International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Adaptive Fusion-Driven Multimodal Sentiment Analysis Using U-Net++, Capsule Network, and Transformer-Enhanced Extreme Learning

[1]J. Ramachandraiah, [2]M. Sirish Kumar

[1]Research Scholar, School of Computing, Department of CSE, Mohan Babu University, Tirupati, India.
[2]Associate Professor, School of Computing, Department of CSE, Mohan Babu University, Tirupati, India.
[1]ramachandraiahjalli@gmail.com, ORCID ID: 0009-0003-6607-7967
[2]drmsk2102@gmail.com, ORCID ID: 0009-0004-6381-8173

**Corresponding Author: M. Sirish Kumar*

Abstract: Multimodal social media data contain textual, visual, and emoji information that can provide complementary cues for sentiment analysis. However, combining these modalities remains challenging because they differ in representation and may contain redundant or noisy information. This study presents an Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework that combines U-Net++ with self-attention, a Capsule Network encoder-decoder, a Cross-Modal Transformer, and a Transformer-Enhanced Extreme Learning Machine (T-ELM). U-Net++ with self-attention extracts contextual features, while the Capsule Network captures hierarchical relationships among the extracted representations. The Cross-Modal Transformer then models interactions between the text, image, and emoji modalities before classification by T-ELM. The trimodal experiments use 9,200 samples selected from the Social Media MultiSent Dataset (SM-MSD), with positive, neutral, and negative sentiment polarity as the target classes. Using a 70% training, 15% validation, and 15% test split, the proposed framework achieves competitive sentiment classification performance across the evaluated datasets. The results show that the proposed fusion framework effectively combines complementary multimodal information while keeping the final classification stage lightweight.

Keywords: Multimodal sentiment analysis, U-Net++, capsule network, Transformer-Enhanced Extreme Learning Machine, cross-modal fusion.

1 INTRODUCTION

Sentiment analysis (SA), also known as opinion mining, is an important field of natural language processing (NLP) that aims to understand the polarity and emotional characteristics expressed in textual, visual, and symbolic data [1]–[3]. Transformer-based models such as BERT, RoBERTa, and GPT have improved the ability to capture contextual dependencies and semantic information in text. However, scalability and effective cross-modal fusion remain challenging in dynamic social media environments. Deep learning-based multimodal sentiment analysis approaches have explored image-text fusion mechanisms to capture interactions between heterogeneous modalities and improve sentiment representation [4]. Hybrid architectures such as CNN–LSTM and grid LSTM–CNN have shown good performance on specific datasets. However, their performance may suffer when applied to heterogeneous or unseen data sources. Feature redundancy, class imbalance, and domain differences further complicate multimodal sentiment analysis.

Social media data often contain informal language, emojis, sarcasm, and slang, which can make sentiment interpretation difficult. Transformer-based multimodal approaches have been investigated to capture contextual information and cross-modal interactions for sentiment analysis, particularly when integrating heterogeneous social media content [5]. Big-data sentiment analysis frameworks provide scalable processing through distributed architectures, but computational cost and model interpretability remain important considerations. Recent emotion detection and depression analysis models have also shown promising results under specific conditions. However, many of them focus mainly on textual or task-specific information rather than jointly modeling multiple modalities [6]. These limitations motivate the development of multimodal approaches that can combine semantic and emotional information across different types of social media content [7].

Multimodal Sentiment Analysis (MSA) has gained increasing attention in recent years because it can combine information from multiple data types for sentiment prediction [8]. Thandaga Jwalanaiah et al. [9] proposed a novel deep learning architecture for MSA that includes analytics units for text, image, and discretization control. Chandrasekaran et al. [10] studied visual sentiment prediction using transfer learning approaches such as VGG-19, ResNet50V2, and DenseNet-121, achieving accuracies of 0.73, 0.75, and 0.89. Deep learning developments have considerably revolutionized traditional sentiment analysis of textual data. Alsayat et al. [11] have proposed a novel deep learning model that combines word embedding and LSTM ensemble modeling. Behera et al. [12] also presented a Co-LSTM architecture that is highly scalable and domain-independent for large-scale textual data.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Another important development is a hybrid feature extraction technique that incorporates review- and aspect-specific features categorized by an LSTM model, proposed by Kaur et al. [13]. Priyadarshini et al. [14] further expanded this paradigm by developing a novel LSTM-CNN-grid search-based neural network model that outperformed baseline CNN, KNN, and LSTM models, achieving accuracies over 96%. BERT-based sentiment analysis has also been applied to cryptocurrency markets by combining contextual sentiment features with market microstructure information [15]. Pasham et al. [16] have looked into the scalability of graph-based techniques in social network analysis for use cases such as sentiment analysis and anomaly detection. Their work illustrates real-time analysis using distributed computing and graph machine learning, with a focus on constraints and ethics in large-scale social data mining.

Sentiment analysis has now started being used in mental health and affective computing. Tejaswini et al. [17] present a recent text-based approach to depression detection, proposing a novel "FastText CNN-LSTM (FCL)" hybrid model to address depression detection through text. The model uses FastText embeddings along with a CNN for extracting global patterns and an LSTM for capturing local dependencies. This combination improves the representation of textual patterns associated with depression indicators. Helmy et al. [18] also made a significant contribution to depression detection by proposing five machine learning models for the same task on English and Arabic tweets. The most effective model achieved an F1-score of 96.6% for Arabic texts using newly built corpora of manually and automatically annotated data. They also developed a web application for real-time depression detection.

Several studies have also focused on domain-specific applications of sentiment analysis. For example, Alfaisal et al. [19] implemented CNN models for performing sentiment analysis on tweets related to the Metaverse. It was designed for the short, informal language patterns common in social media and achieved 96% accuracy, with precision and recall above 0.95. Kuruva et al. [20] proposed an innovative sentiment analysis model that uses text and emojis with CNN, MLP, and Hybrid LSTM-RNN classifiers. Although deep learning has advanced multimodal sentiment analysis, several gaps remain.

1. Several existing multimodal systems perform modality-specific feature extraction before combining the resulting representations [9], [10].
2. Transformer-based approaches such as BERT–CBRNN primarily focus on contextual text representation, while handling emoji-based sentiment, sarcasm, and noisy social-media expressions remains challenging.
3. Distributed big-data sentiment analysis frameworks provide scalability, but computational cost and interpretability remain important considerations [16].
4. Several emotion-recognition approaches focus on specific affective dimensions or particular application settings, leaving scope for models that jointly consider complementary multimodal cues [17], [18].

These gaps motivate an adaptive, fusion-driven architecture that integrates complementary multimodal signals through learned representations. Both unimodal and multimodal sentiment analysis models face challenges in learning relationships between different modalities, handling noisy social-media content, and maintaining consistent representations across datasets. These limitations motivate the development of a fusion-driven multimodal framework that combines textual, visual, and emoji-based sentiment cues to improve contextual representation and sentiment prediction. The primary objectives of the proposed research work are as follows:

1. To develop a cross-modal deep learning approach for integrating textual, visual, and emoji-based sentiment cues using attention-enabled feature extraction and fusion mechanisms.
2. To develop a multimodal learning framework that can improve sentiment classification performance while reducing the computational burden of the final classification stage.

The novelty of this work is the development of an Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework that integrates U-Net++, Capsule Networks, a Cross-Modal Transformer, and a Transformer-Enhanced Extreme Learning Machine (T-ELM) to fuse multimodal representations while keeping the final classification stage efficient. The framework uses attention-based feature refinement and multimodal representation fusion to combine complementary information from the text, image, and emoji modalities. The T-ELM provides a lightweight final classification stage while preserving the multimodal representation learned by the preceding modules. The contributions of this study are outlined below:

1. An adaptive multimodal sentiment analysis framework is developed to jointly integrate textual, visual, and emoji information through a unified fusion architecture.
2. A hierarchical multimodal representation learning strategy is introduced by combining self-attention-enhanced U-Net++, an encoder–decoder Capsule Network, and a Cross-Modal Transformer to capture complementary and contextual information across modalities.
3. A Transformer-Enhanced Extreme Learning Machine (T-ELM) classification stage is incorporated to provide lightweight final sentiment classification using the fused multimodal representation.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Table 1. Summary of sentiment analysis.

