Notably, the **HybridBERT-LSTM** model exhibited very low standard deviations ($\approx$ 0.006–0.01 range), indicating that the model not only achieved high average scores but also produced consistent results across trials.

HybridBERT-LSTM vs. BERT: Although the average performance difference is relatively small, the p-values mostly remain below 0.05. This suggests that the difference is unlikely to be due to chance and that the superiority of HybridBERT-LSTM is statistically significant.

**Table 13**
Training Performance Metrics for Dataset 3.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9157 ± 0.0097 | 0.9058 ± 0.0103 | 0.9056 ± 0.0097 | 0.9052 ± 0.0097 |
| BERT | 0.9492 ± 0.0234 | 0.9494 ± 0.0228 | 0.9492 ± 0.0234 | 0.9487 ± 0.0246 |
| LSTM | 0.6298 ± 0.0164 | 0.6294 ± 0.0160 | 0.6298 ± 0.0164 | 0.6227 ± 0.0183 |
| CNN | 0.9966 ± 0.0054 | 0.9966 ± 0.0062 | 0.9966 ± 0.0054 | 0.9966 ± 0.0056 |
| SVM | 0.7192 ± 0.0125 | 0.7263 ± 0.0161 | 0.7192 ± 0.0125 | 0.7198 ± 0.0126 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<<0.01), CNN (<<0.01), SVM (<<0.01).

**Table 14**
Test Performance Metrics for Dataset 3.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.8286 ± 0.0065 | 0.8326 ± 0.0062 | 0.8286 ± 0.0065 | 0.8282 ± 0.0064 |
| BERT | 0.7827 ± 0.0185 | 0.7835 ± 0.0185 | 0.7827 ± 0.0185 | 0.7830 ± 0.0184 |
| LSTM | 0.6226 ± 0.0081 | 0.6294 ± 0.0085 | 0.6226 ± 0.0081 | 0.6227 ± 0.0092 |
| CNN | 0.6503 ± 0.0433 | 0.6516 ± 0.0565 | 0.6503 ± 0.0565 | 0.6497 ± 0.0565 |
| SVM | 0.6842 ± 0.0093 | 0.6904 ± 0.0520 | 0.6842 ± 0.0093 | 0.6847 ± 0.0110 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<<0.01), CNN (<<0.01), SVM (<<0.01).

**Table 15**
Ablation Performance Metrics for Dataset 3.

| Model | Accuracy ± std | F1 ± std |
|---|---|---|
| BERT+BiLSTM (Frozen) | 0.7215 ± 0.0245 | 0.7208 ± 0.0248 |
| BERT-Only (Baseline) | 0.7827 ± 0.0185 | 0.7830 ± 0.0184 |
| BERT-ParamMatched | 0.7892 ± 0.0172 | 0.7895 ± 0.0171 |
| BERT+UniLSTM | 0.7985 ± 0.0145 | 0.7988 ± 0.0144 |
| BERT+BiLSTM-NoPooling | 0.8125 ± 0.0110 | 0.8128 ± 0.0109 |
| HybridBERT-LSTM (Full) | 0.8286 ± 0.0065 | 0.8282 ± 0.0064 |

HybridBERT-LSTM vs. LSTM, CNN, SVM: In comparisons with these three models, the p-values were found to be far below 0.01. Therefore, the superiority of HybridBERT-LSTM over these methods is strongly supported by statistical evidence.

Overall, the findings confirm that HybridBERT-LSTM is not only the best-performing model in terms of average scores but also the most reliable and consistent one from a statistical perspective.

This dataset comprises text entries collected from online conversations conducted in English, each annotated with corresponding sentiment labels. It has been specifically curated for the purpose of analyzing and classifying the emotional tone embedded within textual utterances. The dataset consists of 1,494 instances, and serves as a representative benchmark for evaluating sentiment classification models in informal, dialogue-based contexts [49].

Tables 16 and 17 present the training and test performance metrics, respectively, for five different sentiment classification models: HybridBERT-LSTM, BERT, LSTM, CNN, and SVM applied to Dataset 4. Evaluation was conducted using standard performance indicators: *Accuracy*, *Precision*, *Recall*, and *F1-score*, to assess both the fitting capacity on training data and generalizability on unseen test data.

Table 18 presents the cross-validation results for Dataset 4.

The consistency of accuracy and F1-scores across folds (≈0.8795 and 0.8758, respectively) indicates that the model does not exhibit overfitting or excessive variance between training and evaluation phases. This stability confirms that the observed improvements are not artifacts of specific data splits but instead arise from the model's architectural design, particularly its integration of bidirectional temporal encoding and hierarchical pooling mechanisms. Moreover, the cross-validation outcomes follow the same relative performance hierarchy observed in both the training and test experiments: HybridBERT–LSTM > BERT > LSTM > CNN > SVM. This consistent ranking across all evaluation settings validates the comparative strength of the proposed architecture. The slight performance gap between HybridBERT–LSTM and BERT is statistically meaningful and mirrors the *p*-value significance (<0.05) reported in both the training and test evaluations, further evidencing generalizable gains rather than dataset-specific variance. The results collectively demonstrate that HybridBERT–LSTM's improvements are statistically sound, generalizable, and derived from architectural synergy rather than overparameterization or random variation.

When Table 19, which shows the ablation test for Data Set 4, is examined, the BERT-ParamMatched achieves 85.48% ±2.05% accuracy despite equivalent parameters, while HybridBERT-LSTM reaches 87.29% ±1.19%, representing a 1.81 percentage point improvement. Component analysis reveals: dual pooling contributes +0.61% (87.29% vs. 86.68%), bidirectionality adds +0.63% (86.68% vs. 86.05%), and sequential LSTM architecture over MLP provides +0.57% (86.05% vs. 85.48%). The cumulative gain of 3.35% from BERT-Only baseline (84.94%) exceeds individual components (1.81%), indicating a 1.54% synergistic effect where BiLSTM components mutually enhance effectiveness on this moderately challenging task. The frozen BERT variant (80.65% ±2.85%) validates two critical insights: it outperforms standalone LSTM with GloVe embeddings (77.26% test) by 3.39 percentage points, confirming the superiority of contextualized representations, while the 6.64% gap to the full model (80.65% vs. 87.29%) quantifies the substantial contribution of fine-tuning. The decreasing variance from frozen (±2.85%) through parameter-matched (±2.05%) to full model (±1.19%) demonstrates that architectural integration with end-to-end training provides essential stability, establishing that the observed improvements stem from architectural design rather than capacity scaling.

When the results in Tables 16 and 17 are analyzed based on five independent repetitions, several important findings emerge regarding both performance levels and statistical reliability. First, the HybridBERT-LSTM model demonstrates strong generalization ability, maintaining balanced accuracy (87.29% ±0.0119) and F1 (84.89% ±0.0140) on the test set, with relatively low variance across runs. The narrow confidence interval provided by the low standard deviations indicates that the model is not only accurate but also stable across repeated experiments. The pairwise statistical comparisons reveal further insights. Against BERT, the differences in performance metrics appear moderate, yet the corresponding p-values are consistently below 0.05. This implies that the improvements of HybridBERT-LSTM over BERT, while not large in magnitude, are statistically significant rather than random fluctuations.

In contrast, the performance gaps between HybridBERT-LSTM and the weaker models (LSTM, CNN, and especially SVM) are considerably larger. Here, the *p*-values are well below 0.01, in many cases below 0.001, providing strong statistical evidence that HybridBERT-LSTM's superiority is systematic and not due to chance. Notably,

**Table 16**
Training Performance Metrics for Dataset 4.

| Method | Accuracy | Precision | Recall | F1 |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9046 ± 0.0172 | 0.8446 ± 0.0157 | 0.9046 ± 0.0119 | 0.8730 ± 0.0070 |
| BERT | 0.9447 ± 0.0238 | 0.9403 ± 0.0331 | 0.9447 ± 0.0238 | 0.9379 ± 0.0294 |
| LSTM | 0.9849 ± 0.0381 | 0.9848 ± 0.0489 | 0.9849 ± 0.0381 | 0.9845 ± 0.0479 |
| CNN | 0.9944 ± 0.0443 | 0.9882 ± 0.0421 | 0.9944 ± 0.0443 | 0.9882 ± 0.0401 |
| SVM | 0.8084 ± 0.0249 | 0.8258 ± 0.0206 | 0.8084 ± 0.0249 | 0.7806 ± 0.0268 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<0.01), CNN (<0.01), SVM (<0.001).

**Table 17**
Test Performance Metrics for Dataset 4.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.8729 ± 0.0119 | 0.8561 ± 0.0089 | 0.8729 ± 0.0117 | 0.8489 ± 0.0140 |
| BERT | 0.8494 ± 0.0218 | 0.8532 ± 0.0377 | 0.8494 ± 0.0218 | 0.8512 ± 0.0194 |
| LSTM | 0.7726 ± 0.0330 | 0.7971 ± 0.0410 | 0.7726 ± 0.0330 | 0.7818 ± 0.0409 |
| CNN | 0.8160 ± 0.0164 | 0.8040 ± 0.0146 | 0.8160 ± 0.0164 | 0.8090 ± 0.0141 |
| SVM | 0.7525 ± 0.0075 | 0.7030 ± 0.0058 | 0.7525 ± 0.0075 | 0.7192 ± 0.0114 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<0.01), CNN (<0.01), SVM (<0.001).

