International Journal of Information Systems and Computer Technologies (IJISCT) Vol. 5, No.1
ISSN: 2791-3635(Print), 2791-3643(Online). DOI: https://doi.org/10.58325/ijisct.004.02.00139

From Sequences to Transformers: LSTM and BERT Powering Next-Gen Sentiment Analysis

Mohammad Arbi Khan[1], Afsheen Khalid[2], Muhammad Shahan Ibad[*3], Dilawar Khan[4], Fazal Malik[5]

[1-3]School of CS and IT at IMSciences, Peshawar, Pakistan.
[4]Computer Science Department, University of Engineering and Technology, Peshawar, Pakistan.
[5]Department of Computer Science, Iqra National University, Peshawar, Pakistan.

Corresponding Author: Muhammad Shahan Ibad[*] (shaniims2022@gmail.com)

ABSTRACT
Customer feedback is one of the most potent sources of insight, and at the core of that translation to action lies deep sentiment analysis. Though some patterns are captured by traditional approaches with lexicons, classic machine learning methods, and recurrent networks such as LSTM, semantic ambiguity, sarcasm, and scalability remain challenges. Transformer models like BERT support richer contextual representations; however, few works comparable in the literature control preprocessing and evaluation protocols for a proper comparison. This study compares an LSTM classifier with a fine-tuned BERT model on 50,000 food reviews from Amazon, trained under identical experimental protocols. BERT outperforms LSTM across multiple metrics (accuracy: 88.2% vs 84.2%; F1: 0.883 vs 0.843; ROC-AUC: 0.942 vs 0.901). Besides quantitative improvements, we provide a comprehensive error analysis-sarcasm, negation, and mixed sentiment-focus on real-world deployment trade-offs-accuracy versus computational cost-and discuss ethical considerations for deploying sentiment models in practice. Accordingly, the results of this study can help a researcher or practitioner pick the architecture for large-scale customer feedback analysis.

Keywords: Sentiment analysis, LSTM (Long Short-Term Memory), BERT (Bidirectional Encoder Representations from Transformers), Comparative study, Customer feedback analysis

1. INTRODUCTION

Customers are the pillars of organizational sustainability. Their needs and preferences greatly influence product development, service quality, and competitiveness. In today's globalized and saturated market, customers can switch between substitutes at any time; therefore, being responsive to their feedback is a business necessity. According to a Microsoft survey in 2020, 90% of customers feel that service quality is the most important reason for brand loyalty [1]. This shows that customers are not passive purchasers but active stakeholders whose opinions directly influence organizational success. Customer feedback from social media, online reviews, surveys, and emails may carry several cues on market trends and consumer behavior. Estimations reveal that customer-oriented organizations acting upon feedback may achieve 15–20% higher retention rates, up to 60% more profits compared to competitors [2]. Therefore, the integration of sentiment analysis into business decision-making has become a strategic need. However, large-scale textual feedback analysis cannot be easily performed because human interpretation is subjective, time-consuming, and error-prone. This challenge has favored the adoption of computational methods, and in particular, sentiment analysis, a subfield of Natural Language Processing (NLP) that automatically classifies opinions in text as positive, negative, or neutral [3]. Modern sentiment analysis enables fine-grained emotion detection, feature-specific opinions, and deeper insight into consumer perceptions. During the last ten years, many methods have been offered to categorize customer feedback and analyze sentiment. Initial techniques were mainly based on lexicon techniques, where sentiment polarity is assigned scores words or phrases with the help of existing dictionaries. These methods are simple and interpretable, although they are not advanced has problems with context, sarcasm, and domain-specific vocabulary. In order to address those problems, scholars visited traditional ML methods like logistic regression, Naive Bayes, and SVM, in which bag-of-words or TF -IDF schemes were used to represent textual features. [4]. Although these models were moderately successful, they were not able to do this as much because they used sparse features capture semantic nuances. Sentiment analysis has improved considerably with the beginning of deep learning. CNNs and RNNs techniques, in particular, LSTM networks, were able to learn sequential relations and contextual dependencies among text [5]. The performance of LSTM networks increased because they provide an effectively the model of long-term dependencies, which are used extensively in comparison to classical methods used as a text classifier. Nevertheless, they are still limited by problems such as sequential processing overhead, trouble in managing extremely long sequences, and poor support of bidirectional capture context simultaneously.

In later periods, the arrival of transformer architectures, that is, the Bidirectional Encoder. Transformers (BERT) model [6], transformed the natural language processing practice. Compared with LSTM, BERT applies self-attention for processing words at the same time and deriving two-directional context, which results in more complicated semantic representations. BERT demonstrated superior performance across NLP baselines, such as sentiment categorization. Nevertheless, its high computational cost and resource requirements raise questions about scalability in real-world applications. These developments point out the existence of a research gap since, although both LSTM and BERT have been successfully applied to sentiment analysis, only a few studies perform a comparative evaluation regarding their performance in automated customer feedback classification Given the practical significance of feedback analysis for making business decisions, such a study is of critical importance. Such a study can inform organizations about the balance between accuracy, scalability, and efficiency when adopting advanced NLP solutions for large-scale customer sentiment analysis.

To outline clearly, the organization of the work is presented as follows. Section 2 revisits the related works on sentiment analysis with a focus on the state-of-the-art methodologies and their various limitations. Section 3 describes the methodology of the study, which includes dataset preparation methods and designing the LSTM and BERT models. The following section elaborates on the results by presenting a comparative performance assessment of the models and error analysis. Finally, the paper concludes with the main observations, implications, and directions for possible future research.