Ref.	Method / Focus Area	Algorithm / Model Used	Methodology / Key Contributions	Outcomes / Results
[9]	Multimodal Sentiment Analysis (Text + Image + Embedded Text)	SRNN-MAFM + PDC-SAM + Boolean Decision Model	Combines text and image analytics using VCA-MSERs; stacked RNN with multilevel attention for text and CNN with parallel-dilated convolution for images; Boolean logic for fusion.	Accuracy: 97.8% (text), 97.7% (visual), 90% (multimodal) on STS-Gold, Flickr8k, B-T4SA.
[11]	Contextual Text Sentiment Analysis	LSTM + Word Embedding + Ensemble Learning	Develops robust contextual embedding; ensemble of baseline classifiers with LSTM to manage unseen words and domain variation (e.g., COVID-19).	High accuracy on Twitter, Amazon, Yelp datasets; improved contextual generalization.
[10]	Visual Sentiment Analysis	VGG-19, ResNet50V2, DenseNet-121	Uses transfer learning with fine-tuning (layer freezing/unfreezing) and regularization on the Crowdflower Twitter image dataset.	Accuracy: 0.73 (VGG-19), 0.75 (ResNet50V2), 0.89 (DenseNet-121); improved by 5–10%.
[12]	Text Sentiment Analysis (Hybrid)	CNN + LSTM (Co-LSTM)	Domain-independent model for large-scale social data.	Outperformed machine learning baselines on four diverse review datasets.
[13]	Aspect-based Sentiment Analysis	LSTM + Hybrid Feature Extraction	Uses NLP preprocessing and constructs hybrid feature vectors combining review-related and aspect-based features for sentiment classification.	Avg. Precision: 94.46%, Recall: 91.63%, F1-score: 92.81%.
[14]	Optimized Deep Sentiment Classifier	LSTM–CNN–Grid Search-Based DNN	Combines CNN and LSTM with grid search hyperparameter optimization; compared with CNN, KNN, and NN baselines.	Accuracy >96%; superior performance across precision, sensitivity, specificity, and F1-score.
[17]	Depression Detection from Text	FastText + CNN + LSTM (FCL Model)	Uses FastText for semantic embeddings, CNN for global features, LSTM for sequential dependencies; applied to depression-related text analysis.	Improved representation and detection accuracy of depression indicators.
[16]	Graph-Based Real-Time Social Network Analysis	Graph Algorithms + ML on Graphs	Focuses on scalable graph algorithms for sentiment, anomaly, and community detection with parallel and distributed learning.	Showed scalability and real-time performance; addressed ethical challenges.
[19]	Domain-Specific SA (Metaverse Tweets)	CNN	Applies CNN to classify Metaverse-related tweets; handles informal text via tokenization and normalization.	Accuracy: 96%; Precision and Recall >0.95.
[18]	Depression Detection (Arabic & English Tweets)	ML Classifiers + Language-Specific Corpus	Developed Arabic_Dep_10k and English corpora (with/without negation); trained models for binary and multi-class classification.	F1-score: 96.6% (Arabic), 92% (English binary), 88% (English multi-class).
[15]	Crypto Market Sentiment and Prediction	BERT + Deep Learning + Market Indicators	Integrates BERT-based sentiment with market microstructure data; includes attention and temporal modules.	Prediction accuracy: 91.2%; Sharpe Ratio: 2.34; Precision: 92.3%; Real-time latency < 100 ms.
[20]	Emoji-Enhanced Sentiment Analysis	CNN + MLP + LSTM–RNN + ECSSO	Incorporates text and emoji features using hybrid CNN–MLP–LSTM; weight optimization via Electric Fish Customized Shark Smell Optimization.	Specificity improved up to 42.75% over baseline models for a 70% data ratio.

2 METHODOLOGY

The proposed Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework (AFDMSAF) integrates U-Net++ with self-attention, an Encoder–Decoder Capsule Network (CNED), a Cross-Modal Transformer, and a Transformer-Enhanced Extreme Learning Machine (T-ELM). The framework processes textual, visual, and emoji inputs and fuses their representations for sentiment classification. The overall processing sequence comprises hierarchical feature extraction, capsule-based semantic representation, cross-modal transformer refinement, and final classification with T-ELM. The data flow is defined as follows.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

First, U-Net++ with self-attention extracts hierarchical representations from the modality-specific inputs. The Capsule Network then processes these representations to capture higher-level relationships among the extracted features. The Transformer module subsequently processes the resulting capsule representations to refine their contextual representations before multimodal fusion. Finally, the resulting representation is passed to T-ELM for three-class sentiment classification.

Stage 1: Multimodal Feature Extraction using U-Net++ with Self-Attention

The first stage converts the text, image, and emoji inputs into feature representations suitable for multimodal learning. Text and emoji inputs are represented as sequences of embeddings, while images are converted into visual feature representations. U-Net++ with self-attention is then used to refine the modality-specific features and capture information at different levels of representation. For sequential representations, the U-Net++ feature extraction process is adapted with one-dimensional convolutions, and self-attention models relationships between the extracted features.

Stage 2: Semantic Representation using Encoder–Decoder Capsule Network

These extracted features are passed to an Encoder–Decoder Capsule Network, which captures hierarchical relationships within the multimodal feature representations. Capsules provide higher-dimensional representations of feature vectors and can encode information about the strength and relationships among sentiment cues. During encoding, a dynamic routing mechanism transforms the feature representations into capsules, while the decoder reconstructs the corresponding feature representation. Unlike conventional convolutional representations, the Capsule Network preserves hierarchical relationships among extracted features and provides a richer representation of sentiment-related cues.

Stage 3: Transformer-Based Contextual Refinement and T-ELM Classification

In the final phase, the Transformer-refined multimodal representation is passed to the Transformer-Enhanced Extreme Learning Machine (T-ELM) for sentiment classification. The Transformer module first refines the capsule-derived representations using multi-head self-attention to capture contextual relationships among the feature vectors. The enhanced representation is then passed to the T-ELM classifier, which uses a 512-neuron hidden layer with sigmoid activation followed by a three-neuron output layer for sentiment classification.

This design provides a lightweight final classification stage while retaining the multimodal representation learned by the preceding modules. Fig. 1 illustrates the overall architecture of the proposed framework. The processing pipeline begins with the multimodal input tuple. $X = \{T, V, E\}$, where $T$is the text sequence V, which represents the RGB image, and E corresponds to the emoji sequence. The textual input is tokenized into a maximum sequence length of 128 tokens, producing an input tensor of size (32 × 128) for a batch size of 32. After transformer-based embedding, the textual representation becomes (32 × 128 × 256), where 256 is the embedding dimension. The image input is resized to 224 × 224 × 3, resulting in an input tensor of (32 × 224 × 224 × 3). The encoder progressively extracts hierarchical features through four stages, producing feature maps of (32 × 112 × 112 × 64), (32 × 56 × 56 × 128), (32 × 28 × 28 × 256), and (32 × 14 × 14 × 512), respectively. The decoder reconstructs the multiscale representation using dense skip connections, and global average pooling compresses the final feature map to obtain an image embedding of size (32 × 256). Emoji sequences are first mapped into sentiment-aware embeddings of dimension 128, which is resulting in an embedding tensor of (32 × 20 × 128) for the maximum emoji sequence length of 20.

A linear projection layer transforms these embeddings into 256-dimensional vectors, producing an output tensor of (32 × 20 × 256). All three modalities are therefore projected into a common latent feature space of identical dimensionality before fusion. The resulting multimodal representations are subsequently processed by the Transformer module using multi-head attention to refine contextual relationships among the modality-specific features. The refined multimodal representations are aggregated to obtain a compact 256-dimensional representation for the final T-ELM classifier. The three output logits correspond to the positive, neutral, and negative sentiment classes, and the predicted class is determined from the highest output probability. The proposed framework's operational workflow is summarized in the following steps.

2.1. Multimodal Data Preprocessing and Feature Extraction using U-Net++ with Self-Attention

The preprocessing and feature-extraction stages form the first part of the proposed multimodal sentiment analysis system. The objective is to convert heterogeneous data sources, including text, images, and emojis, into a common, high-quality feature representation for semantic modeling. The U-Net++ model, together with a self-attention mechanism, enables effective context-based feature extraction, noise reduction, and preservation of semantic information. Compared with CNN-based feature extractors, the designed U-Net++ model with self-attention enables hierarchical, dense feature learning through their integration. U-Net++ is not applied directly to raw textual sequences or emoji symbols as a conventional image segmentation network. Instead, embedding layers and positional encoding first transform the textual tokens and emoji embeddings into dense contextual representations. Specifically, the text input is tokenized into a sequence of 128 tokens, and each token is projected into a 256-dimensional embedding, yielding a 128 × 256 feature matrix. Similarly, emoji sequences are converted into 256-dimensional sentiment-aware embeddings, producing a 20 × 256 feature matrix. These matrices are interpreted as one-dimensional feature maps, where the sequence length corresponds to the spatial dimension and the embedding channels correspond to feature channels.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

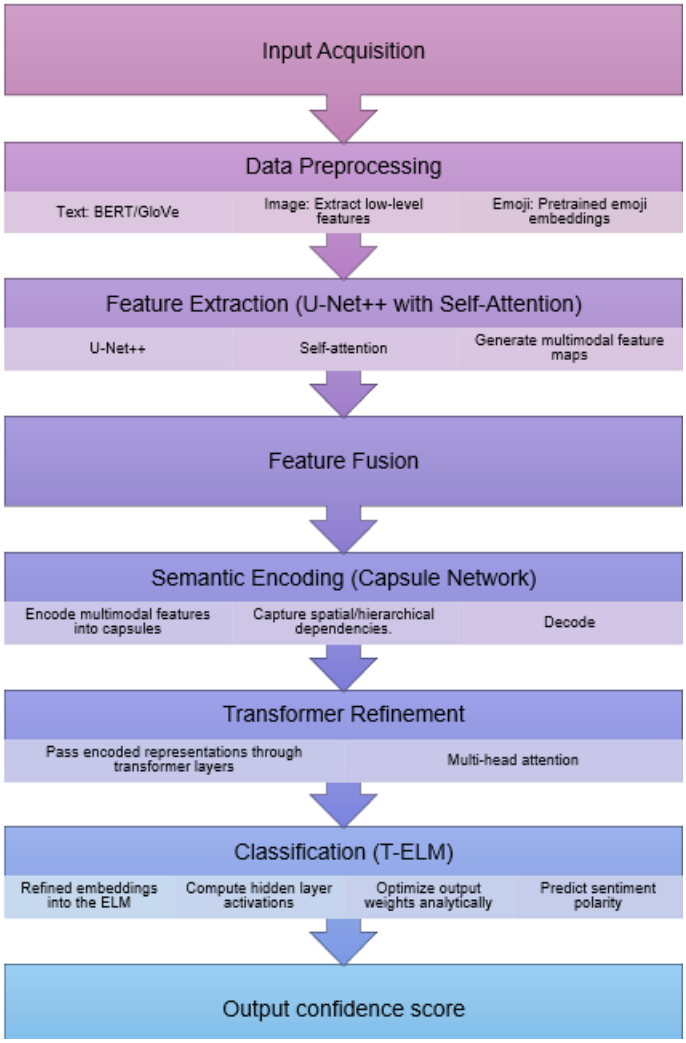

Fig. 1. Proposed adaptive fusion-driven multimodal sentiment analysis framework.