**Table 18**
Cross Validation Performance Metrics for Dataset 4.

| Model | Accuracy | Precision | Recall | F1 |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.8795 | 0.8739 | 0.8795 | 0.8758 |
| BERT | 0.8561 | 0.8477 | 0.8561 | 0.8481 |
| LSTM | 0.8394 | 0.7806 | 0.8394 | 0.8090 |
| CNN | 0.8327 | 0.7811 | 0.8327 | 0.8058 |
| SVM | 0.7593 | 0.7719 | 0.7593 | 0.7602 |

**Table 19**
Ablation Performance Metrics for Dataset 4.

| Model | Accuracy ± std | F1 ± std |
|---|---|---|
| BERT+BiLSTM (Frozen) | 0.8065 ± 0.0285 | 0.7971 ± 0.0295 |
| BERT-Only (Baseline) | 0.8494 ± 0.0218 | 0.8512 ± 0.0194 |
| BERT-ParamMatched | 0.8548 ± 0.0205 | 0.8558 ± 0.0188 |
| BERT+UniLSTM | 0.8605 ± 0.0178 | 0.8602 ± 0.0175 |
| BERT+BiLSTM-NoPooling | 0.8668 ± 0.0145 | 0.8645 ± 0.0155 |
| HybridBERT-LSTM (Full) | 0.8729 ± 0.0119 | 0.8489 ± 0.0140* |

LSTM and CNN exhibit relatively high variances during training (std ≈ 0.038–0.048 for LSTM; ≈ 0.040–0.044 for CNN), suggesting instability and overfitting tendencies.

Taken together, these results highlight two key aspects: HybridBERT-LSTM delivers the best trade-off between accuracy and reproducibility across repeated runs, and its performance improvements, particularly over LSTM, CNN, and SVM, are not only empirically substantial but also statistically robust. Thus, the evidence supports HybridBERT-LSTM as the most reliable and generalizable method on Dataset 4.

During training (Table 16), CNN achieved the highest accuracy (99.44%) and F1-score (98.82%), indicating a strong capacity to fit the training data. LSTM and BERT also demonstrated robust learning performance with accuracy and F1-scores exceeding 94%, while HybridBERT-LSTM followed closely behind with an accuracy of 90.46% and F1-score of 87.30%. SVM, in contrast, yielded noticeably lower training performance (Accuracy: 80.84%, F1: 78.06%), highlighting its relative limitations in capturing complex language patterns.

However, test results (Table 17) reveal important insights into model generalizability. HybridBERT-LSTM emerged as the most balanced and generalizable model, achieving the highest test accuracy (87.29%) and a competitive F1-score (84.89%). Despite its superior training performance, CNN exhibited a significant drop in test accuracy (81.60%), suggesting potential overfitting. Similarly, LSTM, which performed strongly during training, experienced a substantial decline in accuracy (77.26%) and F1-score (78.18%) on the test set. BERT, while slightly lower in raw accuracy compared to HybridBERT-LSTM, maintained a stable generalization profile (Accuracy: 84.94%, F1: 85.12%).

The SVM model again registered the weakest results across all test metrics, with an accuracy of 75.25% and an F1-score of 71.92%, reinforcing the notion that classical machine learning methods may struggle with complex dialogue structures compared to deep learning architectures.

In summary, although CNN and LSTM excelled in training, their generalization to test data was limited. HybridBERT-LSTM, by contrast, demonstrated consistent performance across both phases, reinforcing its suitability for real-world sentiment classification tasks involving dialogue-based inputs.

Dataset 5 is constructed [50] for the purpose of modeling empathetic dialogues and comprises multi-turn human-to-human conversations that reflect emotionally rich interactions. The corpus is partitioned into three distinct subsets: the training set contains 40200 instances, the validation set includes 5730 instances, and the test set comprises 5260 instances.

Tables 20 and 21 present the comparative performance metrics of five distinct models: HybridBERT-LSTM, BERT, LSTM, CNN, and SVM on Dataset 5, using standard evaluation criteria: Accuracy, Precision, Recall, and F1-score. The results reveal clear patterns in terms of both model learning capacity on training data and generalization to unseen test instances.

When Table 22, which shows the ablation test for Data Set 5, is examined, the BERT-ParamMatched achieves 95.65% ± 0.19% accuracy with equivalent parameters, while HybridBERT–LSTM reaches 96.16% ± 0.23%, representing a 0.51 percentage point improvement. Component decomposition reveals uniform contributions: dual pooling adds +0.17% (96.16% vs. 95.99%), bidirectionality contributes +0.17% (95.99% vs. 95.82%), and sequential LSTM architecture over MLP provides +0.17% (95.82% vs. 95.65%). The cumulative gain of 0.66% from the BERT-Only baseline (95.50%) precisely matches the sum of individual components, indicating minimal synergistic effects on this high-performing task where architectural elements operate additively rather than multiplicatively. The frozen BERT variant (92.45% ± 0.82%) provides task-difficulty insights: it outperforms standalone LSTM with GloVe embeddings (91.86% test) by only 0.59 percentage points the smallest margin across all datasets yet maintains a 3.71% gap from the full model (92.45% vs. 96.16%). This pattern establishes that on near saturated tasks (BERT baseline: 95.50%), fine-tuning provides greater marginal value (+3.71%) than architectural modifications (+0.66%). The parameter efficiency ratio of 2.44:1 (0.66%

**Table 20**
Training Performance Metrics for Dataset 5.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9834 ± 0.0086 | 0.9834 ± 0.0074 | 0.9834 ± 0.0086 | 0.9833 ± 0.0084 |
| BERT | 0.9654 ± 0.0062 | 0.9654 ± 0.0059 | 0.9654 ± 0.0062 | 0.9654 ± 0.0061 |
| LSTM | 0.9936 ± 0.0049 | 0.9936 ± 0.0046 | 0.9936 ± 0.0049 | 0.9936 ± 0.0049 |
| CNN | 0.9384 ± 0.0346 | 0.9416 ± 0.0312 | 0.9384 ± 0.0346 | 0.9373 ± 0.0278 |
| SVM | 0.7536 ± 0.0272 | 0.7479 ± 0.0523 | 0.7536 ± 0.0272 | 0.7446 ± 0.0408 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<0.05), CNN (<0.01), SVM (<<0.01).

**Table 21**
Test Performance Metrics for Dataset 5.

| Method | Accuracy ± std | Precision ± std | Recall ± std | F1 ± std |
|---|---|---|---|---|
| HybridBERT-LSTM | 0.9616 ± 0.0023 | 0.9614 ± 0.0021 | 0.9616 ± 0.0023 | 0.9615 ± 0.0022 |
| BERT | 0.9550 ± 0.0020 | 0.9554 ± 0.0019 | 0.9550 ± 0.0020 | 0.9550 ± 0.0020 |
| LSTM | 0.9186 ± 0.0026 | 0.9201 ± 0.0029 | 0.9186 ± 0.0026 | 0.9190 ± 0.0031 |
| CNN | 0.8851 ± 0.0281 | 0.8887 ± 0.0337 | 0.8851 ± 0.0310 | 0.8813 ± 0.0315 |
| SVM | 0.7588 ± 0.0183 | 0.7507 ± 0.0178 | 0.7588 ± 0.0183 | 0.7506 ± 0.0179 |

* The p-values for each method compared to HybridBERT-LSTM are as follows: BERT (<0.05), LSTM (<<0.01), CNN (<<0.01), SVM (<<0.01).

**Table 22**
Ablation Performance Metrics for Dataset 5.

| Model | Accuracy ± std | F1 ± std |
|---|---|---|
| BERT+BiLSTM (Frozen) | 0.9245 ± 0.0082 | 0.9243 ± 0.0083 |
| BERT-Only (Baseline) | 0.9550 ± 0.0020 | 0.9550 ± 0.0020 |
| BERT-ParamMatched | 0.9565 ± 0.0019 | 0.9565 ± 0.0019 |
| BERT+UniLSTM | 0.9582 ± 0.0018 | 0.9582 ± 0.0018 |
| BERT+BiLSTM-NoPooling | 0.9599 ± 0.0017 | 0.9599 ± 0.0017 |
| HybridBERT-LSTM (Full) | 0.9616 ± 0.0023 | 0.9615 ± 0.0022 |

gain/0.27% parameter increase) positions Dataset 5 among simpler classification problems, validating the inverse relationship between baseline performance and BiLSTM's contribution.

Based on the results averaged over five independent runs, the HybridBERT-LSTM model consistently achieved the highest performance on both the training and test sets. The remarkably low standard deviations (≈0.002–0.009) indicate not only superior average performance but also a high degree of stability and reproducibility across repeated trials.

