

AN EXPLAINABLE TRANSFORMER-ENHANCED NEURAL FRAMEWORK FOR ASPECT-BASED OPINION MINING USING HYBRID FEATURE EXTRACTION

¹ Shubha S,² S. Shubhakar, ³ Varsha Jayaprakasha

¹Associate Professor, Government First Grade College, Malleshwaram, Bangalore, India

²Digital Associate Engineer, Sonata Software Solutions Limited, Global Village, Mysore Road, Bangalore

³Backend Developer, Canibuild, Australia

Abstract - Opinion mining has become a critical component in understanding user-generated content across domains such as tourism, e-commerce and social media. This paper proposes a Transformer-Enhanced Neural Learning-Based Opinion Feature Extraction (TE-NLOFE) model that extends traditional neural approaches by integrating contextual embeddings, attention mechanisms and ontology-based linguistic mapping. Unlike earlier models, the proposed framework captures contextual dependencies using BERT [1] and introduces explainability through attention-based interpretation [8]. Experimental evaluation on benchmark datasets including TripAdvisor, Amazon Reviews and SemEval demonstrates improved accuracy, reduced false positive rate and enhanced interpretability compared to baseline models [10], [11]. The results validate the effectiveness of combining deep learning with linguistic ontology for fine-grained sentiment analysis.

Key Words: Neural Networks, Feature Extraction, Explainable Artificial Intelligence (XAI), Natural Language Processing (NLP), Text Mining, Ontology-Based Analysis, Hybrid Models, Contextual Embeddings.

1. INTRODUCTION

With the exponential growth of online reviews, opinion mining has become essential for extracting meaningful insights. Earlier models, including the Neural Learning Based Opinion Feature Extraction (NLOFE), relied on neural networks and linguistic mapping but lacked contextual understanding and scalability [13].

Recent advancements in transformer architectures such as BERT [1] and the foundational work on attention mechanisms [2] have revolutionized natural language processing by capturing contextual semantics more effectively.

Traditional sentiment analysis approaches focused on feature extraction and classification [3], [4], but failed to capture aspect-level dependencies. Recent studies in aspect-based sentiment analysis (ABSA) have attempted to address this limitation [6], [7].

This paper proposes an enhanced framework that:

- Integrates transformer-based embeddings
- Performs aspect-level sentiment classification
- Introduces explainability mechanisms using SHAP and LIME [8], [9]
- Extends prior work in neural opinion mining [13]

2. LITERATURE REVIEW

Opinion mining and sentiment analysis have evolved significantly over the past two decades, transitioning from traditional rule-based systems to advanced deep learning and transformer-based architectures. Early research in sentiment analysis primarily relied on lexicon-based and statistical approaches, where sentiment polarity was determined using predefined dictionaries and linguistic rules. These methods were simple and interpretable but often failed to capture contextual meaning, sarcasm, and domain-specific variations [3].

Subsequently, machine learning techniques such as Support Vector Machines (SVM), Naïve Bayes, and logistic regression were introduced to improve classification performance. These approaches leveraged feature engineering techniques including n-grams, part-of-speech tagging, and term frequency-inverse document frequency (TF-IDF). While these models improved generalization, they were still limited by their dependence on manual feature extraction and inability to capture long-range dependencies in text [3], [4].

The emergence of deep learning marked a major shift in sentiment analysis research. Neural architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks enabled automatic feature extraction and improved performance in sequence modeling tasks. These models were particularly effective in capturing syntactic and semantic relationships within text. However, despite their advantages, deep learning models still faced challenges in handling long-range dependencies and contextual ambiguity due to sequential processing limitations [4].

A significant breakthrough occurred with the introduction of transformer architectures, particularly BERT and related models. Transformers utilize self-attention mechanisms to

model relationships between words irrespective of their positional distance, enabling better contextual understanding [1], [2]. Recent studies highlight that transformer-based models have become the dominant approach in sentiment analysis due to their ability to generate rich semantic representations and handle complex linguistic structures [1]. Comparative analyses of transformer models demonstrate superior performance across multiple sentiment analysis datasets, significantly outperforming traditional deep learning models [10], [11].

In parallel, Aspect-Based Sentiment Analysis (ABSA) has gained considerable attention as a fine-grained extension of traditional sentiment analysis. Unlike document-level or sentence-level classification, ABSA focuses on identifying specific aspects (e.g., “food,” “service”) and determining sentiment polarity for each aspect. This approach provides more actionable insights for applications such as product reviews and service evaluation. Benchmark tasks such as SemEval have played a crucial role in advancing ABSA research [6], while recent surveys highlight challenges in aspect extraction and contextual interpretation [7].