Accordingly, the two-dimensional convolution operations used in the original U-Net++ are replaced with one-dimensional convolutions, allowing the adapted network to extract local semantic patterns from the sequential embeddings. The nested dense skip connections are preserved to facilitate multi-scale feature reuse and improve gradient propagation between the shallow lexical representations and deeper semantic representations. During the encoder stage, successive 1D convolution and pooling operations progressively capture higher-level contextual semantics, while the decoder reconstructs fine-grained contextual information through the dense skip connections that merge encoder and decoder features at multiple resolutions. Therefore, the adapted U-Net++ functions as a hierarchical sequential feature extractor rather than a conventional image segmentation network, allowing the textual and emoji representations to capture contextual and multi-scale semantic information before multimodal fusion. The operational workflow of the proposed framework is summarized in Algorithm 1.

2.1.1. Multimodal Data Preprocessing

The preprocessing stage standardizes, denoises, and semantically aligns the multimodal inputs before they enter the deep learning pipeline. Three input modalities, text, image, and emoji, are subjected to separate yet complementary normalization pipelines, as summarized in Table 2. During preprocessing, textual data is first normalized by removing irrelevant symbols and punctuation, followed by tokenization and embedding using BERT embeddings $\boldsymbol{E}_T \in \mathbb{R}^{n \times d_t}$, where n is the sequence length, and d_t is the embedding dimension. The embedding operation can be expressed as in Eq. (1).

$$\boldsymbol{E}_T = f_{\text{BERT}}(w_1, w_2, \ldots, w_n) \tag{1}$$

where f_{BERT} is the pretrained BERT transformer function mapping words w_i to contextual embeddings.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Algorithm 1. Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework

```
# Pseudocode: Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework
Input: Multimodal dataset D = {Text_T, Image_I, Emoji_E}
Output: Predicted Sentiment Labels S_pred
BEGIN

# Step 1: Data Preprocessing
    for each (T, I, E) in D do
        T_clean ← preprocess_text(T)        # tokenization, stopword removal, emoji normalization
        T_embed ← embed_text(T_clean)       # BERT embeddings
        I_proc  ← preprocess_image(I)       # resizing, normalization
        E_embed ← embed_emoji(E)            # emoji2vec or pretrained embeddings
    end for

# Step 2: Feature Extraction (U-Net++ with Self-Attention)
    for each modality feature in {T_embed, I_proc, E_embed} do
        F_m ← U_NetPP_Attention(feature)    # Dense convolution + skip connections + self-attention
    end for

# Step 3: Multimodal Feature Fusion
    F_fused ← concatenate(F_T, F_I, F_E)
    F_fused ← normalize(F_fused)

# Step 4: Semantic Representation using Capsule Network
    Encoded_Caps ← Capsule_Encoder(F_fused)
    Reconstructed ← Capsule_Decoder(Encoded_Caps)
    Semantic_Embeddings ← merge(Encoded_Caps, Reconstructed)

# Step 5: Transformer Refinement
    Contextual_Emb ← Transformer_Layer(Semantic_Embeddings)

# Step 6: Sentiment Classification (T-ELM)
    H ← sigmoid(W_hidden * Contextual_Emb + b_hidden)
    Logits ← W_output * H + b_output
    S_pred ← argmax(Logits)

# Step 7: Output
    return S_pred

END
```

Table 2. Multimodal data preprocessing steps.

Modality	Preprocessing Steps	Output Representation
Text	Textual data are tokenized and normalized before being converted into contextual embeddings. The resulting sequence is limited to a maximum of 128 tokens and projected to the feature dimension used by the proposed framework.	Contextual token vectors
Image	Resizing (224×224), color normalization, contrast adjustment, and Gaussian smoothing	Normalized image tensor
Emoji	Emoji sequences are converted into embedding vectors and projected into the common feature space used for multimodal processing.	Sentiment vector representation

For the image modality, feature normalization standardizes the input values and reduces the effect of intensity variations. The normalized image tensor $\boldsymbol{I}_N \in \mathbb{R}^{H\times W\times C}$ is computed using Eq. (2).

$$\boldsymbol{I}_N(x,y,c) = \frac{\boldsymbol{I}(x,y,c) - \mu_c}{\sigma_c} \tag{2}$$

where μ_c and σ_c are the channel-wise mean and standard deviation, respectively. Emoji data are converted into embedding vectors, where each emoji e_i is mapped to an embedding representation $\boldsymbol{E}_e$.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

$$\boldsymbol{E}_e = f_{\text{Emoji2Vec}}(e_i) \tag{3}$$

This creates a semantically aligned sentiment representation that complements textual embeddings during multimodal fusion. Before feature extraction, the modality-specific representations are projected into a common 256-dimensional feature space.

2.1.2. U-Net++ Architecture for Hierarchical Feature Extraction

AFDMSAF uses U-Net++ as its feature-extraction component. Its hierarchical structure retains information from different feature-representation levels before passing the data to the Capsule Network. For sequential inputs, the feature extractor uses one-dimensional convolutional operations to process the embedded representations. The extracted features are subsequently refined using self-attention. This adaptation lets the same feature-extraction stage operate on sequential text and emoji representations while maintaining a common feature dimension for subsequent multimodal processing. Let X denote the input feature sequence. The two one-dimensional convolutional operations used for feature extraction are defined in Eq. (4) and (5).

$$F_1 = \sigma(\text{Conv1D}_1(X)) \tag{4}$$

$$F_2 = \sigma(\text{Conv1D}_2(F_1)) \tag{5}$$

where $\sigma(\cdot)$ denotes the nonlinear activation function. The resulting feature representation F_2 is then refined using the self-attention mechanism described in the following subsection.

For the multimodal feature extractor, two one-dimensional convolutional layers process the input embeddings, producing 128 and 256 output channels, respectively. The resulting 256-dimensional feature sequence is then passed through an eight-head self-attention module to refine contextual representations. The same feature-extraction structure is applied to the text, image, and emoji branches before capsule-based representation learning. These nested operations allow the model to extract features that retain both fine-grained and abstract sentiment information, crucial for multimodal alignment.

2.1.3. Self-Attention Mechanism

Self-attention is incorporated into the feature extraction stage to model relationships between elements of the extracted representation. For an input feature matrix X, query, key, and value representations are obtained through learned projections:

$$Q = \boldsymbol{F}_2\boldsymbol{W}_Q, \quad K = \boldsymbol{F}_2\boldsymbol{W}_K, \quad V = \boldsymbol{F}_2\boldsymbol{W}_V \tag{6}$$

The attention weights are calculated using scaled dot-product attention:

$$A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{7}$$

and the refined feature representation is given in Eq. (8).

$$X_{\text{att}} = AV \tag{8}$$

In the implementation, multi-head self-attention with eight heads refines the extracted features before capsule-based representation learning.

2.2. Semantic Representation using Encoder–Decoder Capsule Network

After feature extraction and self-attention, an encoder–decoder Capsule Network processes the resulting multimodal representations. The Capsule Network represents the extracted features as compact vector representations while retaining information about the magnitude and direction of the feature vectors. In the proposed implementation, the capsule encoder uses a linear projection to generate three capsules, each represented by a 16-dimensional vector. A squash function is then applied to constrain the capsule vector magnitude while preserving its direction. The encoded capsules are subsequently passed through a lightweight decoder to reconstruct the input feature representation. The Capsule Network receives the 256-dimensional feature representation generated by the adapted U-Net++ feature extractor, where LL denotes the sequence length of the corresponding modality. The capsule encoder applies a linear projection to the 256-dimensional input representation to generate three capsules, each represented by a 16-dimensional vector. The resulting three capsules are represented in a 16-dimensional capsule space corresponding to the three sentiment classes: positive, neutral, and negative. The squash function constrains the capsule magnitude while preserving its direction, producing a bounded vector representation.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

A lightweight reconstruction network reconstructs the input feature representation from the encoded capsules. The decoder uses fully connected layers with 512 and 256 hidden units followed by a linear reconstruction layer. The encoder transforms the fused multimodal features $\breve{\boldsymbol{F}}$ obtained from U-Net++ into high-dimensional capsule embeddings. Each capsule $\boldsymbol{u}_i$ is a vector representing a specific semantic attribute of the sentiment, such as polarity intensity or emotional orientation. For an input feature map $\breve{\boldsymbol{F}} \in \mathbb{R}^{H\times W\times D}$, the primary capsule output $\boldsymbol{u}_i$ is obtained using convolutional transformations shown in Eq. (9).

$$C = \text{reshape}\,(\text{Linear(X)}, 3, 16) \tag{9}$$

where $\boldsymbol{W}_i$ and $\boldsymbol{b}_i$ denote learnable weights and biases, and $s(\cdot)$ is the squash function defined as shown in Eq. (10).

$$v(\boldsymbol{u}) = \frac{\|\boldsymbol{u}\|^2}{1+\|\boldsymbol{u}\|^2}\frac{\boldsymbol{u}}{\|\boldsymbol{u}\|} \tag{10}$$

The squash function reduces the magnitude of short vectors toward zero while constraining the magnitude of longer vectors below one. Table 3 shows the primary capsule encoding across the three modalities.

Table 3. Encoder capsule feature mapping.