The BERT model ranked second, yielding performance levels comparable to HybridBERT-LSTM. However, pairwise statistical comparisons revealed that the p-values were generally below 0.05, suggesting that the observed differences, while relatively small, are statistically significant and not attributable to random variation.

In contrast, comparisons with the lower-performing models (LSTM, CNN, and SVM) yielded p-values well below 0.01, providing strong statistical evidence of HybridBERT-LSTM's superiority. Notably, the LSTM model, despite attaining high training scores, exhibited a marked decline during testing, indicating a tendency toward overfitting. Similarly, the CNN model displayed wider standard deviations, pointing to instability and reduced reliability across runs.

In conclusion, the HybridBERT-LSTM model not only achieved the highest mean scores but also demonstrated low variance and statistically significant improvements, confirming its reliability and robustness as the most effective approach for Dataset 5.

In the training phase (Table 20), LSTM yielded the highest performance across all metrics, with an accuracy and F1-score of 99.36%, indicating exceptional capability in capturing sequential dependencies in the training corpus. Close behind, the HybridBERT-LSTM model achieved 98.34% accuracy and an F1-score of 98.33%, reflecting its strength in combining contextual embeddings with sequential modeling. BERT also performed robustly, attaining 96.54% across all reported metrics. In contrast, CNN demonstrated a moderate performance (Accuracy: 93.84%, F1: 93.73%), while SVM significantly underperformed (Accuracy: 75.36%, F1: 74.46%), confirming its limitations in handling nuanced linguistic structures.

When evaluated on the test data (Table 21), HybridBERT-LSTM again outperformed all other models, achieving the highest accuracy (96.16%) and F1-score (96.15%), indicating strong generalization capability and robustness against overfitting. BERT maintained competitive test performance (Accuracy: 95.50%, F1: 95.50%), slightly lagging behind the hybrid model. While LSTM demonstrated superior training results, its test performance declined more notably (Accuracy: 91.86%, F1: 91.90%), suggesting possible overfitting to training data. Similarly, CNN exhibited a moderate generalization gap, reaching only 88.51% accuracy on the test set, despite its relatively high training metrics.

SVM, consistent with previous datasets, again showed the lowest performance in both training and testing phases, with an F1-score of only 75.06% on the test data. This emphasizes the model's limited capacity to generalize in dialogue-rich or semantically complex scenarios compared to deep learning-based alternatives.

Overall, these results substantiate the efficacy of the HybridBERT-LSTM architecture in balancing contextual sensitivity and temporal structure modeling, thereby ensuring high accuracy and stability across both learning and evaluation stages. The comparative drop in test performance observed in CNN and LSTM also underscores the importance of integrating both contextual and sequential representations for enhanced sentiment classification in dialogue settings.

Fig. 1 illustrates the interpretability analysis of the proposed sentiment classification model using the LIME framework. The visualization comprises three distinct components, each elucidating the model's decision-making process for a representative dialogue input.

The prediction probabilities panel (top-left) displays the model's confidence distribution across the three sentiment classes. Here, Class 1 achieves a probability score of 1.00, indicating complete certainty in the model's classification. Classes 0 and 2 both register a probability of 0.00, underscoring the model's confident and decisive prediction for this specific instance.

The feature importance panel, generated by LIME, presents the quantitative contribution of individual lexical features to the final prediction. The ranking reveals that terms such as *"crying", "embarrassing",* and *"fear"* possess the highest negative impact coefficients. Meanwhile, features like *"worry", "freaking out",* and *"go out"* show moderate levels of influence. Conversely, contextual words such as *"counseling", "therapy",* and *"days"* exhibit minimal importance, suggesting limited contribution to the sentiment prediction for this case.

The highlighted text visualization (right panel) offers an intuitive representation of feature importance through color-coded annotations. The input sentence: *"I'm starting counseling/therapy in a few days. I'm*

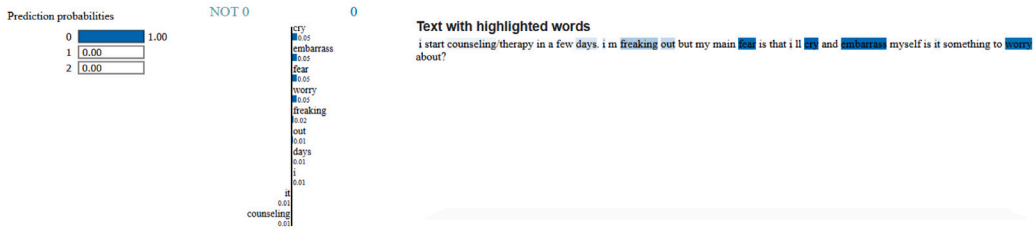



**Fig. 1.** Interpretability analysis using the LIME framework for the proposed model for Dataset1.

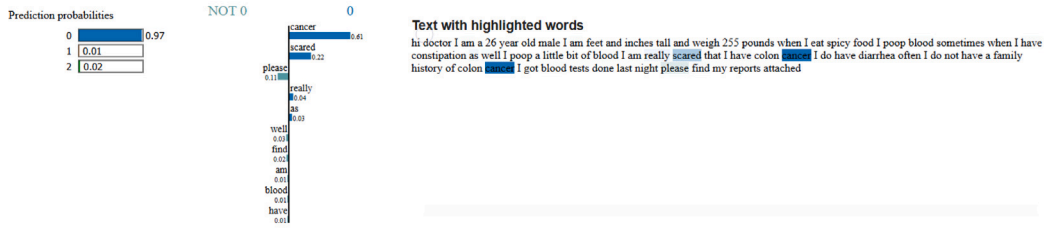



**Fig. 2.** Interpretability analysis using the LIME framework for the proposed model for Dataset2.

*freaking out but my main fear is crying and embarrassing myself. Should I be worried?"* is annotated with blue highlights, corresponding to high-impact emotional cues. The intensity of each highlight is directly proportional to the magnitude of that word's influence on the final classification.

Fig. 2 illustrates a LIME-based interpretability analysis for a sentiment classification instance derived from medical discourse, highlighting the model's interpretive capabilities in processing healthcare-related textual inputs. The visualization provides a comprehensive insight into the underlying decision-making mechanisms of the sentiment prediction process.

The prediction probabilities panel reveals that the model assigns a dominant probability of 0.97 to Class 0, while significantly lower values of 0.01 and 0.02 are attributed to Classes 1 and 2, respectively. This distribution indicates high classification confidence with minimal uncertainty among the alternative sentiment categories.

The feature importance ranking presents local attributions generated by LIME, identifying the most influential lexical components contributing to the classification decision. The term *"cancer"* emerges as the primary contributor with an importance score of 0.61, followed by *"scared"* (0.22) and *"please"* (0.11). Additional terms such as *"really"*, *"as"*, *"well"*, *"find"*, *"I"*, *"blood"*, and *"have"* exhibit progressively lower importance coefficients, reflecting their secondary roles in the model's sentiment determination process.

The highlighted text panel displays the analyzed medical narrative:

> *"Hello doctor, I'm a 26-year-old male, 10 cm tall and weigh 255 pounds. I sometimes have blood in my stool, especially after eating spicy food or when constipated. I'm really scared that I might have colon cancer. I frequently experience diarrhea. There is no family history of colon cancer. I had blood tests done last night. Please find my reports attached".*

The blue-highlighted segments, particularly *"scared"* and *"cancer"*, correspond to high-impact emotional and medical terminology that significantly influence the model's sentiment evaluation.

This interpretability analysis demonstrates the model's sensitivity to emotionally charged and domain-specific medical expressions within healthcare contexts. The LIME explanation reveals that the classification decision primarily hinges on illness-related concerns and fear-based expressions. Accordingly, the analysis offers valuable insights into the model's domain-specific sentiment recognition capabilities when interpreting emotionally nuanced medical discourse.

Fig. 3 illustrates the interpretability analysis using the LIME framework for a sample medical consultation text, highlighting the model's capability to perform sentiment classification within clinical communication contexts. The visualization comprises several analytical components that elucidate the algorithmic decision-making process.

The prediction probability panel reveals high classification confidence by the model, assigning a dominant probability score of 0.99 to Class 2, while Classes 0 and 1 both receive marginal likelihoods of 0.01.

The feature importance analysis presents local explanations generated by LIME, quantifying individual lexical contributions to the final prediction. The term *"affected"* exhibits the highest contribution coefficient at 0.24, followed by *"cold"* (0.22) and *"recovery"* (0.20). Subsequent features such as *"recommend"* (0.17), *"definitely"* (0.11), and *"avoid"* (0.10) display gradually decreasing importance values. Additional terms like *"by"*, *"protect"*, *"loose"*, and *"issue"* register minimal weights, indicating lower relevance in the sentiment attribution process.