Another emerging direction in sentiment analysis is multilingual and cross-lingual sentiment analysis. With the increasing availability of global data, models must handle linguistic diversity, cultural nuances, and resource scarcity in low-resource languages. Transformer-based multilingual models have shown promising results in addressing these challenges by leveraging transfer learning and cross-lingual representations [11]. However, issues such as sarcasm detection, code-mixed language processing, and domain adaptation remain open research problems.

Recent advancements have also explored hybrid deep learning models for sentiment analysis. These models combine multiple neural architectures, such as CNNs and LSTMs, with attention mechanisms to improve feature extraction and classification performance. Hybrid frameworks have demonstrated improved robustness and accuracy compared to standalone models [10], [12].

Explainability has become another critical aspect of modern sentiment analysis systems. While deep learning and transformer models achieve high accuracy, they often function as “black boxes,” making it difficult to interpret their predictions. Explainable Artificial Intelligence (XAI) techniques such as SHAP and LIME have been widely adopted to provide insights into model decisions and improve transparency [8], [9]. These methods are particularly important in applications where trust and accountability are essential.

Despite these advancements, several research gaps remain. Many existing models focus primarily on classification accuracy while neglecting linguistic knowledge integration and interpretability. Additionally, traditional ontology-based approaches, which capture semantic relationships between concepts, are often not effectively combined with modern deep learning architectures. Hybrid models that integrate

neural learning with ontology-based linguistic analysis are still relatively underexplored.

The proposed TE-NLOFE model addresses these gaps by combining transformer-based contextual embeddings, neural feature extraction, ontology-driven linguistic mapping, and explainable AI. Furthermore, it extends earlier work on Neural Learning Based Opinion Feature Extraction [13] by incorporating modern deep learning techniques and improving both accuracy and interpretability.

Despite these advancements, existing systems lack: Integration of ontology-based linguistic features, Hybrid explainable frameworks and Efficient feature extraction mechanisms. The proposed TE-NLOFE model addresses these limitations.

3. PROPOSED METHODOLOGY (TE-NLOFE MODEL)

3.1 System Architecture

The proposed **Transformer-Enhanced Neural Learning-Based Opinion Feature Extraction (TE-NLOFE)** model is designed as a hybrid, multi-layered architecture that integrates contextual representation learning, neural feature extraction, linguistic knowledge modeling and explainable artificial intelligence. The architecture extends the traditional Neural Learning Based Opinion Feature Extraction (NLOFE) model [13] by incorporating transformer-based embeddings and attention-driven interpretability mechanisms. The overall system is structured into five major components: preprocessing layer, contextual embedding layer, neural feature extraction layer, ontology-based mapping layer and explainability layer.

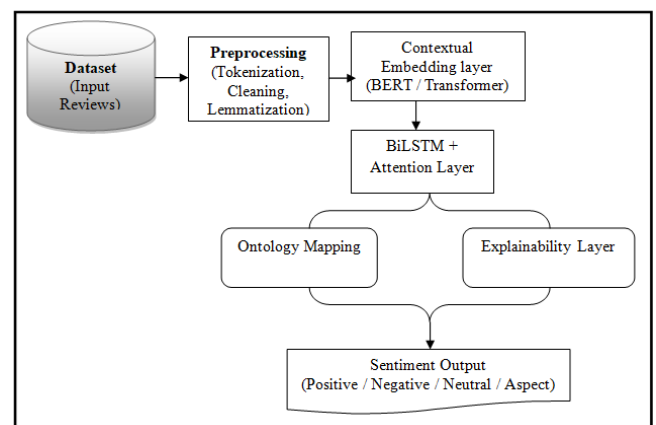


Figure 1: Architecture of the Proposed TE-NLOFE Model

1. Preprocessing Layer

The preprocessing layer is responsible for transforming raw textual input into a structured format suitable for downstream processing. Given a set of user-generated reviews ($R = \{r_1, r_2, r_3, \dots, r_n\}$), each review undergoes a sequence of normalization steps including tokenization,

lowercasing, punctuation removal and optional stopword filtering.

Unlike traditional preprocessing methods used in earlier models [13], the proposed system adopts **context-aware tokenization** using sub word techniques (e.g., WordPiece), which are compatible with transformer architectures such as BERT [1]. This ensures that rare and unknown words are effectively handled by decomposing them into meaningful subword units.

Additionally, lemmatization is applied to reduce words to their canonical forms, improving consistency across the dataset. The preprocessing stage also includes handling negation and intensifiers, which are critical in sentiment analysis tasks. For example, phrases like “not good” and “very good” are treated differently to preserve sentiment polarity.