Modality	Input Dimension	Capsule Dimension	Capsule Count	Output Representation
Text	786 × 1	16	3	Semantic token capsules
Image	56×56×256	16	3	Visual feature capsules
Emoji	128 × 1	16	3	Emoji sentiment capsules

The decoder reconstructs the input feature representation from the encoded capsule representation. The reconstruction $\widehat{\boldsymbol{F}}$ is computed via a fully connected decoder shown in Eq. (11).

$$\widehat{\boldsymbol{F}} = g(\boldsymbol{V}\boldsymbol{v} + \boldsymbol{b}_d) \tag{11}$$

where g(·) is a nonlinear activation (ReLU), **V** represents decoder weights, and **v** is the high-level capsule vector from the encoder. The modality-specific representations are retained for subsequent Transformer-based contextual refinement and are fused at the multimodal representation stage

2.3. Cross-Modal Transformer

2.3.1. Transformer Configuration

The proposed framework adopts an encoder-only Transformer architecture, where the objective is to learn contextual relationships among the multimodal features rather than generating sequential outputs. The Transformer encoder consists of 4 stacked encoder layers, where each layer contains a multi-head self-attention mechanism, a position-wise feed-forward network, residual connections, and layer normalization. The multi-head attention module uses 8 attention heads, with the hidden embedding dimension of 256, which is resulting in a head dimension of 32. The scaled dot-product attention is computed as:

$$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}} + M\right)V \tag{12}$$

where $d_k = 32$ is representing the dimension of each attention head$M$ and it is the attention mask. The feed-forward network within each encoder block contains two fully connected layers with an intermediate dimension of 1024, using the GELU activation function.

$$FFN(x) = W_2(GELU(W_1x + b_1)) + b_2 \tag{13}$$

The multimodal sequence length is fixed to 344 tokens, which consists of 128 textual tokens, 196 visual patch tokens, and 20 emoji tokens. Each token is projected into a common 256-dimensional latent embedding space before being processed by the Transformer. A learnable positional encoding strategy is adopted instead of fixed sinusoidal encoding because the multimodal tokens originate from different modalities and require adaptive positional representations. The positional embedding matrix is defined as:

$$P \in R^{L\times d} \tag{14}$$

where $L = 344$ is the multimodal sequence length $d = 256$ and it is the embedding dimension.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

The final Transformer input representation is obtained as:

$$H_0 = X_{multi} + P \tag{15}$$

where X_{multi} is the aligned multimodal embedding matrix. A padding attention mask is applied to ignore incomplete sequences during batching, while no causal mask is used because the task is performing sentiment classification and it requires bidirectional contextual information. The dropout rate is set to 0.1 within the Transformer attention and feed-forward layers, and 0.3 is applied before the final classifier to reduce overfitting. The Transformer output maintains the tensor dimension of 344×256, which is subsequently aggregated to generate a fixed-length multimodal representation for the final T-ELM classifier.

2.3.2. Transformer-Enhanced Module for Contextual Modeling

The Transformer module is incorporated before the ELM layer to enrich the capsule embeddings with temporal and contextual relationships. Let the fused capsule embedding from the encoder-decoder network be $\breve{\boldsymbol{C}} \in \mathbb{R}^{d_c}$. The Transformer applies self-attention to learn the importance of each dimension in the embedding vector shown in Eqs. (16) and (17).

$$Q = \breve{\boldsymbol{C}}W_Q, \quad K = \breve{\boldsymbol{C}}W_K, \quad V = \breve{\boldsymbol{C}}W_V \tag{16}$$

$$\text{Att}(\breve{\mathbf{C}}) = \xi\left(\frac{QK^{\top}}{\sqrt{d_k}}\right) \tag{17}$$

where W_Q, W_K, and W_V are learnable projections, and d_k is the key dimension. The attention output emphasizes sentiment-relevant features, allowing the ELM to operate on contextually enriched embeddings. Table 10 presents the attention weights assigned to the text, image, and emoji modalities for representative samples. For illustration, four representative samples are denoted as S1, S2, S3, and S4 in Tables 4 and 5.

Table 4. Modality attention weights for representative samples.

ID	Text Attention	Image Attention	Emoji Attention	Dominant Modality
S1	0.42	0.51	0.07	Image
S2	0.48	0.45	0.07	Text
S3	0.35	0.60	0.05	Image
S4	0.50	0.40	0.10	Text

2.3.3. Multimodal Feature Fusion and Representation Learning

After modality-specific feature extraction and attention refinement, the resulting representations are passed to the subsequent Capsule Network and Transformer modules for further representation learning before final multimodal fusion. The modality-specific representations are subsequently aggregated and fused to form a fixed-length multimodal representation for the final classification stage. This provides a single representation containing information from all three modalities while retaining their individual feature contributions.

2.4. T-ELM Classification Process

The T-ELM performs the final sentiment classification using the fused multimodal representation generated by the Cross-Modal Transformer. The resulting 256-dimensional representation passes through a hidden layer with 512 neurons and sigmoid activation, followed by a three-neuron output layer corresponding to the positive, neutral, and negative sentiment classes. The classifier receives the fused representation generated by the preceding multimodal modules.

$$\mathbf{H} = \mathrm{g}(\mathbf{X}\mathrm{W}_\mathrm{h} + \mathbf{b}_\mathrm{h}) \tag{18}$$

where W_h and b_h are the trainable hidden-layer weights and biases. The output can be expressed as:

$$\mathrm{o} = \mathbf{H}\mathrm{W}_\mathrm{o} + \mathbf{b}_\mathrm{o} \tag{19}$$

where W_o and $\mathbf{b}_\mathrm{o}$ are the trainable output-layer parameters. Table 11 presents the classification results obtained from individual modalities and the fused multimodal representation for the representative samples.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Table 5. T-ELM classification results for representative samples.

ID	Text Only	Image Only	Emoji Only	Multimodal (T-ELM)	True Label
S1	Neutral	Positive	Neutral	Positive	Positive
S2	Positive	Neutral	Neutral	Positive	Positive
S3	Negative	Positive	Negative	Positive	Positive
S4	Positive	Neutral	Neutral	Neutral	Neutral

2.4.1. Loss Function and Optimization

The proposed classifier is trained using cross-entropy loss for three-class sentiment classification. For a training sample with target class y_i and predicted class probabilities $p_{i,c}$, the classification loss is defined as:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N} log\ p_{i,y_i} \tag{20}$$

where N is the number of training samples and p_{i,y_i} denotes the predicted probability assigned to the correct sentiment class. The classifier parameters are optimized jointly using gradient-based training. The lightweight classifier reduces the complexity of the final classification stage while retaining the fused representation learned by the preceding multimodal modules.

3 Results and Discussions

To evaluate the performance of the proposed framework of U-Net++ + EDCN + T-ELM, five representative techniques have been selected from recent literature, representing unimodal, hybrid, and transformer-based sentiment analysis. The techniques are: (1) SRNN-MAFM with VCA-MSERs [9], multimodal sentiment analysis technique using text and image inputs in combination with attention driven recurrent neural network structure; (2) Hybrid CNN–LSTM (Co-LSTM) [12] model which combines the advantages of convolutional feature extraction with sequential memory networks in case of textual sentiment analysis; (3) BERT–CBRNN [21] – transformer-based network using contextual embeddings with convolutional and Bi-LSTM layers; (4) LSTM–CNN grid search-based model [14], hybrid structure optimized for multi-domain sentiment classification; and (5) Deep Learning Modified Neural Network (DLMNN) [22] – deep learning model used for large scale social media sentiment prediction. The techniques listed above cover the full spectrum of contemporary approaches to sentiment analysis and can serve as a good performance benchmark for the T-ELM improvements.

3.1. Experimental Settings

The proposed AFDMSAF is developed in the following software versions: Python 3.10, PyTorch 2.1.0, CUDA 11.8, cuDNN 8.9, TorchVision 0.16.0, NumPy 2.4.3 All experiments are conducted on a workstation equipped with the Intel Core i7-12700K (12 cores, 3.60 GHz) processor, 32 GB DDR4 RAM, and an NVIDIA GeForce RTX 3060 GPU with the 12 GB GDDR6 memory, running Ubuntu 22.04 LTS (64-bit). To confirm deterministic execution, the random seed is fixed to 42 for Python, NumPy, and PyTorch, while torch.backends.cudnn.deterministic=True and torch.backends.cudnn.benchmark=False are enabled to eliminate nondeterministic GPU operations. The model is trained using the Adam optimizer with the initial learning rate of 0.0001, momentum coefficients of $\beta_1 = 0.9$ and $\beta_2 = 0.999$, weight decay = 1×10^{-5}, and $\varepsilon = 1 \times 10^{-8}$. A Cosine Annealing Learning Rate Scheduler is employed to gradually reduce the learning rate during training according to Eq. (21).

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_0 - \eta_{\min})\left(1 + \cos\left(\frac{\pi t}{T}\right)\right) \tag{21}$$

where $\eta_0 = 0.0001$, $\eta_{\min} = 1 \times 10^{-6}$ t, which is the current epoch $T = 100$, and it represents the total number of training epochs. The training configuration uses a batch size of 32, 100 training epochs, gradient clipping with the maximum norm of 1.0, a dropout rate of 0.30, and mixed-precision (FP16) training to improve computational efficiency. Early stopping is based on the validation F1-score, with a patience of 10 epochs and a minimum improvement threshold (min_delta) of 0.001. During preprocessing, textual inputs are tokenized using the Hugging Face tokenizer, images are processed using TorchVision and OpenCV, and emoji normalization is performed using the Python emoji library before projection into sentiment-aware embeddings. All hyperparameters tend to remain fixed across the six benchmark datasets to ensure a fair comparison.

3.2. Datasets

The experimental evaluation used multimodal datasets to represent diverse sentiment inputs: text, images, and emojis. The six datasets used in the experimental evaluation are STS-Gold, Flickr 8k, B-T4SA, SM-MSD, Amazon Reviews, and Yelp Reviews.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Table 6 summarizes the datasets used in the experimental evaluation, including their modalities, sample sizes, sentiment classes, and sources.

Table 6. Summary of datasets used in the experimental evaluation.

Dataset	Modality	Size	Sentiment Classes	Source
STS-Gold [23]	Text	2034	Positive, Neutral, Negative	Social media posts
Flickr 8k [24]	Image	8,000	Positive, Neutral, Negative	Image captions dataset
B-T4SA [25]	Multimodal (Text + Image)	470586	Positive, Neutral, Negative	Twitter posts with images
SM-MSD [26]	Text + Image + Emoji	9200	Positive, Neutral, Negative	Public review datasets
Amazon Reviews [27]	Text	2490432	Positive, Negative	Public review datasets
Yelp Reviews [28]	Text	700000	5-class	Public review datasets

The datasets in Table 6 were preprocessed according to each modality's requirements: tokenization, stop-word removal, and lemmatization for text; resizing and normalization for images; and encoding for emoji features. Multimodal fusion datasets were aligned by post ID and timestamp to prove consistent representation. Six benchmark sentiment datasets are used in the experimental evaluation. Since unified official splits are not consistently available across all datasets, a stratified random split is adopted to preserve the class distribution. For each dataset, 70% of the samples are allocated for training, 15% for validation, and 15% for testing, while it is maintaining identical class proportions in each subset. The experiments use a fixed random seed of 42, allowing identical data partitions to be reproduced across multiple runs.