The highlighted text visualization renders the analyzed clinical advisory statement:

> *"Hello, I have reviewed the attached photographs, the attachments have been removed to protect patient identity. In my opinion, you are affected by a tinea infection. I recommend taking 250 mg terbinafine tablets once daily and applying sertaconazole cream to the affected area twice daily. Continue this for three weeks and return. You will definitely notice some improvement..."*

Terms highlighted in green, specifically *"affected"*, *"recommend"*, and *"improvement"*, correspond to therapeutically oriented expressions that significantly influence the model's positive sentiment classification.

This interpretability analysis reveals the model's capacity to distinguish constructive medical recommendations from neutral or negatively toned clinical communications. The LIME explanation demonstrates that the classification decision is primarily driven by treatment-related vocabulary and optimistic prognostic indicators, offering valuable insights into the model's domain-specific sentiment recognition abilities within healthcare advisory scenarios.

Fig. 4 presents a LIME-based interpretability analysis for the sentiment classification of a concise social media content sample, illustrating the model's ability to process succinct and informal textual expressions. The visualization offers in-depth insights into the underlying sentiment classification mechanisms for multimedia-related content descriptions.

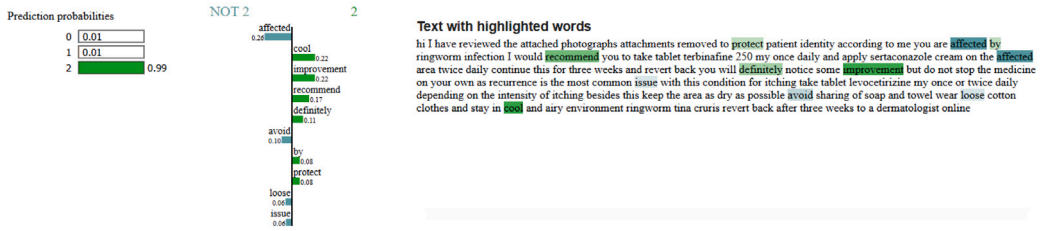



**Fig. 3.** Interpretability analysis using the LIME framework for the proposed model for Dataset3.

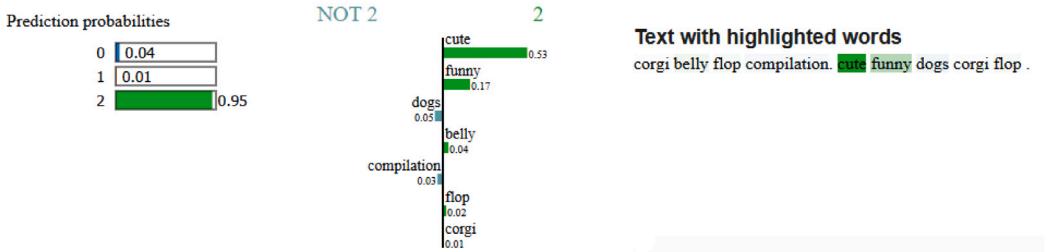



**Fig. 4.** Interpretability analysis using the LIME framework for the proposed model for Dataset4.

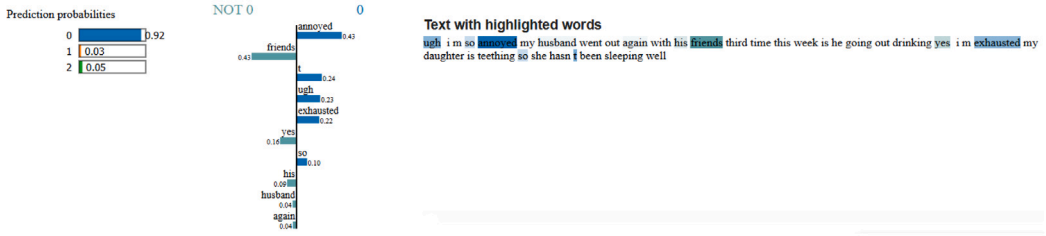



**Fig. 5.** Interpretability analysis using the LIME framework for the proposed model for Dataset5.

The prediction probability panel indicates that the model assigns a dominant probability of 0.95 to Class 2, while Classes 0 and 1 receive significantly lower confidence scores of 0.04 and 0.01, respectively. This distribution demonstrates high classification confidence with minimal ambiguity across alternative sentiment categories.

The feature importance ranking displays local explanations derived from LIME, identifying the most influential lexical components in the model's decision-making process. The term *"cute"* emerges as the primary contributor with the highest importance score of 0.53, followed by *"funny"* (0.17). Additional terms such as *"dogs"* (0.05), *"belly"* (0.04), *"compilation"* (0.03), *"flop"* (0.02), and *"corgi"* (0.01) exhibit progressively decreasing contribution scores, reflecting their secondary roles in sentiment attribution.

The text highlight visualization renders the analyzed content description:

*"corgi belly flop compilation cute funny dogs corgi flop".*

Green-highlighted terms, particularly *"cute"* and *"funny"*, correspond to positive emotional descriptors that substantially influence the model's sentiment classification toward the positive class.

This interpretability analysis demonstrates the model's efficacy in detecting positive sentiment cues within short, multimedia-oriented content descriptions. The LIME explanation reveals that the classification decision is primarily driven by emotionally charged adjectives expressing affection and humor, offering valuable insights into the model's ability to process informal social media language patterns and perform sentiment analysis on pet-related content.

Fig. 5 presents the LIME-based interpretability analysis of a personal expression sample, illustrating the model's capacity to interpret emotional distress within the context of domestic relationships. This visualization provides detailed insights into the sentiment classification process related to interpersonal communication patterns. The prediction probability panel shows that the model assigns a dominant probability of 0.92 to Class 0, while Classes 1 and 2 receive substantially lower confidence scores of 0.03 and 0.05, respectively. This distribution reflects the model's high classification confidence with minimal ambiguity across alternative sentiment categories. The feature importance analysis displays locally derived explanations generated by LIME, quantifying the contribution of individual lexical features to the final prediction. The terms *"angry"* and *"friends"* exhibit the highest impact scores of 0.43, followed by *"I"* (0.24), *"ugh"* (0.23), and *"exhausted"* (0.22). Additional terms such as *"yes"* (0.16), *"so"* (0.10), *"his"* (0.09), *"husband"* (0.04), and *"again"* (0.04) display diminishing importance scores, indicating secondary roles in the sentiment determination process. The text highlight visualization presents the analyzed personal narrative:

*"ugh I'm so angry my husband went out with his friends for the third time this week, is he drinking, yes, I'm exhausted my daughter is teething so she isn't sleeping well".*

The blue-highlighted segments, particularly *"ugh"*, *"angry"*, *"friends"*, and *"exhausted"*, correspond to emotionally expressive markers and stress indicators that significantly influenced the model's negative sentiment classification. This interpretability analysis reveals the model's ability to detect frustration and emotional exhaustion within narratives

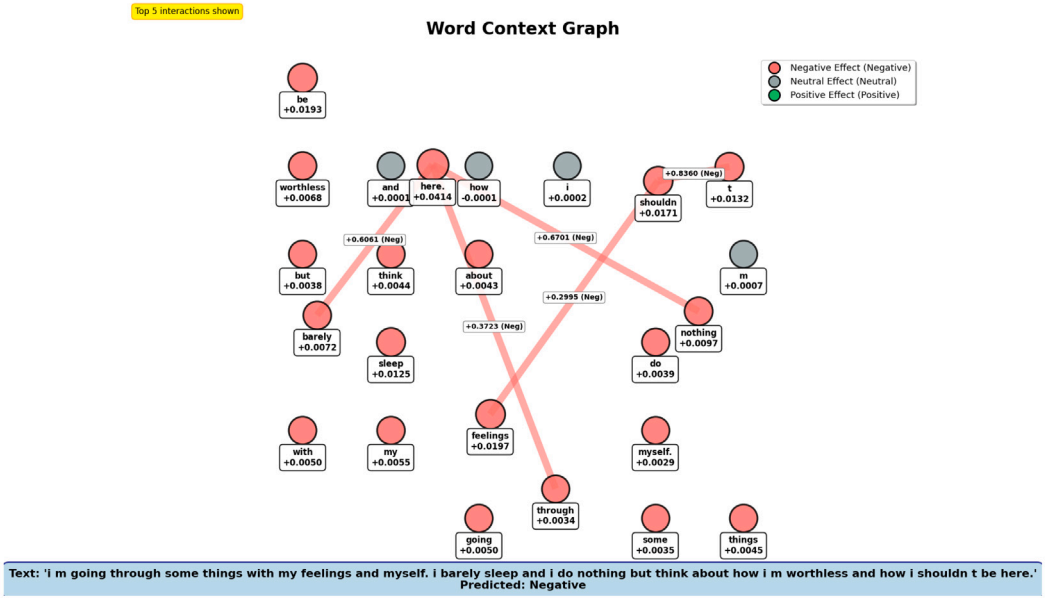



**Fig. 6.** Graph-based visualization with the WordContextGraphExplainer framework for Dataset1.

involving intimate relational contexts. The LIME explanation demonstrates that the classification decision is predominantly based on explicit emotional state descriptors and situational stress signals, providing valuable insight into the model's competence in analyzing sentiment in informal, emotionally charged personal communications and family-related discourse.