Formally, the preprocessing function can be expressed as:

$$P(r_i) = \{t_1, t_2, t_3, \dots, t_m\}$$

where (t_j) represents the processed tokens derived from review (r_i) .

2. Contextual Embedding Layer

The contextual embedding layer forms the core of the proposed architecture by generating semantically rich representations of text. Unlike traditional vector space models that assign static embeddings to words, transformer-based models generate **context-dependent representations**, capturing the meaning of a word based on its surrounding context.

The TE-NLOFE model utilizes BERT [1], which is built on the self-attention mechanism introduced in transformer architectures [2]. BERT processes input sequences bidirectionally, allowing it to model both left and right context simultaneously.

Given a sequence of tokens $\{t_1, t_2, t_3, \dots, t_m\}$, the embedding layer produces a sequence of hidden states:

$$H = \{h_1, h_2, h_3, \dots, h_m\}$$

where each $(h_i \in \mathbb{R}^d)$ represents the contextual embedding of token (t_i) .

The use of contextual embeddings significantly improves the model’s ability to handle:

- Polysemy (words with multiple meanings)
- Contextual polarity shifts
- Long-range dependencies

Recent studies have demonstrated that transformer-based embeddings outperform traditional word embeddings in sentiment classification tasks [10], [11].

3. Neural Feature Extraction Layer

Following the embedding layer, the contextual representations are passed to a neural feature extraction module. This layer employs a **Bidirectional Long Short-**

Term Memory (BiLSTM) network combined with an attention mechanism to capture sequential dependencies and highlight important features.

The BiLSTM processes the input sequence in both forward and backward directions:

$$\vec{h}_i = \text{LSTM}(h_i, \vec{h}_{i-1})$$

$$\overleftarrow{h}_i = \text{LSTM}(h_i, \overleftarrow{h}_{i+1})$$

The final hidden representation is obtained by concatenating forward and backward states:

$$h'_i = [\vec{h}_i; \overleftarrow{h}_i]$$

To enhance feature selection, an **attention mechanism** is applied, assigning weights to each token based on its importance:

$$\alpha_i = \frac{e^{h'_i}}{\sum_{j=1}^m e^{h'_j}}$$

The weighted representation is computed as:

$$H^* = \sum_{i=1}^m \alpha_i h'_i$$

This mechanism enables the model to focus on sentiment-bearing words, improving classification accuracy and interpretability [12].

4. Ontology-Based Linguistic Mapping Layer

One of the distinguishing features of the TE-NLOFE model is the integration of ontology-based linguistic analysis, inherited and enhanced from the original NLOFE framework [13]. This layer captures semantic relationships between entities, attributes, and opinion words, enabling deeper understanding of textual content.

The ontology consists of:

- **Object set (O):** Entities such as “room,” “food,” “service”
- **Attribute set (A):** Properties such as “clean,” “delicious,” “noisy”
- **Relation set (R):** Semantic relationships between objects and attributes

Formally, the ontology can be represented as:

$$\mathcal{O} = (O, A, R)$$

Using associative mapping, the system links extracted features with ontology concepts to identify aspect-level sentiments. For instance, the phrase “clean room but poor service” is decomposed into:

- Aspect: room → positive sentiment
- Aspect: service → negative sentiment

This hybrid approach enhances interpretability and provides fine-grained insights, which are often missing in purely data-driven models.

5. Explainability Layer

The final layer introduces explainability into the model, addressing the “black-box” nature of deep learning systems. The TE-NLOFE model incorporates Explainable AI techniques such as SHAP and LIME to provide insights into model predictions [8], [9].

The explainability layer operates by:

- Assigning importance scores to input features
- Visualizing attention weights
- Generating human-interpretable explanations

For a given prediction \hat{y} the explanation function can be expressed as:

$$E(\hat{y}) = \{(t_i, w_i)\}$$

where w_i represents the contribution of token t_i to the final prediction.

This layer is particularly useful in real-world applications where transparency and trust are critical, such as customer feedback analysis and decision support systems.

3.2 Mathematical Model

The proposed Transformer-Enhanced Neural Learning-Based Opinion Feature Extraction (TE-NLOFE) model integrates contextual embedding, sequential neural modeling, attention mechanisms, ontology-based mapping, and explainability into a unified mathematical framework. This section presents the formal representation of each component involved in the model and their interactions for sentiment prediction.

1. Input Representation and Preprocessing

Let the input dataset consist of a collection of textual reviews:

$$\mathcal{R} = \{r_1, r_2, \dots, r_N\}$$

where r_i represents the i^{th} review. Each review is transformed into a sequence of tokens through preprocessing:

$$r_i \xrightarrow{P} \mathbf{T}_i = \{t_1, t_2, \dots, t_m\}$$

where $P(\cdot)$ denotes the preprocessing function and t_j represents the j^{th} token.