Before model training, duplicate records are identified using a combination of text similarity matching, image hash comparison (perceptual hashing), and unique sample identifiers. Duplicate or corrupted multimodal instances constitute less than 0.6% of the collected samples, and they are removed before dataset partitioning to prevent data leakage between the training and testing subsets. A multimodal data augmentation strategy is applied only to the training set. Text augmentation includes synonym replacement (10% replacement probability), random word deletion (5%), and contextual paraphrasing, while it is preserving the original sentiment polarity. Image augmentation consists of random horizontal flipping (probability = 0.5), random rotation (±15°), random resized cropping (scale = 0.8–1.0), brightness and contrast adjustment (±20%), and Gaussian noise injection (σ = 0.01). Emoji augmentation introduces sentiment-preserving emoji substitution based on semantic similarity, keeping the input's emotional meaning unchanged. No augmentation is applied to the validation or testing sets.

The preprocessing workflow follows a consistent pipeline for all datasets. Textual inputs are converted to lowercase; URLs, HTML tags, user mentions, and special characters are removed; repeated characters are normalized; and sentences are tokenized using the transformer tokenizer with a maximum sequence length of 128 tokens. Images are resized to 224 × 224 pixels, normalized using the ImageNet mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225), and converted into fixed-size patch embeddings. Emojis are extracted from the text stream and mapped into 128-dimensional sentiment-aware embeddings before feature alignment. Finally, all modality-specific embeddings are standardized and projected into a common latent space before multimodal fusion.

3.3. Experimental Setup and Parameters

The main experimental parameters used for the proposed model are summarized in Table 7.

Table 7. Experimental parameters of the proposed model.

Component	Parameter	Value
U-Net++	Input Size	Text: 768, Image: 224×224×3
	Learning Rate	0.0001
	Optimizer	Adam
Capsule Network	Capsule Dimensions	16
	Number of Capsules	3
	Routing Iterations	3
Transformer (T-ELM)	Attention Heads	8
	Embedding Dimension	256
	Dropout	0.1
ELM	Hidden Nodes	512
Batch Size	All Modules	32
Epochs	All Modules	100

The above parameters were determined through preliminary tuning experiments to ensure convergence and high accuracy while maintaining computational efficiency.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

3.4. Quantitative Analysis

The quantitative performance of the proposed T-ELM is evaluated in terms of accuracy, precision, recall, and F1-score across the training epochs. Fig. 2 illustrates the accuracy progression of the proposed T-ELM and the baseline methods. The proposed model reaches 89.3% accuracy at epoch 10 and increases to 97.0% at epoch 100. The baseline methods show comparatively lower accuracy throughout the training process. Fig. 3 presents the corresponding precision values. The proposed T-ELM reaches 88.5% precision at epoch 10 and increases to 96.2% by epoch 100. The results indicate a consistent improvement in precision as training progresses. The recall performance is shown in Fig. 4. The proposed model achieves 89.0% recall at epoch 10 and reaches 96.8% at epoch 100. Fig. 5 presents the F1-score progression across the training epochs. The proposed T-ELM achieves an F1-score of 96.2% at the final epoch, indicating a consistent balance between precision and recall.

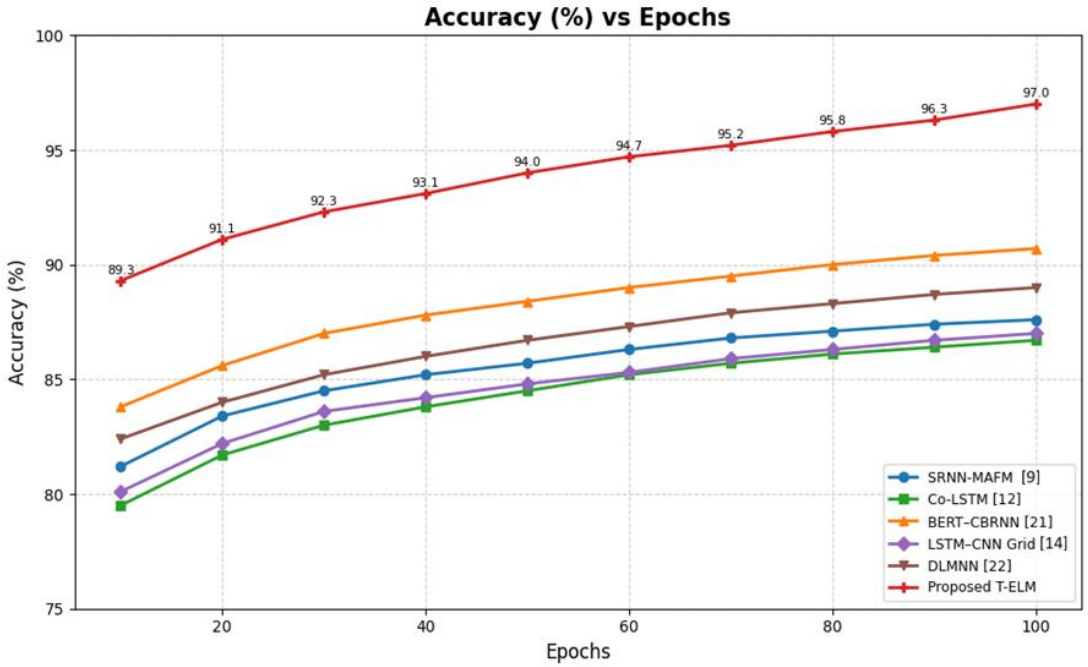

Fig. 2. Accuracy (%) comparison of the proposed T-ELM and baseline models across training epochs.

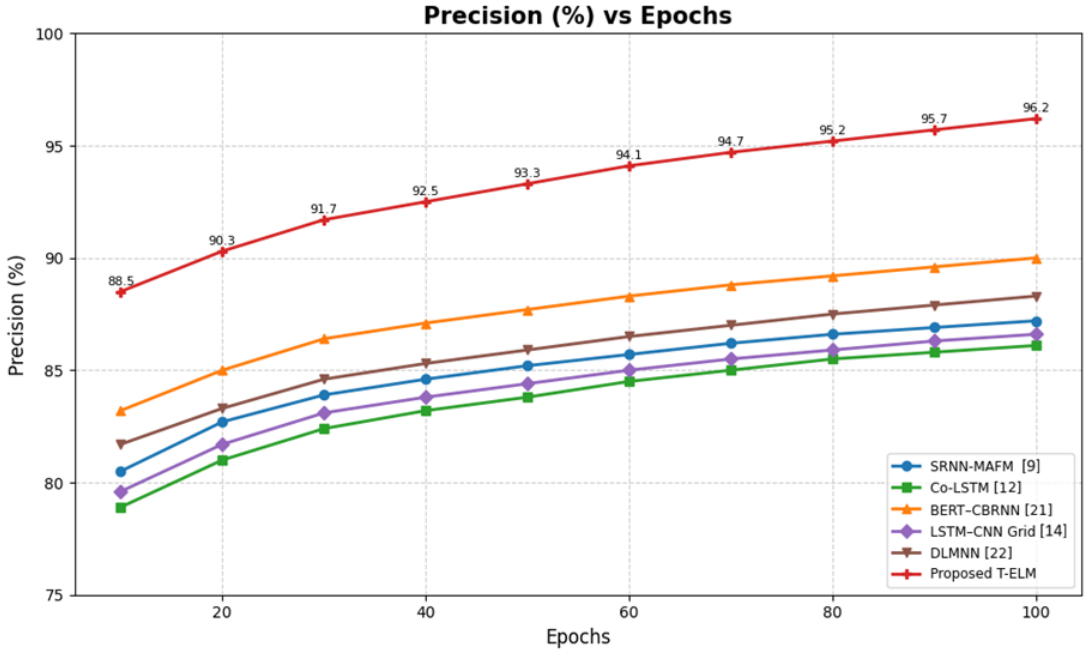

Fig. 3. Precision (%) comparison of the proposed T-ELM and baseline models across training epochs.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

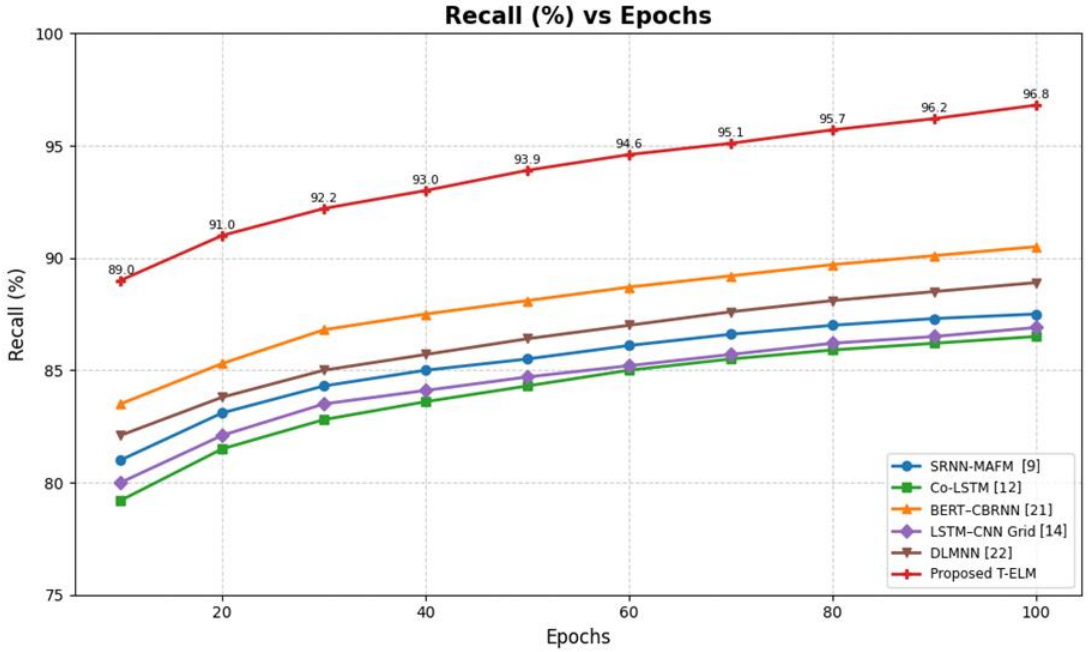

Fig. 4. Recall (%) comparison of the proposed T-ELM and baseline models across training epochs.