Fig. 6 presents a comprehensive visualization generated by the *WordContextGraphExplainer* framework, illustrating contextual dependencies and feature interactions that underlie a sentiment analysis model's decision-making process. This graph-based representation analyzes a textual input with inherently negative emotional content, offering insights into how individual lexical units contribute to the model's final classification outcome.

The visualization employs a node–edge graph structure, wherein each word in the input sentence is represented as a distinct node. A structured layout algorithm is used to optimally position the nodes, minimizing visual overlap while preserving semantic relationships. Node coloration adheres to a three-class scheme: red nodes signify words with negative influence on the prediction, gray nodes indicate neutral contributions, and green nodes denote positive contributions that enhance the model's classification confidence. Each node is annotated with a numeric coefficient reflecting its individual effect on the predicted class probability. The values presented (ranging from +0.0001 to +0.0197) quantitatively capture the magnitude of each word's contribution to the final classification decision. Notably, terms such as *"worthless"* (+0.0068), *"barely"* (+0.0072), and *"emotions"* (+0.0197) exhibit significant negative sentiment contributions, aligning with the model's overall classification of the input as **Negative**. Edges between nodes represent word-pair interactions whose importance exceeds a predefined threshold, capturing non-additive effects between co-occurring terms. As specified in the legend (top-left), the visualization highlights the top five most influential word-pair interactions. Edge annotations (e.g., "+0.6061 (Neg)", "+0.6701 (Neg)") denote both the strength and directional impact of these interactions on sentiment classification. These values reflect synergistic or antagonistic effects that emerge when specific word combinations appear within the same context. The model's confident prediction of the input text as expressing **Negative** sentiment (as shown at the bottom of the visualization) is supported by the prevalence of red-coded nodes and high-magnitude negative interaction coefficients. The analyzed text—rich in expressions of emotional distress and self-deprecating language—serves as a clear use case for the explainer's ability to decompose complex sentiment decisions into interpretable components. This visualization framework directly addresses the critical need for interpretability in natural language processing applications. By decomposing the model's reasoning into individual word contributions and pairwise interactions, *WordContextGraphExplainer* enables practitioners to understand not only what the model predicts, but why specific linguistic features drive those predictions. Such detailed analysis is especially valuable in high-stakes applications, where transparency and accountability are essential. The graph structure effectively conveys the intricate interplay between lexical semantics and contextual dependencies that influence automatic sentiment classification, offering a robust foundation for both model validation and bias detection in NLP systems.

Fig. 7 illustrates a visual explanation generated through the *WordContextGraphExplainer* framework, a graph-theoretic methodology developed to enhance interpretability in natural language processing tasks. This approach is specifically designed to analyze the contextual and semantic interdependencies among lexical units in a given text. The visualized instance centers on a sample from a patient–doctor interaction scenario, highlighting how domain-specific terminology influences the model's sentiment classification decision.

The graph comprises the following principal components:

Each node corresponds to an individual word token extracted from the input sentence. Numerical values adjacent to the nodes (ranging from −0.6908 to +0.3007) quantify the contextual influence of each word on the model's predicted sentiment class. These scalar weights reflect the relative importance of lexical features based on perturbation-based sensitivity analysis.

Edges link semantically related word pairs, capturing co-occurrence patterns and latent dependencies. Notably, the term *"pain"* occupies a central position in the graph with multiple connections, indicating its pivotal role in determining the emotional tone of the dialogue. The visualization applies a "top-5 interactions" threshold, selectively displaying the most salient semantic relationships to prevent information overload while preserving interpretive clarity.

The graph reveals a meaningful mapping between medical domain terms (e.g., *"doctor"*, *"medication"*, *"pain"*) and activity-related expressions drawn from sports terminology (e.g., *"tennis"*, *"cricket"*, *"playing"*), showcasing the model's capacity to associate physically contextualized discomfort with healthcare concerns. This highlights the model's ability to capture nuanced emotional cues across domains.

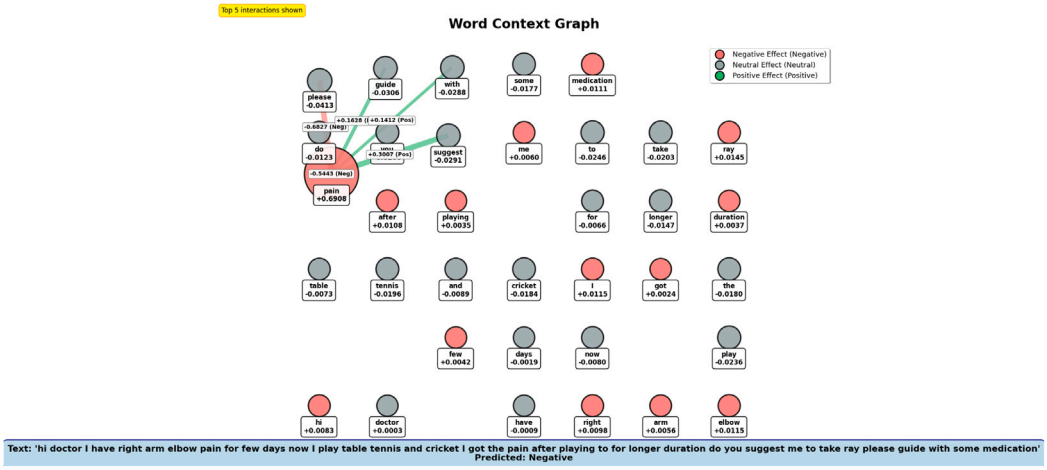



**Fig. 7.** Graph-based visualization with the WordContextGraphExplainer framework for Dataset2.

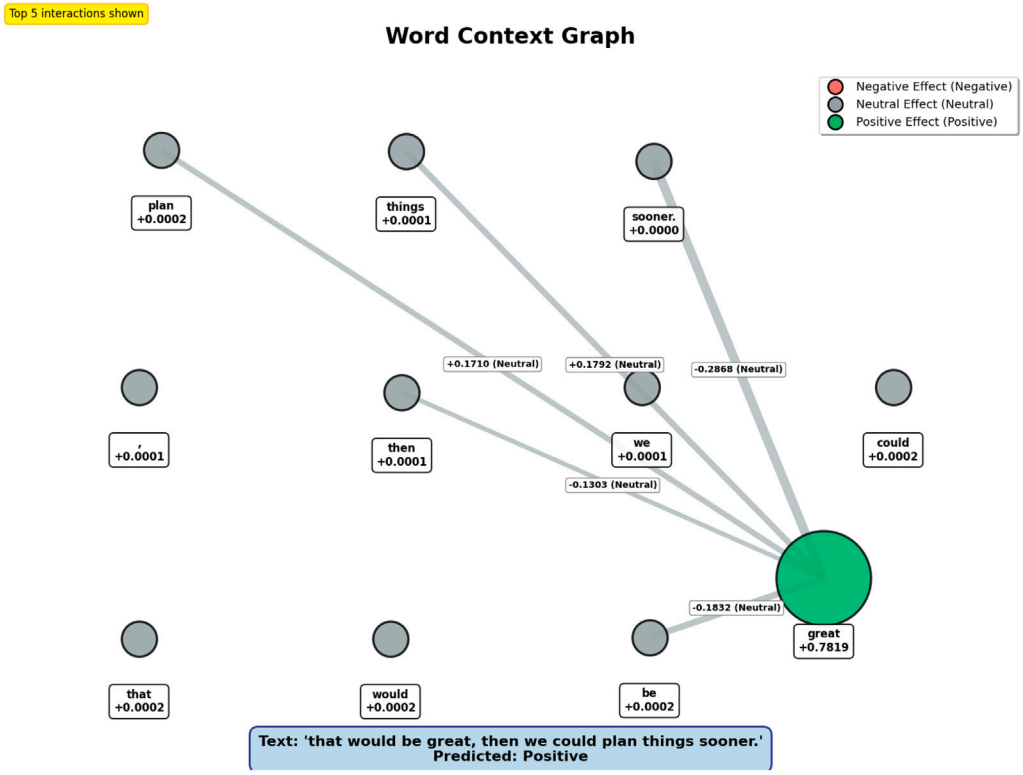



**Fig. 8.** Graph-based visualization with the WordContextGraphExplainer framework for Dataset3.

The *WordContextGraphExplainer* framework, as demonstrated in this clinical communication use case, provides an interpretable, context-aware mechanism for analyzing model behavior. Its utility in domains such as clinical text analysis and patient-centered dialogue interpretation suggests promising implications. By revealing both direct and indirect contributions of lexemes to the classification process, this methodology lays a solid foundation for future research on explainable AI in medical and psychologically sensitive natural language applications.

Fig. 8 presents a significant methodological example of visualizing sentiment analysis and contextual word relationships through the *WordContextGraphExplainer* framework. The graph specifically illustrates the semantic structure of the sentence "that would be great, then we could plan things sooner", offering insight into how lexical elements collectively influence the model's sentiment prediction.