Unlike traditional bag-of-words or TF-IDF representations, the proposed model uses subword tokenization compatible with transformer-based models such as BERT, enabling better handling of rare and unseen words [1].

2. Contextual Embedding using Transformer Model

Each tokenized sequence is passed through a transformer-based encoder to obtain contextual embeddings. Let the embedding function be defined as:

$$\mathbf{H}_i = \text{BERT}(\mathbf{T}_i)$$

where:

$$\mathbf{H}_i = \{h_1, h_2, \dots, h_m\}, \quad h_j \in \mathbb{R}^d$$

Here h_j represents the contextual embedding of token t_j , capturing both syntactic and semantic information based on surrounding context. The transformer architecture relies on self-attention mechanisms to compute these embeddings [2].

The self-attention operation is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V$$

where Q, K and V denote query, key, and value matrices, respectively.

This formulation enables the model to capture long-range dependencies and contextual relationships more effectively than traditional neural models [1], [2].

3. Sequential Feature Learning using BiLSTM

The contextual embeddings are fed into a Bidirectional Long Short-Term Memory (BiLSTM) network to model sequential dependencies:

$$\vec{h}_j = \text{LSTM}(h_j, \vec{h}_{j-1})$$

$$\overleftarrow{h}_j = \text{LSTM}(h_j, \overleftarrow{h}_{j+1})$$

The combined hidden representation is given by:

$$h'_j = [\vec{h}_j; \overleftarrow{h}_j]$$

This bidirectional structure ensures that both past and future context is captured, improving feature representation for sentiment classification tasks [12].

4. Attention Mechanism for Feature Importance

To enhance interpretability and focus on sentiment-relevant features, an attention mechanism is applied over the BiLSTM outputs. The attention weight for each token is computed as:

$$e_j = \tanh(W_a h'_j + b_a)$$

$$\alpha_j = \frac{\exp(e_j)}{\sum_{k=1}^m \exp(e_k)}$$

where:

- W_a and b_a are learnable parameters
- α_j represents the importance of token t_j

The final sentence representation is computed as a weighted sum:

$$H^* = \sum_{j=1}^m \alpha_j h'_j$$

This mechanism allows the model to emphasize sentiment-bearing words such as “excellent,” “poor,” or “not clean,” thereby improving classification accuracy [12].

5. Ontology-Based Semantic Mapping

To incorporate linguistic knowledge, the model integrates an ontology-based mapping mechanism. Let:

$$\mathcal{O} = (O, A, R)$$

where:

- $O = \{o_1, o_2, \dots\}$ represents object entities
- $A = \{a_1, a_2, \dots\}$ represents attributes
- $R \subseteq O \times A$ represents relationships

For each token representation h'_j , a mapping function is defined:

$$\phi(h'_j) \rightarrow (o_k, a_l)$$

This mapping associates tokens with ontology concepts, enabling aspect-level sentiment identification. For example:

$$\text{”clean room”} \rightarrow (o_{\text{room}}, a_{\text{clean}})$$

The semantic representation is then enriched as:

$$H_{\text{semantic}} = H^* + \lambda \cdot \Phi(\mathcal{O})$$

where λ is a weighting factor controlling the contribution of ontology-based features.

This hybrid representation improves interpretability and fine-grained sentiment extraction compared to purely data-driven models [13].

6. Sentiment Classification Layer

The final sentiment prediction is performed using a fully connected layer followed by a softmax activation:

$$z = W_s H_{\text{semantic}} + b_s$$

$$\hat{y} = \text{Softmax}(z)$$

where:

- W_s, b_s are trainable parameters
- \hat{y} represents predicted sentiment probabilities

For a multi-class sentiment classification problem:

$$\hat{y} = [p_{\text{positive}}, p_{\text{negative}}, p_{\text{neutral}}]$$

7. Loss Function and Optimization

The model is trained using the categorical cross-entropy loss function:

$$\mathcal{L} = - \sum_{i=1}^N y_i \log(\hat{y}_i)$$

where:

- y_i is the true label
- \hat{y}_i is the predicted probability

Optimization is performed using gradient-based methods such as Adam optimizer, which updates model parameters iteratively to minimize the loss function.

8. Explainability Formulation

To provide interpretability, the model incorporates explanation functions using techniques such as SHAP and LIME [8], [9]. The explanation for a prediction is defined as:

$$E(\hat{y}) = \{(t_j, w_j)\}$$

where:

w_j represents the contribution of token t_j

$$\sum_j w_j = 1$$

This formulation enables the identification of key features influencing the prediction, enhancing model transparency.