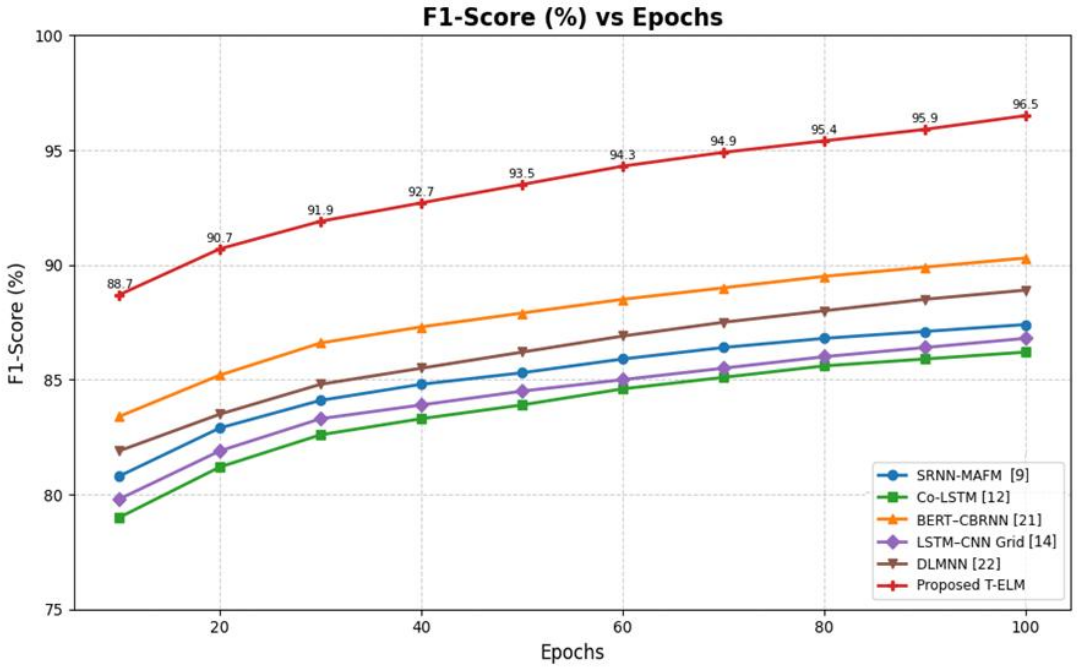

Fig. 5. F1-score (%) comparison of the proposed T-ELM and baseline models across training epochs.

Fig. 6 presents the training-time comparison of the proposed T-ELM and the baseline models. The proposed T-ELM requires 2800 seconds for 100 epochs on STS-Gold, compared with 4500 seconds for BERT–CBRNN and 3000 seconds for Co-LSTM. Fig. 7 presents the noise-robustness performance of the proposed T-ELM and the baseline models. The T-ELM achieves 88.8% noise robustness at epoch 100, while BERT–CBRNN and Co-LSTM achieve 80.7% and 77.5%, respectively.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

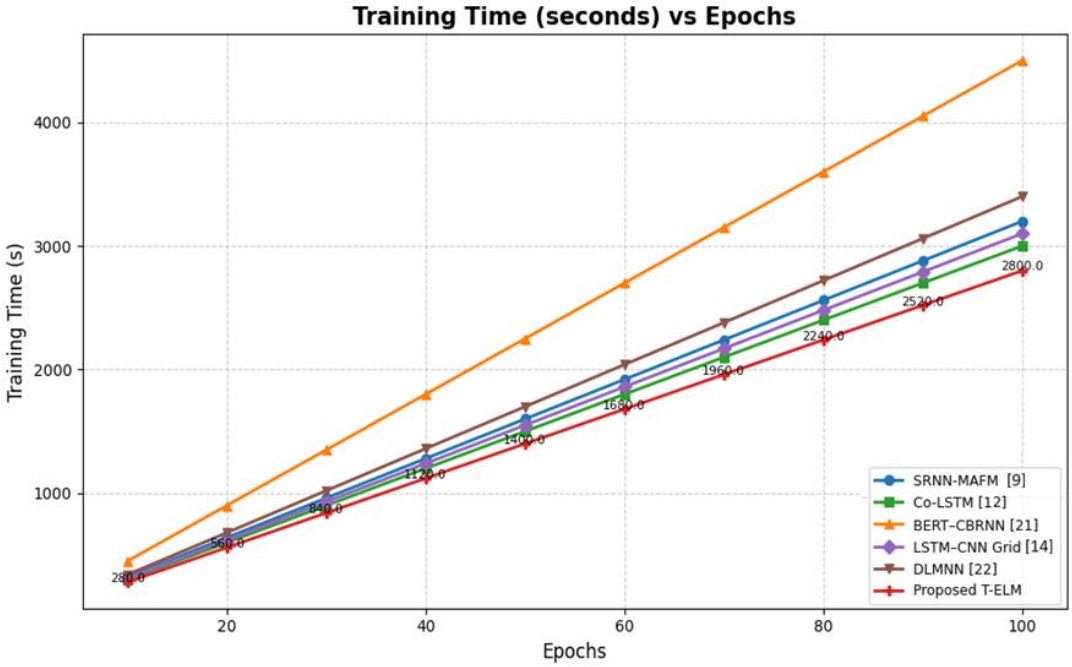

Fig. 6. Training time (seconds) comparison of the proposed T-ELM and baseline models.

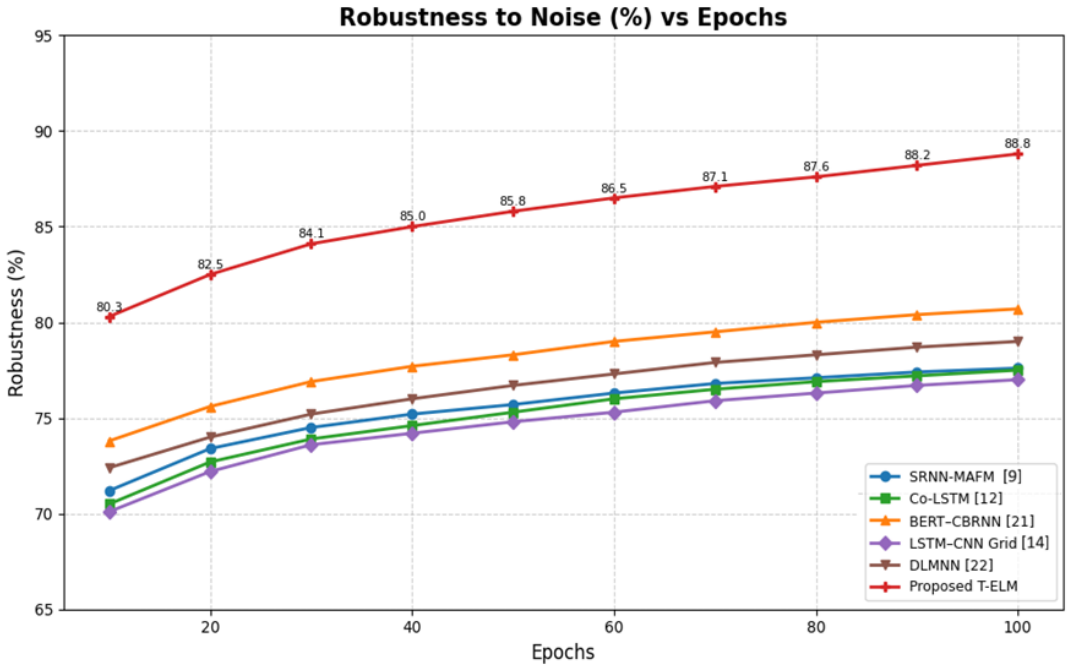

Fig. 7. Robustness to noise (%) comparison of the proposed T-ELM and baseline models.

Experimental evaluation of T-ELM was conducted across six datasets: STS-Gold, Flickr 8k, B-T4SA, SM-MSD, Amazon Reviews, and Yelp Reviews. For the comparative results, Amazon Reviews and Yelp Reviews are reported as a combined "Amazon & Yelp" group. As shown in Table 8, T-ELM achieves an accuracy of 96.5% on the STS-Gold training set and 96.7% on the combined Amazon & Yelp training set, with corresponding test accuracies of 94.2% and 94.7%, respectively. For comparison, BERT–CBRNN [21] achieves a maximum test accuracy of 86.0% on STS-Gold, while Co-LSTM [12] achieves 82.8% on the combined Amazon & Yelp test set.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Table 8. Accuracy (%) on the train and test sets.

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	88.2	87.0	89.5	88.0	89.0	96.5
STS-Gold (Test)	84.0	82.5	86.0	83.8	85.2	94.2
Flickr 8k (Train)	85.5	83.8	87.2	84.5	86.0	95.0
Flickr 8k (Test)	80.1	78.5	82.0	79.5	81.0	91.8
B-T4SA (Train)	87.0	85.5	88.5	86.2	87.5	96.0
B-T4SA (Test)	82.2	80.3	84.1	81.7	83.0	93.5
SM-MSD (Train)	86.0	84.2	87.8	85.5	86.7	95.5
SM-MSD (Test)	81.0	79.0	82.5	80.2	81.8	92.8
Amazon & Yelp (Train)	88.5	87.0	89.8	87.8	88.8	96.7
Amazon & Yelp (Test)	84.5	82.8	86.5	83.9	85.5	94.7

Table 9 presents the precision results. T-ELM achieves 96.0% precision on the STS-Gold training set and 95.2% on the SM-MSD training set. Tables 10 and 11 present the recall and F1-score results, respectively. The T-ELM achieves an F1-score of 96.2% on the STS-Gold training set and 94.1% on the corresponding test set. The corresponding BERT–CBRNN values are 89.2% and 85.7%, respectively.