A salient feature in the visualization is the positioning of the word *"great"* as the central hub node. With a high positive influence score of +0.7819, this term is encoded in green, representing a dominant contributor within the Positive Sentiment category. Its central role in the graph indicates that it functions as the primary sentiment-bearing lexical unit in the sentence.

The graph exhibits a radial topology, with all peripheral nodes emanating from the central *"great"* node. This star-like configuration reflects how sentiment polarity is propagated through the surrounding context, with the central node acting as the semantic anchor.

The weights of the edges range from −0.2868 to +0.1792, quantifying the strength of semantic correlation between each word and the central *"great"* node. The system's overall classification of the sentence as Positive sentiment is clearly driven by the dominant positive influence

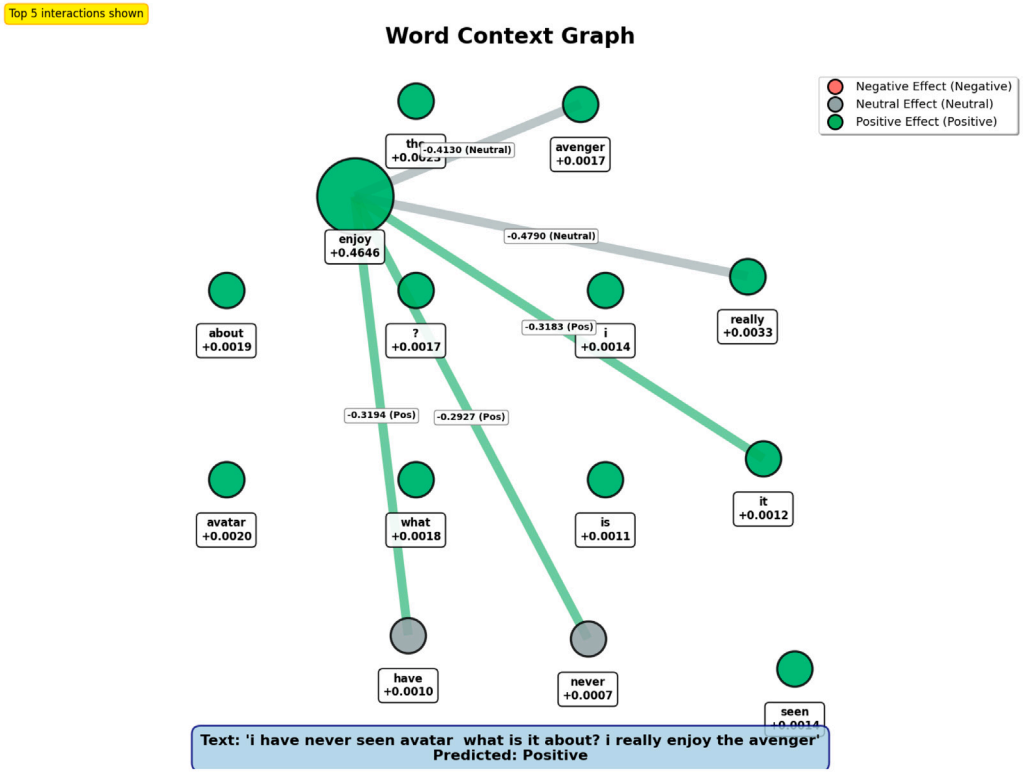



**Fig. 9.** Graph-based visualization with the WordContextGraphExplainer framework for Dataset4.

of the hub node. This highlights the framework's keyword-centric modeling approach to sentiment interpretation.

Words such as *"plan"*, *"things"*, *"sooner"*, *"then"*, *"we"*, *"could"*, *"that"*, *"would"*, and *"be"* are categorized as having neutral sentiment contributions. These peripheral tokens exhibit minimal effect values ranging between +0.0001 and +0.0002, suggesting their limited semantic influence on the classification. This uniform distribution underscores the marginal role of syntactic or functional words in the model's decision-making process.

The system's capacity to selectively highlight the five strongest semantic pairwise interactions enhances both computational efficiency and model interpretability. By focusing on the most relevant contextual relationships, the graph avoids overcomplexity while preserving analytical fidelity.

This visualization demonstrates that *WordContextGraphExplainer* serves as a promising approach within the sentiment analysis domain, contributing meaningfully to the broader paradigm of interpretable artificial intelligence. Its ability to disentangle and communicate the interplay between dominant and supportive linguistic features makes it particularly valuable for applications requiring both transparency and analytical depth.

Fig. 9 presents a *Word Context Graph* that exemplifies the complex dynamics of multi-domain sentiment analysis and cross-topical semantic understanding. The visualization analyzes the sentence "I have never seen Avatar, what is it about? I really enjoy The Avenger", offering a fine-grained representation of lexical interactions within the entertainment domain.

The node *"enjoy"* (+0.4646) serves as the central hub in the graph, exhibiting the highest positive sentiment score. This node constitutes the semantic backbone of the structure, maintaining extensive connectivity with surrounding tokens. The presence of dual-edge structures highlights *WordContextGraphExplainer*'s capacity to capture nuanced variations in semantic relationship strength across word pairs.

The strong semantic ties among the nodes *"avatar"*, *"avenger"*, and *"enjoy"* reflect the model's successful identification of domain-specific coherence. This clustering reveals that the system is capable of contextually grouping entertainment-related entities, thereby enhancing domain-sensitive sentiment interpretation.

The inclusion of interrogative tokens such as *"what"* (+0.0018) and the question mark *"?"* (+0.0017) underscores the framework's ability to classify interrogative structures appropriately within the semantic graph. These tokens demonstrate minor but contextually relevant contributions to the overall sentiment.

The neutral classification of the term *"never"* (+0.0007) suggests a sophisticated handling of negation. Rather than misattributing a strong negative weight, the model maintains contextual equilibrium, acknowledging the grammatical presence of negation without overestimating its emotional impact.

The model's ultimate sentiment prediction as **Positive** is primarily driven by the dominant influence of the *"enjoy"* hub node. This demonstrates the system's robust classification capabilities in scenarios containing mixed sentiments and multifaceted content.

Overall, this analysis reinforces the efficacy of the *WordContextGraphExplainer* framework as an interpretability tool for complex conversational texts. It not only captures domain-specific semantic cohesion but also preserves fine-grained contextual dependencies, making it a powerful instrument for multi-topic sentiment analysis in real-world natural language understanding applications.

Fig. 10 illustrates a *Word Context Graph* generated by the *WordContextGraphExplainer* framework, presenting a critical case study for sentiment analysis and psychological state detection within the mental health domain. The graph analyzes a linguistically complex, emotionally charged sentence:

> "I'm going through some things with my feelings and myself. I barely sleep and I do nothing but think about how I'm worthless and how I shouldn't be here".

The term *"feelings"* (+0.0197) is positioned as the central hub node, forming the core component of the negative sentiment cluster. This

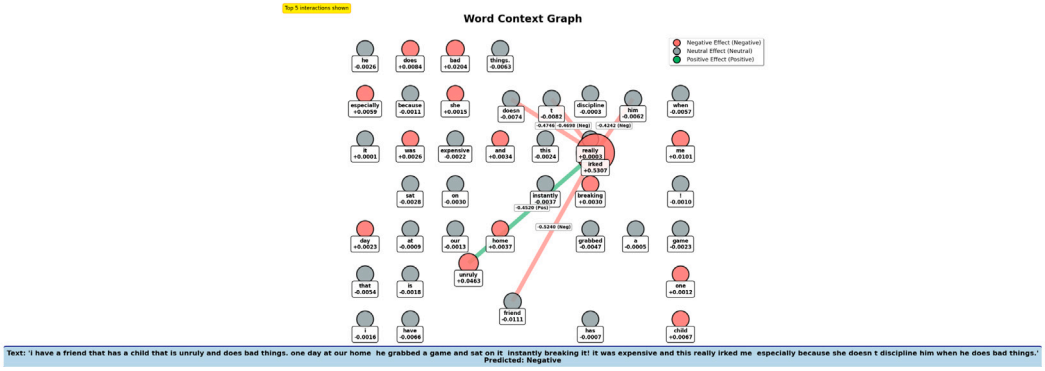



**Fig. 10.** Graph-based visualization with the WordContextGraphExplainer framework for Dataset5.

central positioning reflects the dominant role of emotional discourse within the narrative and highlights the lexical anchor around which semantic interactions are organized.