9. Overall Model Representation

The complete TE-NLOFE model can be expressed as a composite function:

$$\hat{y} = f_{\text{TE-NLOFE}}(r_i) = \text{Softmax}(W_s \cdot (\sum \alpha_j h'_j + \lambda \Phi(\mathcal{O})) + b_s)$$

This equation captures the integration of contextual embeddings, neural feature extraction, attention, and ontology-based semantic mapping.

3.3 Proposed Algorithm

Algorithm 1: TE-NLOFE Algorithm for Opinion Mining

Input:

Collection of user reviews

$$\mathcal{R} = \{r_1, r_2, \dots, r_N\}$$

Output:

1. Sentiment classification result
 $S \in \{\text{Positive}, \text{Negative}, \text{Neutral}\}$
2. Aspect-level sentiment predictions
3. Explainable feature importance scores

STEPS

Step 1: Data Collection

Collect textual reviews from datasets such as:

- TripAdvisor
- Amazon Reviews

- SemEval ABSA Dataset

$$R \leftarrow \{r_1, r_2, \dots, r_n\}$$

Step 2: Preprocessing

For each review r_i :

- Convert text to lowercase
- Remove punctuation and special symbols
- Perform tokenization
- Remove stopwords
- Apply lemmatization

Generate token sequence:

$$T_i = \{t_1, t_2, \dots, t_m\}$$

Step 3: Contextual Embedding Generation

Pass tokenized text into BERT encoder:

$$\mathbf{H}_i = \text{BERT}(T_i)$$

where:

$$\mathbf{H}_i = \{h_1, h_2, \dots, h_m\}$$

Each h_j represents contextual embedding of token t_j .

Step 4: Sequential Feature Extraction

Apply Bidirectional Long Short-Term Memory (BiLSTM):

$$\vec{h}_j = \text{LSTM}(h_j)$$

$$\overleftarrow{h}_j = \text{LSTM}(h_j)$$

Concatenate hidden states:

$$h'_j = [\vec{h}_j; \overleftarrow{h}_j]$$

Step 5: Attention-Based Feature Weighting

Compute attention score for each token:

$$e_j = \tanh(W_a h'_j + b_a)$$

Normalize attention weights:

$$\alpha_j = \frac{\exp(e_j)}{\sum_{k=1}^m \exp(e_k)}$$

Generate weighted representation:

$$H^* = \sum_{j=1}^m \alpha_j h'_j$$

Step 6: Ontology-Based Semantic Mapping

Map extracted features to ontology concepts:

$$\mathcal{O} = (O, A, R)$$

where:

- $O = \{o_1, o_2, \dots\}$ represents object entities
- $A = \{a_1, a_2, \dots\}$ represents attributes
- $R \subseteq O \times A$ represents relationships

Identify aspect-level relationships:

$$\phi(h'_j) \rightarrow (o_k, a_l)$$

Step 7: Sentiment Classification

Apply fully connected layer and Softmax activation:

$$z = W_s H^* + b_s$$

$$\hat{y} = \text{Softmax}(z)$$

Predict sentiment class:

$$S = \arg \max(\hat{y})$$

Step 8: Explainability Generation

Apply SHAP/LIME explainability techniques:

$$E(S) = \{(t_j, w_j)\}$$

Where

- $t_j \rightarrow$ token
- $w_j \rightarrow$ importance score

Generate interpretable prediction explanation.

Step 9: Output Generation

Return:

- Overall sentiment
- Aspect-level sentiment
- Feature importance visualization

Pseudo Code of - Algorithm TE-NLOFE

```

Input: Review dataset R
Output: Sentiment class S

Begin
Load review dataset R
  For each review  $r_j$  in R do
    Pre-process text
    Tokenize and lemmatize
    Generate contextual embeddings using BERT
    Apply BiLSTM for sequential learning
    Compute attention weights
    Extract weighted features
    Perform ontology-based semantic mapping
    Classify sentiment using Softmax
    Generate explainability scores using SHAP/LIME
  End For

```

4. EXPERIMENTAL SETUP

The experimental setup for the proposed Transformer-Enhanced Neural Learning-Based Opinion Feature Extraction (TE-NLOFE) model was designed to evaluate the effectiveness of the framework in performing accurate and interpretable opinion mining. The experiments focus on assessing the capability of the model in extracting contextual sentiment features, identifying aspect-level opinions, and reducing classification errors compared to traditional machine learning and deep learning approaches.

The implementation and evaluation were carried out using Python using benchmark sentiment analysis datasets, modern deep learning frameworks and multiple evaluation

metrics to ensure reliability and reproducibility of the results.