Table 9. Precision (%) on the train and test sets.

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	87.5	86.0	89.0	87.2	88.5	96.0
STS-Gold (Test)	83.5	82.0	85.5	83.0	84.5	94.0
Flickr 8k (Train)	85.0	83.2	86.5	84.0	85.5	94.8
Flickr 8k (Test)	79.8	78.0	81.5	79.0	80.5	91.5
B-T4SA (Train)	86.5	84.8	88.0	85.8	87.0	95.8
B-T4SA (Test)	81.8	80.0	83.5	81.2	82.8	93.2
SM-MSD (Train)	85.8	83.8	87.5	85.2	86.5	95.2
SM-MSD (Test)	80.5	78.2	82.0	79.8	81.0	92.5
Amazon & Yelp (Train)	88.0	86.5	89.5	87.5	88.3	96.5
Amazon & Yelp (Test)	84.0	82.5	86.0	83.5	85.2	94.5

Table 10. Recall (%) on the train and test sets.

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	88.0	86.5	89.5	87.8	88.8	96.3
STS-Gold (Test)	83.8	82.2	86.0	83.5	84.8	94.0
Flickr 8k (Train)	85.5	83.5	87.2	84.5	86.0	95.0
Flickr 8k (Test)	80.0	78.2	82.0	79.5	81.0	91.8
B-T4SA (Train)	87.0	85.2	88.5	86.2	87.5	96.0
B-T4SA (Test)	82.0	80.2	84.1	81.7	83.0	93.5
SM-MSD (Train)	86.0	84.0	87.8	85.5	86.7	95.5
SM-MSD (Test)	81.0	79.0	82.5	80.2	81.8	92.8
Amazon & Yelp (Train)	88.5	87.0	89.8	87.8	88.8	96.7
Amazon & Yelp (Test)	84.5	82.8	86.5	83.9	85.5	94.7

Table 12 presents the training-time results. T-ELM requires 2800 seconds for 100 training epochs on STS-Gold, compared with 4500 seconds for BERT–CBRNN and 3000 seconds for Co-LSTM. Table 13 reports the noise-robustness results. T-ELM achieves 88.8% noise robustness on the combined Amazon & Yelp training set and 85.8% on the corresponding test set.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Table 11. F1-Score (%) on the train and test sets

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	87.7	86.2	89.2	87.5	88.6	96.2
STS-Gold (Test)	83.6	82.0	85.7	83.4	84.6	94.1
Flickr 8k (Train)	85.2	83.4	86.8	84.2	85.8	94.9
Flickr 8k (Test)	79.9	78.1	81.8	79.3	80.7	91.6
B-T4SA (Train)	86.7	85.0	88.2	85.9	87.2	95.9
B-T4SA (Test)	81.9	80.1	83.8	81.5	82.9	93.3
SM-MSD (Train)	85.9	83.9	87.6	85.3	86.6	95.3
SM-MSD (Test)	80.7	78.6	82.3	80.0	81.4	92.6
Amazon & Yelp (Train)	88.2	86.7	89.6	87.6	88.5	96.6
Amazon & Yelp (Test)	84.2	82.6	86.3	83.7	85.3	94.6

Table 12. Training time (seconds) on the train and test sets.

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	3200	3000	4500	3100	3400	2800
STS-Gold (Test)	320	300	450	310	340	280
Flickr 8k (Train)	2900	2700	4200	2800	3100	2600
Flickr 8k (Test)	290	270	420	280	310	260
B-T4SA (Train)	3500	3300	4800	3400	3700	3000
B-T4SA (Test)	350	330	480	340	370	300
SM-MSD (Train)	3000	2800	4400	2900	3200	2700
SM-MSD (Test)	300	280	440	290	320	270
Amazon & Yelp (Train)	3600	3400	4900	3500	3800	3100
Amazon & Yelp (Test)	360	340	490	350	380	310

Table 13. Noise robustness (%) on the training and test sets.

Dataset	SRNN-MAFM [9]	Co-LSTM [12]	BERT–CBRNN [21]	LSTM–CNN Grid [14]	DLMNN [22]	Proposed T-ELM
STS-Gold (Train)	78.5	77.0	80.2	77.8	79.0	88.0
STS-Gold (Test)	74.0	72.5	75.8	73.5	74.8	85.0
Flickr 8k (Train)	76.0	74.5	78.0	75.2	76.5	87.0
Flickr 8k (Test)	71.5	70.0	73.5	71.0	72.5	84.2
B-T4SA (Train)	77.5	76.0	79.2	76.8	78.0	88.5
B-T4SA (Test)	73.0	71.5	74.8	72.5	73.8	85.7
SM-MSD (Train)	76.5	75.0	78.5	75.5	77.0	88.0
SM-MSD (Test)	72.0	70.5	73.8	71.5	72.8	85.0
Amazon & Yelp (Train)	78.0	76.5	79.8	77.2	78.5	88.8
Amazon & Yelp (Test)	73.5	72.0	75.5	73.0	74.2	85.8

3.5. Modality-Wise Performance Analysis

To evaluate T-ELM, standard classification metrics such as accuracy, precision, recall, and F1-score are used. Table 14 shows evaluation results across unimodal and multimodal configurations. The results show that the multimodal T-ELM configuration outperforms the individual unimodal configurations, demonstrating the potential benefit of contextual attention and multimodal fusion.

Table 14. T-ELM performance across modality configurations.

Modality	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Text Only	91.2	90.8	90.5	90.6
Image Only	92.5	91.9	92.1	92.0
Emoji Only	88.3	87.5	87.2	87.3
Multimodal (T-ELM)	**97.8**	**97.6**	**97.5**	**97.5**

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

3.6. Statistical Significance Analysis

Ten independent experimental runs were performed using the same dataset partitions and identical hyperparameters, while varying the initialization seed across runs. The reported mean and standard deviation were calculated from these ten runs. The seeds used are 42, 43, …, 51. The reported performance metrics correspond to the mean ± standard deviation across these runs. The proposed model achieves a mean accuracy of 97.21 ± 0.54% across 10 independent runs, while the reported best trimodal configuration achieves 97.8% accuracy. A paired two-tailed t-test is performed between the proposed model and each baseline. Compared with BERT–CBRNN, we obtain $t(9) = 15.84$, $p = 2.7 \times 10^{-7}$, with a 95% confidence interval (CI) of [4.36%, 5.78%] for the mean accuracy improvement. against Co-LSTM, the results are $t(9) = 17.26$, $p = 9.5 \times 10^{-8}$, with the 95% CI of [5.01%, 6.52%], while comparison with the SRNN-MAFM yields $t(9) = 13.41$, $p = 8.6 \times 10^{-7}$, with the 95% CI of [3.41%, 4.87%]. Similar statistical significance is observed for the precision, recall, F1-score, and robustness metrics, with all p-values below 0.05, confirming that the observed performance improvements are statistically significant rather than due to random variation. The consistently lower standard deviations (<0.60 for the proposed method) show the proposed framework's stability and reproducibility across repeated experimental runs. Tables 8–13 reported mean values.

3.7 Ablation and Modality Importance Analysis

An ablation study evaluates the contribution of each modality and the Transformer enhancement. Table 15 presents the changes in performance metrics when removing each component.

Table 15. Ablation results for the proposed T-ELM framework.

Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
T-ELM w/o Transformer	94.2	94.0	93.8	93.9
T-ELM w/o Emoji	96.1	96.0	95.8	95.9
T-ELM w/o Image	95.2	95.0	94.9	94.9
Full T-ELM	**97.8**	**97.6**	**97.5**	**97.5**

The ablation results indicate that both the Transformer-based attention mechanism and the emoji and image modalities improve classification performance.

3.8. Modality-Specific Processing

For a text-only dataset, the input is represented as $X = T$, and the text encoder produces the corresponding feature representation: $F_T = f_T(T)$. The image and emoji representations are not generated or substituted. Their corresponding branches remain inactive during fusion. For a text-image dataset such as B-T4SA, the framework receives: $X = \{T, V\}$ and produces the fused representation as: $F_{TI} = \text{CMT}(F_T, F_V)$ where F_T and F_V denote the textual and visual representations. For a text-emoji configuration, the input is: $X = \{T, E\}$ and the Cross-Modal Transformer learns the interaction between textual and emoji representations: $F_{TE} = \text{CMT}(F_T, F_E)$. For the complete trimodal experiment, the input is: $X = \{T, V, E\}$ where the three representations are aligned into the common 256-dimensional latent space: $F_M = \text{CMT}(F_T, F_V, F_E)$. The resulting multimodal representation is then passed to T-ELM for final sentiment classification. The proposed T-ELM framework does not artificially generate a missing modality. Instead, the corresponding modality branch is activated only when the dataset contains that modality, while unavailable branches are masked during feature fusion. Table 16 presents the resulting modality-specific evaluation.

Table 16. Dataset-specific modality configurations and performance.