The graph predominantly features nodes classified as negative, such as *"worthless"* (+0.0068), *"nothing"* (+0.0097), and *"barely"* (+0.0072). These contribute to the accurate identification of depressive language patterns and reinforce the system's capacity to localize affectively significant tokens. Edge weights span a broad spectrum from +0.8360 to −0.6061, indicating considerable variance in the strength of inter-word interactions. Notably, the strongest negative correlations are concentrated around the *"feelings"* hub, supporting its centrality in semantic influence. The nodes *"shouldn"* (+0.0171) and *"be"* (+0.0132) are negatively classified, reflecting the system's ability to detect linguistic indicators of suicidal ideation. This demonstrates the model's sensitivity to subtle syntactic constructions associated with psychological distress. The node *"sleep"* (+0.0125) is identified within the negative sentiment category, indicating the model's capacity to recognize sleep disruption — an important marker in clinical mental health assessments. The term *"think"* (+0.0044) reflects ruminative thought patterns and is correctly positioned within the semantic network. This demonstrates the system's effectiveness in modeling internal cognitive processes associated with depressive episodes. The model's overall prediction of **Negative** sentiment aligns with clinical assessment criteria, suggesting that the system achieves a promising level of accuracy for mental health screening applications. This classification is supported by the density of negative sentiment nodes and their semantically coherent interactions.

This analysis demonstrates that the *WordContextGraphExplainer* framework provides a robust interpretability mechanism for psychologically sensitive content. By quantifying both individual lexical contributions and inter-word semantic interactions, the system delivers a fine-grained visualization of emotional discourse, making it particularly valuable in clinical decision support systems.

The fidelity metric [51] implemented in this framework quantifies the correspondence between explanation-based feature importance rankings and observable model behavior changes through a perturbation-based assessment methodology.

Let $M$ represent the trained model, $x$ denote the original input text, and $E(x)$ represent the explanation method that produces a set of important features $F = \{f_1, f_2, \ldots, f_k\}$ with associated importance scores.

The fidelity score for a single instance is defined as:

$$\text{Fidelity}(x, E) = |M(x) - M(x')| \tag{1}$$

where $x'$ represents the perturbed text obtained by removing the top-$k$ most important features identified by the explanation method $E$.

The fidelity [52] assessment follows this systematic procedure. First, we compute the original model prediction $p_0 = M(x)$ to establish a baseline reference point. Next, we extract the most important features $F = E(x, k)$ using the specified explanation method, where $k$ determines the number of top-ranked features to consider. Subsequently, we create a modified input $x' = \text{Remove}(x, F)$ by removing the identified important features from the original text. We then compute a new prediction $p' = M(x')$ using this perturbed input to observe how the model's behavior changes. Finally, we calculate the fidelity score as fidelity $= |p_0 - p'|$, which quantifies the absolute difference between the original and perturbed predictions.

The underlying hypothesis assumes that if an explanation method accurately identifies decision-critical features, their removal should produce substantial changes in model predictions. Mathematically, this can be expressed as:

$$\text{High Fidelity} \Leftrightarrow \arg\max(M(x)) \neq \arg\max(M(x')) \tag{2}$$

The absolute difference metric captures both direction-preserving and direction-changing prediction modifications, providing a comprehensive assessment of explanation accuracy.

For comprehensive evaluation, individual fidelity scores are aggregated using the arithmetic mean:

$$\text{Mean Fidelity} = \frac{1}{n}\sum_{i=1}^{n} |M(x_i) - M(x_i')| \tag{3}$$

where $n$ represents the total number of test instances.

In the broader context of XAI for natural language processing, *WordContextGraphExplainer* offers methodological advantages over traditional frameworks such as LIME. Unlike LIME, which assumes feature independence and linearity, *WordContextGraphExplainer* employs a graph-theoretic structure capable of capturing non-linear relationships and contextual dependencies features essential for modeling complex, multi-sentiment narratives. These findings underscore the superiority of graph-based interpretability in high-stakes domains and suggest promising future directions for next-generation explainable NLP systems (see Table 23).

**Table 23**
Interpretability Fidelity Score Comparison Across Datasets.

| Dataset | LIME | WordContextGraphExplainer (%) | Improvement (%) |
|---|---|---|---|
| Dataset 1 | 0.8100 | 0.8900 | +9.88 |
| Dataset 2 | 0.8000 | 0.8600 | +7.50 |
| Dataset 3 | 0.6540 | 0.7380 | +12.84 |
| Dataset 4 | 0.6920 | 0.7120 | +2.89 |
| Dataset 5 | 0.6800 | 0.8200 | +20.59 |

## 5. Conclusion

This study presents a comprehensive framework for sentiment classification in dialogue-based scenarios through the development of a

novel HybridBERT-LSTM architecture coupled with an innovative interpretability methodology. The proposed hybrid model demonstrates superior performance on both benchmark datasets, including the widely-adopted IMDb corpus, and real-world dialogue datasets, consistently outperforming standalone architectures such as traditional LSTM, BERT, CNN, and SVM implementations. The empirical results validate the model's enhanced capacity to capture both the semantic richness of individual utterances and the sequential dependencies inherent in multi-turn conversational contexts.

The architectural innovation of HybridBERT-LSTM leverages pre-trained BERT encodings for deep contextualized embeddings, subsequently processed through bidirectional LSTM layers to model temporal dependencies and discourse-level structures. The integration of dual pooling mechanisms (average and maximum) followed by dense classification layers enables the model to synthesize learned representations effectively, making it particularly suitable for dialogue sentiment analysis where contextual flow and sequential relationships are paramount.

A significant contribution of this research lies in the development of explainable context-aware sentiment reasoning capabilities. Beyond the scope of traditional local explanation techniques, a novel graph-theoretic interpretability framework, WordContextGraphExplainer, has been proposed to address the fundamental limitations inherent in existing methodologies. Unlike LIME, which operates under linear additivity assumptions and treats tokens as independent entities, WordContextGraphExplainer employs sophisticated perturbation analysis to model non-linear semantic interactions between word pairs. This methodology constructs semantic interaction graphs where nodes represent individual word contributions and edges encode inter-word dependencies, providing intuitive visualization of complex linguistic relationships through NetworkX-based representations. The comparative analysis reveals that while LIME provides granular word-level attributions, it operates independently of sequential context and fails to capture the synergistic effects crucial for accurate sentiment interpretation in conversational settings. In contrast, WordContextGraphExplainer's graph-based approach explicitly models contextual interdependencies, semantic propagation patterns, and negation scope effects that are essential for understanding transformer decision-making processes. This advancement enables practitioners to trace how sentiment emerges through word interactions and temporal flow across dialogue turns, providing unprecedented insights into model reasoning mechanisms. The integration of WordContextGraphExplainer with HybridBERT-LSTM establishes a new paradigm for interpretable dialogue sentiment analysis, where prediction accuracy and explainability are synergistically enhanced. This framework demonstrates particular efficacy in clinical applications and mental health assessment scenarios, where understanding the rationale behind sentiment predictions is as critical as the predictions themselves. Future research directions include extending the graph-based interpretability framework to multilingual contexts and exploring its applications in other NLP tasks requiring fine-grained semantic understanding. Future work should focus on developing simplified visualization layers and adaptive user interfaces that can present graph-based explanations at varying levels of complexity, enabling domain experts to access meaningful interpretability insights without requiring deep technical expertise in graph theory or network analysis. Future research should incorporate systematic human evaluation studies to assess the explanatory quality and clinical applicability of WordContextGraphExplainer outputs among domain practitioners.

## CRediT authorship contribution statement

**Ercan Atagün:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Conceptualization. **Günay Temür:** Validation, Methodology. **Serdar Biroğul:** Supervision, Project administration, Conceptualization.

## Declaration of competing 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.

## Data availability

The authors do not have permission to share data.