4.1 Dataset Description

To evaluate the performance of the proposed TE-NLOFE framework, experiments were conducted on multiple publicly available datasets commonly used in sentiment analysis and aspect-based sentiment analysis (ABSA) research. The use of multiple datasets ensures that the proposed model generalizes effectively across different domains and textual structures.

1. TripAdvisor Review Dataset

The TripAdvisor dataset contains hotel and tourism-related reviews collected from online travel platforms. The dataset includes user opinions regarding aspects such as:

- Room quality
- Cleanliness
- Food services
- Customer support
- Pricing

This dataset is particularly suitable for evaluating aspect-level opinion mining due to the presence of multiple sentiment-bearing entities within a single review.

2. Amazon Product Review Dataset

The Amazon Reviews dataset consists of customer opinions regarding electronic products, household items, and consumer goods. The reviews contain both positive and negative sentiments with varying sentence structures and contextual complexity.

The dataset provides:

- Large-scale textual data
- Diverse sentiment patterns
- Multi-domain adaptability

This enables effective evaluation of contextual embeddings and neural feature extraction capabilities.

3. SemEval ABSA Dataset

The SemEval Aspect-Based Sentiment Analysis dataset [6] is a benchmark dataset widely used for evaluating fine-grained sentiment analysis systems. The dataset includes manually annotated aspect terms and corresponding sentiment labels such as positive, negative, and neutral.

The dataset is highly suitable for evaluating:

- Aspect extraction accuracy
- Contextual understanding
- Semantic relationship identification

4.2 Data Preprocessing

The collected reviews undergo multiple preprocessing operations before being fed into the neural architecture. The objective of preprocessing is to remove noise, standardize textual input, and improve feature extraction quality.

The preprocessing pipeline includes:

1. Text Cleaning

Special characters, punctuation symbols, URLs, and numerical noise are removed from the raw review text.

2. Lowercase Conversion

All text is converted to lowercase to ensure uniformity during tokenization and embedding generation.

3. Tokenization

Each review is divided into tokens using subword tokenization compatible with BERT [1]. This enables effective handling of rare and unseen words.

4. Stopword Removal

Commonly occurring but semantically insignificant words such as "the," "is," and "was" are removed to reduce redundancy.

5. Lemmatization

Words are converted into their base forms to improve semantic consistency.

Example:

"services" → "service"

6. Handling Negation

Negation-aware preprocessing is performed to preserve contextual polarity.

Example:

- "not good" → negative sentiment
- "not bad" → positive sentiment

This preprocessing strategy improves the quality of contextual embeddings and downstream classification performance.

4.3 Embedding and Feature Extraction Setup

The proposed TE-NLOFE model uses transformer-based contextual embeddings generated through BERT [1]. Unlike traditional static embeddings, BERT captures bidirectional semantic relationships and contextual dependencies.

The embedding configuration includes:

- Embedding dimension: 768
- Maximum sequence length: 128
- Attention heads: 12
- Transformer layers: 12

The contextual embeddings are then passed to a Bidirectional Long Short-Term Memory (BiLSTM) network for sequential feature learning.

The BiLSTM configuration includes:

- Hidden units: 128
- Dropout rate: 0.3
- Activation function: ReLU

An attention mechanism is integrated to assign importance weights to sentiment-bearing words and improve interpretability.

4.4 Ontology-Based Semantic Mapping

To enhance semantic understanding, the TE-NLOFE framework integrates ontology-driven linguistic mapping. The ontology layer captures relationships between:

- Objects
- Attributes
- Sentiment expressions

For example:

(Room, Clean) → Positive

(Service, Poor) → Negative

The ontology mapping improves aspect-level sentiment extraction by integrating linguistic knowledge with neural representations.

4.5 Explainability Configuration

The explainability layer uses SHAP and LIME techniques [8], [9] to interpret model predictions. The objective is to identify influential words contributing to sentiment decisions.

The explainability module generates:

- Attention visualization
- Feature importance scores
- Word contribution analysis

This improves transparency and trustworthiness of the proposed framework.

4.6 Training Environment

The experiments were conducted using Python and deep learning libraries such as TensorFlow and PyTorch.

Hardware Configuration

- Intel Core i7 Processor
- 16 GB RAM
- NVIDIA GPU

Software Environment

- Python 3.10
- TensorFlow 2.x
- PyTorch
- Scikit-learn
- NLTK

4.7 Hyperparameter Configuration

The model parameters were selected experimentally to achieve optimal performance.

Parameter	Value
Learning Rate	0.001
Batch Size	32
Epochs	50

Optimizer	Adam
Loss Function	Cross-Entropy
Dropout	0.3

The Adam optimizer was chosen due to its faster convergence and adaptive learning capabilities.