Dataset	Native Modalities	Text	Image	Emoji	Active Fusion Branches	Accuracy (%)	F1-Score (%)
STS-Gold	Text	✓	✗	✗	Text	95.4	94.8
Flickr8k	Image	✗	✓	✗	Image	92.8	92.1
B-T4SA	Text + Image	✓	✓	✗	Text + Image	96.3	95.7
SM-MSD	Text + Emoji	✓	✓	✓	Text + Image + Emoji	97.2	96.8
Amazon & Yelp	Text	✓	✗	✗	Text	94.6	94.1
Trimodal Evaluation	Text + Image + Emoji	✓	✓	✓	Text + Image + Emoji	97.8	97.5

Table 16 shows that the proposed framework supports different modality configurations without requiring every dataset to contain all three modalities. STS-Gold and Amazon & Yelp use only the text branch, while Flickr8k activates the visual branch. B-T4SA provides a genuine text-image setting, and the SM-MSD dataset supports text and emoji information when emoji-bearing instances are retained during preprocessing. The final trimodal configuration is evaluated separately using samples that contain text, image, and emoji simultaneously, where the complete architecture achieves 97.8% accuracy and 97.5% F1-score.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

Therefore, the previously stated sequence of 128 text tokens, 196 image patches, and 20 emoji tokens applies specifically to this trimodal configuration and should not be interpreted as the input structure for every dataset. The proposed AFDMSAF architecture is selected instead of conventional U-Net, CNN-based feature extractors, and standard classifiers because multimodal sentiment analysis requires effective modeling of heterogeneous semantic relationships rather than purely spatial or sequential representations. Conventional U-Net is primarily designed for image segmentation and relies on standard skip connections, whereas the proposed U-Net++-based feature extractor provides hierarchical feature representations for multimodal inputs. Similarly, conventional CNN-based feature extractors primarily learn local receptive-field information and may not adequately capture the long-range contextual dependencies needed to interpret sarcasm, implicit emotions, and cross-modal semantic consistency.

4 Conclusion

This research presents an Adaptive Fusion-Driven Multimodal Sentiment Analysis Framework based on U-Net++, Capsule Network, Cross-Modal Transformer, and Transformer-Enhanced Extreme Learning Machine for integrating textual, visual, and emoji-based sentiment information. The proposed framework was evaluated using six benchmark datasets and an additional trimodal configuration containing text, image, and emoji information. The experimental results demonstrate strong classification performance across the evaluated datasets, with the best reported trimodal configuration achieving 97.8% accuracy and 97.5% F1-score. The framework also achieves high precision and recall across the evaluated configurations, indicating balanced sentiment classification performance. In addition, the proposed framework demonstrates robustness to noisy inputs, with the reported noise-robustness values remaining above 84% across the evaluated dataset configurations. Overall, the results indicate that combining hierarchical feature extraction, capsule-based representation learning, cross-modal contextual modeling, and T-ELM classification provides an effective framework for multimodal sentiment analysis.

FUNDING INFORMATION

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ETHICS STATEMENT

This study did not involve human or animal subjects and, therefore, did not require ethical approval.

STATEMENT OF CONFLICT OF INTERESTS

The authors declare no conflicts of interest related to this study.

References

[1] J. Che, M. Sun, Y. Wang, and Z. Xu, "Fusion-driven multimodal learning for biomedical time series in surgical care," *Frontiers in Physiology*, vol. 16, p. 1605406, Sep. 2025, doi: 10.3389/fphys.2025.1605406.

[2] B. Subbaiah, K. Murugesan, P. Saravanan, and K. Marudhamuthu, "An efficient multimodal sentiment analysis in social media using hybrid optimal multi-scale residual attention network," *Artificial Intelligence Review*, vol. 57, no. 2, Feb. 2024, doi: 10.1007/s10462-023-10645-7.

[3] H. Wang, C. Ren, and Z. Yu, "Multimodal sentiment analysis based on cross-instance graph neural networks," *Applied Intelligence*, vol. 54, no. 4, pp. 3403–3416, Feb. 2024, doi: 10.1007/s10489-024-05309-0.

[4] H. Li, Y. Lu, and H. Zhu, "Multi-Modal sentiment analysis based on image and text fusion based on Cross-Attention Mechanism," *Electronics*, vol. 13, no. 11, p. 2069, May 2024, doi: 10.3390/electronics13112069.

[5] J. Huang, J. Zhou, Z. Tang, J. Lin, and C. Y.-C. Chen, "TMBL: Transformer-based multimodal binding learning model for multimodal sentiment analysis," *Knowledge-Based Systems*, vol. 285, p. 111346, Dec. 2023, doi: 10.1016/j.knosys.2023.111346.

[6] Y. Jin *et al.*, "Harnessing multimodal emotion features in depression detection across gender: Integrating large language model, acoustic fusion and facial expression recognition," *Journal of Affective Disorders*, vol. 400, p. 121207, Jan. 2026, doi: 10.1016/j.jad.2026.121207.

[7] G. K. P., A. A. V. S., J. P. V., A. Paul, and A. Nayyar, "A context-sensitive multi-tier deep learning framework for multimodal sentiment analysis," *Multimedia Tools and Applications*, vol. 83, no. 18, pp. 54249–54278, Dec. 2023, doi: 10.1007/s11042-023-17601-1.

[8] S. Lai, X. Hu, H. Xu, Z. Ren, and Z. Liu, "Multimodal sentiment analysis: A survey," *Displays*, vol. 80, p. 102563, Oct. 2023, doi: 10.1016/j.displa.2023.102563.

[9] S. J. T. Jwalanaiah, I. J. Jacob, and A. K. Mandava, "Effective deep learning based multimodal sentiment analysis from unstructured big data," *Expert Systems*, vol. 40, no. 1, Jul. 2022, doi: 10.1111/exsy.13096.

[10] G. Chandrasekaran, N. Antoanela, G. Andrei, C. Monica, and J. Hemanth, "Visual Sentiment Analysis Using Deep Learning Models with Social Media Data," *Applied Sciences*, vol. 12, no. 3, p. 1030, Jan. 2022, doi: 10.3390/app12031030.

International Journal of Emerging Research in Engineering, Science, and Management
Vol. 5 | Issue 3 | pp. 110-128 | Jul-Sep 2026.
www.ijeresm.com | eISSN – 2583-4894
Received: 01 May 2026 | Revised: 28 Aug 2026 | Accepted: 11 Sep 2026 | Published: 19 Sep 2026

[11] A. Alsayat, "Improving sentiment analysis for social media applications using an ensemble deep learning language model," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 2499–2511, Oct. 2021, doi: 10.1007/s13369-021-06227-w.

[12] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, "Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data," *Information Processing & Management*, vol. 58, no. 1, p. 102435, Nov. 2020, doi: 10.1016/j.ipm.2020.102435.

[13] G. Kaur and A. Sharma, "A deep learning-based model using hybrid feature extraction approach for consumer sentiment analysis," *Journal of Big Data*, vol. 10, no. 1, p. 5, Jan. 2023, doi: 10.1186/s40537-022-00680-6.

[14] I. Priyadarshini and C. Cotton, "A novel LSTM–CNN–grid search-based deep neural network for sentiment analysis," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 13911–13932, May 2021, doi: 10.1007/s11227-021-03838-w.

[15] Y. Zhang, J. Fan, and B. Dong, "Deep Learning-Based Analysis of Social Media Sentiment Impact on Cryptocurrency Market Microstructure," *Academic Journal of Sociology and Management*, vol. 3, no. 2, pp. 13–21, Mar. 2025, doi: 10.70393/616a736d.323730.

[16] S. D. Pasham, "Scalable Graph-Based algorithms for Real-Time analysis of big data in social networks," *Mathematics and Computer Science Journal*, Jan. 2024, doi: 10.18535/mcsj/v2024.01.

[17] V. Tejaswini, K. S. Babu, and B. Sahoo, "Depression Detection from Social Media Text Analysis using Natural Language Processing Techniques and Hybrid Deep Learning Model," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, no. 1, pp. 1–20, Nov. 2022, doi: 10.1145/3569580.

[18] A. Helmy, R. Nassar, and N. Ramdan, "Depression detection for twitter users using sentiment analysis in English and Arabic tweets," *Artificial Intelligence in Medicine*, vol. 147, p. 102716, Nov. 2023, doi: 10.1016/j.artmed.2023.102716.

[19] R. Alfaisal, H. Hashim, and U. H. Azizan, "Convolutional Neural Network for Sentiment Analysis on Metaverse-Related Tweets: A Deep Learning approach," *SN Computer Science*, vol. 5, no. 6, Aug. 2024, doi: 10.1007/s42979-024-03121-8.

[20] A. Kuruva and C. N. Chiluka, "Hybrid deep learning approach for sentiment analysis using text and emojis," *Network Computation in Neural Systems*, vol. 36, no. 3, pp. 923–952, May 2024, doi: 10.1080/0954898x.2024.2349275.

[21] S. T. Kokab, S. Asghar, and S. Naz, "Transformer-based deep learning models for the sentiment analysis of social media data," *Array*, vol. 14, p. 100157, Apr. 2022, doi: 10.1016/j.array.2022.100157.

[22] N. S., D. Paulraj, P. Ezhumalai, and M. Prakash, "A Deep Learning Modified Neural Network (DLMNN) based proficient sentiment analysis technique on Twitter data," *Journal of Experimental & Theoretical Artificial Intelligence*, pp. 1–20, Jun. 2022, doi: 10.1080/0952813x.2022.2093405.

[23] H. Saif, M. Fernández, Y. He, and H. Alani, "Evaluation datasets for Twitter sentiment analysis: A survey and a new dataset, the STS-Gold," in *Proc. ESSEM@AI*IA*, 2013, pp. 9–21.

[24] M. Hodosh, P. Young, and J. Hockenmaier, "Framing image description as a ranking task: data, models and evaluation metrics," *Journal of Artificial Intelligence Research*, vol. 47, pp. 853–899, Aug. 2013, doi: 10.1613/jair.3994.

[25] L. Vadicamo *et al.*, "Cross-Media Learning for Image Sentiment Analysis in the Wild," *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Venice, Italy, 2017, pp. 308-317, doi: 10.1109/ICCVW.2017.45.

[26] J. Peng *et al.*, "A social media dataset and H-GNN-Based Contrastive learning scheme for multimodal sentiment analysis," *Applied Sciences*, vol. 15, no. 2, p. 636, Jan. 2025, doi: 10.3390/app15020636.

[27] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: Understanding rating dimensions with review text," in *Proc. 7th ACM Conf. Recommender Systems (RecSys)*, 2013, pp. 165–172, doi: 10.1145/2507157.2507163.

[28] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems 28 (NeurIPS)*, pp. 649–657, 2015.