## References


[1] L. Song, et al., CASA: Conversational aspect sentiment analysis for dialogue understanding, J. Artificial Intelligence Res. 73 (2022) 511–533.
[2] M. Firdaus, et al., MEISD: A multimodal multi-label emotion, intensity and sentiment dialogue dataset, in: COLING, 2020, pp. 4441–4453.
[3] I. Carvalho, et al., The importance of context for sentiment analysis in dialogues, IEEE Access 11 (2023) 86088–86103.
[4] J. Wang, et al., Sentiment classification in customer service dialogue with topic-aware multi-task learning, AAAI 34 (05) (2020) 9177–9184.
[5] D. Bertero, et al., Real-time speech emotion and sentiment recognition, EMNLP 104 (2016) 2–1047.
[6] C. Bothe, et al., Dialogue-based neural learning to estimate sentiment, in: ICANN, 2017, pp. 477–485.
[7] M. Firdaus, et al., EmoSen: Generating sentiment and emotion controlled responses, IEEE Trans. Affect. Comput. 13 (3) (2020) 1555–1566.
[8] A. Mallol-Ragolta, B. Schuller, Coupling sentiment and arousal analysis, IEEE Access 12 (2024) 20654–20662.
[9] Z. Akbar, M.U. Ghani, U. Aziz, Boosting viewer experience with emotion-driven video analysis: A BERT-based framework for social media content, J. Artif. Intell. Behav. (2025).
[10] J. Zhao, W. Gao, A semantic-enhanced heterogeneous dialogue graph network, IEEE ICETCI 131 (2024) 5–1322.
[11] M. Yang, et al., GME-dialogue-NET, Acad. J. Comput. Inf. Sci. 4 (8) (2021) 10–18.
[12] M. Parmar, A. Tiwari, Emotion and sentiment analysis in dialogue: A multimodal strategy employing the BERT model, in: 2024 Parul International Conference on Engineering and Technology, PICET, 2024, pp. 1–7.
[13] Mustapha Z., Aspect-based emotion analysis for dialogue understanding, 2024.
[14] W. Li, W. Shao, S. Ji, E. Cambria, BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis, Neurocomputing 467 (2022) 73–82.
[15] S. Poria, D. Hazarika, N. Majumder, R. Mihalcea, Beneath the tip of the iceberg: Current challenges and new directions in sentiment analysis research, IEEE Trans. Affect. Comput. 14 (1) (2020) 108–132.
[16] L. Zhu, R. Mao, E. Cambria, B.J. Jansen, Neurosymbolic AI for personalized sentiment analysis, in: International Conference on Human-Computer Interaction, 269–290, Springer Nature Switzerland, Cham, 2024.
[17] M. Luo, H. Fei, B. Li, S. Wu, Q. Liu, S. Poria, et al., Panosent: A panoptic sextuple extraction benchmark for multimodal conversational aspect-based sentiment analysis, in: Proceedings of the 32nd ACM International Conference on Multimedia, 2024, pp. 7667–7676.
[18] Y. Zhang, Q. Li, D. Song, P. Zhang, P. Wang, Quantum-inspired interactive networks for conversational sentiment analysis, 2019.
[19] L. Yang, Q. Yang, J. Zeng, T. Peng, Z. Yang, H. Lin, Dialogue sentiment analysis based on dialogue structure pre-training, Multimedia Syst. 31 (2) (2025) 1–13.
[20] K. Horesh, A. Kumar, A. Anand, A. Sabu, T. Jain, Sentiment Analysis on Amazon Electronics Product Reviews using Machine Learning Techniques, IEEE, 2023, http://dx.doi.org/10.1109/gcat59970.2023.10353467.
[21] A. Matsui, E. Ferrara, Word embedding for social sciences: An interdisciplinary survey, PeerJ Comput. Sci. 10 (2024) e2562.
[22] S. Anitha, P. Gnanasekaran, Advanced sentiment classification using RoBERTa and aspect-based analysis on large-scale e-commerce datasets, Nanotechnol. Perceptions 20 (S16) (2024) 336–348.
[23] P. Borah, D. Gupta, B.B. Hazarika, ConCave-convex procedure for support vector machines with Huber loss for text classification, Comput. Electr. Eng. 122 (2025) 109925.
[24] Z. Hua, Y. Tong, Y. Zheng, Y. Li, Y. Zhang, PPGloVe: privacy-preserving GloVe for training word vectors in the dark, IEEE Trans. Inf. Forensics Secur. 19 (2024) 3644–3658.
[25] A. Rasool, S. Aslam, N. Hussain, S. Imtiaz, W. Riaz, nbert: Harnessing NLP for emotion recognition in psychotherapy to transform mental health care, Information 16 (4) (2025) 301.
[26] E. Mitera-Kiełbasa, K. Zima, Automated classification of exchange information requirements for construction projects using Word2Vec and SVM, Infrastructures 9 (11) (2024) 194.
[27] Z. Yang, F. Emmert-Streib, Optimal performance of Binary Relevance CNN in targeted multi-label text classification, Knowl.-Based Syst. 284 (2024) 111286.

[28] J. Peng, S. Huo, Application of an improved convolutional neural network algorithm in text classification, J. Web Eng. 23 (3) (2024) 315–339.
[29] K. Nithya, M. Krishnamoorthi, S.V. Easwaramoorthy, C.R. Dhivyaa, S. Yoo, J. Cho, Hybrid approach of deep feature extraction using BERT–OPCNN & FIAC with customized Bi-LSTM for rumor text classification, Alex. Eng. J. 90 (2024) 65–75.
[30] S. Jamshidi, M. Mohammadi, S. Bagheri, H.E. Najafabadi, A. Rezvanian, M. Gheisari, et al., Effective text classification using BERT, MTM LSTM, and DT, Data Knowl. Eng. 151 (2024) 102306.
[31] O. Galal, A.H. Abdel-Gawad, M. Farouk, Federated freeze BERT for text classification, J. Big Data 11 (1) (2024) 28.
[32] C. Eang, S. Lee, Improving the accuracy and effectiveness of text classification based on the integration of the bert model and a recurrent neural network (RNN_Bert_Based), Appl. Sci. 14 (18) (2024) 8388.
[33] M. Ahmed, M.S. Hossain, R.U. Islam, K. Andersson, Explainable text classification model for COVID-19 fake news detection, J. Internet Serv. Inf. Secur. 12 (2) (2022) 51–69.
[34] K. Zahoor, N.Z. Bawany, T. Qamar, Evaluating text classification with explainable artificial intelligence, Int. J. Artif. Intell. ISSN 225 (2024) 2–8938.
[35] D. Kalla, N. Smith, F. Samaah, Deep learning-based sentiment analysis: Enhancing IMDb review classification with LSTM models, 2025, Available at SSRN 5103558.
[36] R. Beniwal, A.K. Dinkar, A. Kumar, A. Panchal, A hybrid deep learning model for sentiment analysis of IMDB movies reviews, in: 2024 Asia Pacific Conference on Innovation in Technology, APCIT, IEEE, 2024, pp. 1–7.
[37] N. Tabassum, T. Alyas, M. Hamid, M. Saleem, S. Malik, Z. Ali, U. Farooq, Semantic analysis of Urdu English tweets empowered by machine learning, Intell. Autom. Soft Comput. 30 (1) (2021) 175–186.
[38] A. Pandey, R. Yadav, A. Pathak, N. Shivani, B. Garg, A. Pandey, Sentiment analysis of IMDB movie reviews, in: 2024 First International Conference on Software, Systems and Information Technology, SSITCON, IEEE, 2024, pp. 1–6.
[39] R. Amin, R. Gantassi, N. Ahmed, A.H. Alshehri, F.S. Alsubaei, J. Frnda, A hybrid approach for adversarial attack detection based on sentiment analysis model using machine learning, Eng. Sci. Technol. an Int. J. 58 (2024) 101829.
[40] A. Bajaj, D.K. Vishwakarma, HOMOCHAR: A novel adversarial attack framework for exposing the vulnerability of text-based neural sentiment classifiers, Eng. Appl. Artif. Intell. 126 (2023) 106815, http://dx.doi.org/10.1016/j.engappai.2023.106815.
[41] A. Bajaj, D.K. Vishwakarma, Evading text-based emotion detection mechanism via adversarial attacks, Neurocomputing 558 (2023).
[42] G.A. de Oliveira, R.T. de Sousa, R. de O. Albuquerque, L.J.G. Villalba, Adversarial attacks on a lexical sentiment analysis classifier, Comput. Commun. 174 (2021) 154–171, http://dx.doi.org/10.1016/j.comcom.2021.04.026.
[43] M. Hussain, M. Naseer, Comparative analysis of logistic regression, LSTM, and Bi-LSTM models for sentiment analysis on IMDB movie reviews, J. Artif. Intell. Comput. 2 (1) (2024) 1–8.
[44] C.D. Kulathilake, J. Udupihille, S.P. Abeysundara, A. Senoo, Deep learning-driven multi-class classification of brain strokes using computed tomography: A step towards enhanced diagnostic precision, Eur. J. Radiol. 187 (2025) 112109.
[45] Amod, Mental health counseling conversations dataset, 2024, Retrieved from https://huggingface.co/datasets/Amod/mental_health_counseling_conversations/tree/main.
[46] B. Yao, P. Tiwari, Q. Li, Self-supervised pre-trained neural network for quantum natural language processing, Neural Netw. 184 (2025) 107004, Elsevier.
[47] SohamGhadge, Casual conversation dataset, 2024, Retrieved from https://huggingface.co/datasets/SohamGhadge/casual-conversation/tree/main.
[48] Mahfoos, Patient-doctor conversation dataset, 2024, Retrieved from https://huggingface.co/datasets/mahfoos/Patient-Doctor-Conversation/tree/main.
[49] Alimistro123, English chat sentiment dataset, 2024, Retrieved from https://www.kaggle.com/code/alimistro123/english-chat-sentiment-dataset-found.
[50] Adapting, Empathetic dialogues v2 dataset, 2024, Retrieved from https://huggingface.co/datasets/Adapting/empathetic_dialogues_v2.
[51] Y. Singh, Q.A. Hathaway, V. Keishing, S. Salehi, Y. Wei, N. Horvat, D.V. Vera-Garcia, A. Choudhary, A.Mula. Kh, E. Quaia, et al., Beyond post hoc explanations: A comprehensive framework for accountable AI in medical imaging through transparency, Interpret. Explain. Bioeng. 12 (8) (2025) 879.
[52] M. Bayesh, S. Jahan, Embedding security awareness in IoT systems: A framework for providing change impact insights, Appl. Sci. 15 (14) (2025) 7871.