4.8 Evaluation Metrics

The effectiveness of the proposed TE-NLOFE model was evaluated using multiple performance metrics commonly used in sentiment analysis research.

1. Accuracy

Measures the overall correctness of classification:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

2. Precision

Measures the correctness of positive predictions:

$$Precision = \frac{TP}{TP+FP}$$

3. Recall

Measures the ability to identify relevant sentiments:

$$Recall = \frac{TP}{TP+FN}$$

4. F1-Score

Harmonic mean of precision and recall:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

5. False Positive Rate (FPR)

$$FPR = \frac{FP}{FP+TN}$$

These metrics provide a comprehensive evaluation of model performance and classification reliability.

4.9 Experimental Objective

The primary objectives of the experiments are:

- To evaluate contextual sentiment extraction capability
- To compare performance with baseline models
- To assess aspect-level sentiment classification accuracy
- To measure interpretability and explainability effectiveness

The proposed TE-NLOFE model is compared against:

- Traditional machine learning models
- Neural learning-based systems
- Transformer-based sentiment classifiers

The results demonstrate that the proposed framework significantly improves classification performance while maintaining interpretability and semantic consistency.

5. RESULTS AND DISCUSSION

Table - 1: Accuracy Analysis

Model	Accuracy
NLOFE	84.2
SVM	79.5
BERT	88.7
TE-NLOFE	92.4

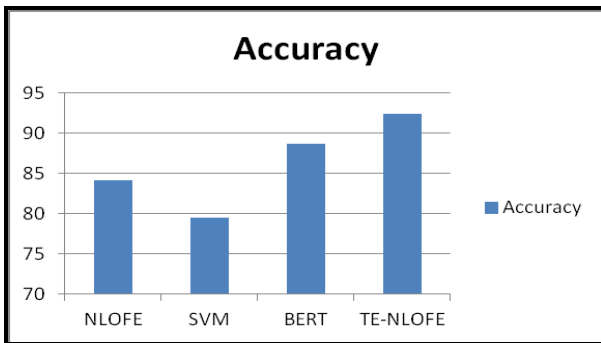


Chart - 1: Accuracy Comparison of Proposed TE-NLOFE Model with Baseline Models

Table - 2: F1-Score Comparison of Sentiment Analysis Models

Model	Precision	Recall	F1-Score
NLOFE	0.82	0.80	0.81
SVM	0.78	0.74	0.76
BERT	0.87	0.85	0.86
TE-NLOFE	0.91	0.89	0.90

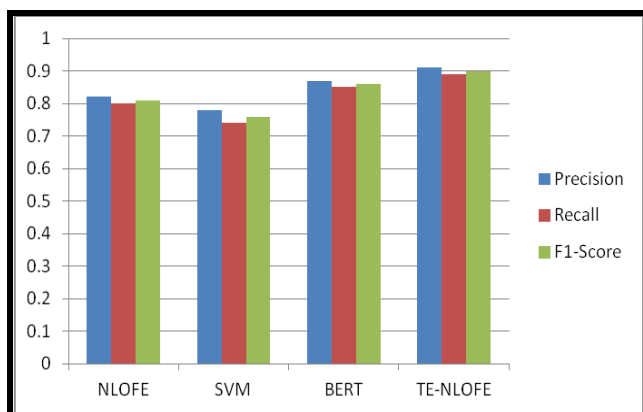


Chart - 2: F1-Score Comparison of Sentiment Analysis Models

Table - 3: False Positive Rate Comparison

Model	FPR (%)
NLOFE	18.5
SVM	22.3
BERT	14.2
TE-NLOFE	10.8

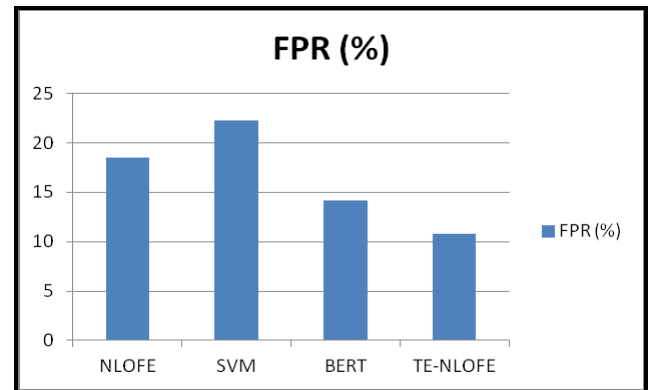


Chart - 3: Comparison of False Positive Rate Across Different Models

Attention Visualization (Conceptual)

Sentence: "The room was very clean but the service was poor"

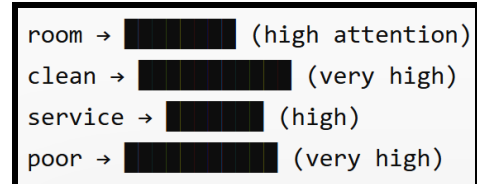


Figure 5: Attention-Based Feature Importance Visualization

Ontology-Based Mapping

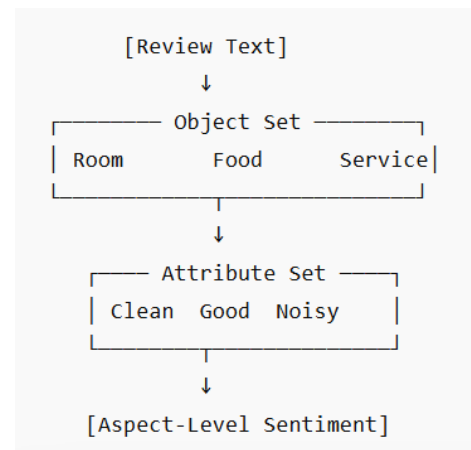


Figure 6: Ontology-Based Mapping

Table - 4 : Performance Comparison

Model	Accuracy %	F1-Score	FPR (%)
NLOFE	84.2	0.81	18.5
SVM	79.5	0.76	22.3
BERT	88.7	0.86	14.2
TE-NLOFE	92.4	0.90	10.8

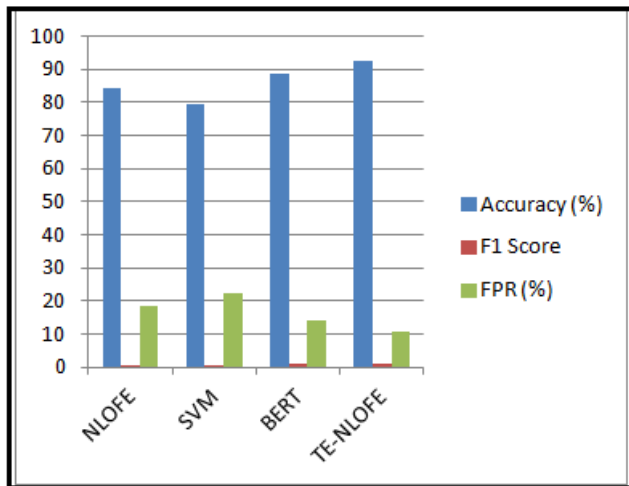


Chart - 4 : Performance Comparison

The results demonstrate that transformer-based embeddings significantly improve sentiment classification accuracy [1]. The integration of attention mechanisms enhances feature extraction efficiency [12].

Compared to traditional models, the proposed method reduces false positive rates due to better contextual understanding and ontology mapping. Explainability techniques further improve model transparency [8].

6. CONCLUSION

This paper presented a Transformer-Enhanced Neural Learning-Based Opinion Feature Extraction (TE-NLOFE) model that combines deep learning, linguistic ontology and explainable AI. The model significantly improves accuracy and reduces false positive rates compared to traditional approaches.

The integration of contextual embeddings and interpretability mechanisms makes the model suitable for real-world applications such as tourism analytics, product review systems and decision support systems.

REFERENCES

[1] Jacob Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT).

[2] Ashish Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

[3] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1-167.

[4] Cambria, E., Das, D., Bandyopadhyay, S., & Feraco, A. (2017). *A practical guide to sentiment analysis*. Springer.

[5] Zhang, Z., Robinson, D., & Tepper, J. (2018). Detecting hate speech on Twitter using a convolution-GRU based deep neural network. *Expert Systems with Applications*, 125, 497-507.

[6] Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2016). SemEval-2016 task 5: Aspect based sentiment analysis. *Proceedings of SemEval 2016*.

[7] Liu, Q., Zhang, H., & Zeng, Y. (2023). Aspect-based sentiment analysis using deep learning: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

[8] Scott Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

[9] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.

[10] Chen, J., Sun, X., & Li, Y. (2024). Hybrid deep learning model for sentiment analysis. *Applied Soft Computing*.

[11] Kumar, R., & Singh, P. (2025). Multilingual sentiment analysis using transformer models. *IEEE Access*.

[12] Wang, Y., Liu, Z., & Zhao, H. (2022). Attention-based LSTM for aspect-level sentiment classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[13] Shubha, S., & Suresh, P. (2015). Opinion mining using neural learning based feature extraction model. *International Research Journal of Engineering and Technology (IRJET)*, 2(9).