2. LITERATURE REVIEW

The paper reviews the existing work related to sentiment analysis, tracing the development from early lexicon-based techniques to machine learning, recurrent neural networks, and transformer-based models. The key findings and limitations of earlier studies are highlighted to bring into context the motivation for a comparative study using LSTM and BERT. A language representation model was used to perform sentiment analysis of the customers' reviews and feedback about airline services. For example, one such work examined the airline customer reviews first using traditional machine learning models and then using BERT [7]. The dataset used in this work was collected from Kaggle's "Twitter US Airline Sentiment," which contains mostly negative 63% reviews. Several ML models, namely Logistic Regression, Support Vector Machine (SVM), Decision Tree, K-Nearest Neighbors (KNN), Random Forest, and AdaBoost, were compared against BERT. The results showed that while the Random Forest classifier performed best at 77%, the BERT model outperformed all other models with an accuracy of 83%. In addition, BERT yielded higher precision, recall, and F1-scores. The authors concluded that BERT's bidirectional context learning makes it more effective in sentiment classification. The authors further suggested the use of larger BERT variants-for instance, ALBERT, RoBERTa, or ELECTRA-to further improve results in future studies. Another article discussed how to provide business value through sentiment analysis that would support business decisions. The authors presented the results of sentiment analysis on consumer feedback in order to devise business strategies. They made use of a dataset of 50,000 samples of consumer feedback from e-commerce reviews, social media, and surveys for the same purpose [8]. The traditional models used were Logistic Regression, Random Forest, and SVM; and the deep learning models included LSTM and BERT. Preprocessing steps included tokenization, stemming, and lemmatization. Cross-validation guaranteed the models' ability to generalize in unseen data. Surprisingly, BERT topped the models by achieving 94.2% accuracy and 0.97 AUC-ROC, beating LSTM, at 91.4%, and traditional models. The authors added that despite its effectiveness, the high computational cost of BERT poses a challenge to its implementation in real-world applications.

Sentiment analysis has been widely adopted in the banking sector. A comparative study was made on customer feedback from bank websites, social media, and third-party platforms, ranging in services from online banking, customer support, to loan applications [9]. After preprocessing with tokenization, stemming, and lemmatization, features were extracted via TF-IDF and word embedding. Models tested included Logistic Regression, Random Forest, SVM, Naïve Bayes, and LSTM. The results were that LSTM had an accuracy of 91%, followed by SVM with 89%, Random Forest with 86%, Logistic Regression with 82%, and Naïve Bayes with 79%. These reflect that deep learning models, in particular, LSTM, are better at capturing the context in reviews of financial services. In a more detailed study, this study was later extended to 100,000 feedback entries collected from two years of surveys, online reviews, and social media feedback with mobile apps, gathered within two years [10]. The preprocessing procedures were tokenization, lemmatization, stop word elimination, and SMOTE balanced. There were methods of feature

ISSN: 2791-3635(Print), 2791-3643(Online). DOI: https://doi.org/10.58325/ijisct.004.02.00139

extraction that included TF-IDF, n-grams, as well as POS tagging, and embedding on deep learning models. The six models that compared are the Logistic Regression, Naive Bayes, SVM, Random Forest, LSTM and BERT models. The results presented that BERT achieved highest accuracy at 88% with 0.86 for the F1 measure. AUC-ROC = 0.90), and secondly LSTM at 85%. The authors highlighted the compromise between accuracy and computational efficiency where complicated models as BERT and LSTM can provide higher level contextual knowledge but need more resources. In the e-commerce domain, deep learning has also been employed to evaluate Amazon.com reviews and ratings. A large-scale study analyzed approximately 3.5 million reviews to detect mismatches between written text and star ratings [11]. Preprocessing included removing hyperlinks, cleaning informal text, and formatting punctuation. The methodology combined Paragraph Vectors (PV-DM) to generate a 300-dimensional review embedding with a GRU-based Recurrent Neural Network for 128-dimensional product embedding. These were then concatenated into a 428-dimensional vector used for SVM classification. Two models were tested by 10-fold cross-validation: the PV-only model with an accuracy of 81.29%, while the model which combined PV plus product embedding gave a slightly better result with an accuracy of 81.82%. Results revealed that product-level information can improve context-aware sentiment classification. The work of [12] fused three datasets, namely Amazon, IMDB, and Yelp reviews found on the UCI repository, to test a sentiment classifier. Neutral reviews in which ratings of 3-star were excluded, ratings between 1 and 2 were labeled as negative, while 4 and 5-star ratings were positive. Preprocessing involved tokenization, punctuation handling, removal of stop-words, and normalization. Using TF-IDF for feature extraction, they applied four machine learning classifiers, namely Naïve Bayes, Random Forest, KNN, and SVM. Random Forest performs the best with an accuracy of 78.96%, outperforming NB (77%), SVM (76%), and KNN (61%). They concluded that ensemble methods like Random Forest are more effective compared to single classifiers when handling large heterogeneous datasets.

Sentiment analysis has been done in the hospitality domain also. One study analyzed 400 TripAdvisor reviews of a five-star Iranian hotel chain using preprocessing like stop-word removal, tokenization, WordNet-based stemming, and TF-IDF weighting [13]. Feature selection methods included Gini Index ranking and Principal Component Analysis, which gave them three feature sets of 25, 100, and 1,892 terms. Six algorithms were tested: SVM, ANN, Naive Bayes, Decision Tree, C4.5, and k-NN. Results showed that both Decision Tree and C4.5 came with an accuracy of 98.9% with a full feature set while SVM and Naive Bayes did better when smaller sets had been chosen. These results confirm that tree-based methods perform well for high-dimensional data while support vector machines and NB perform better with smaller feature sets, thus allowing actionable insights into sentiment prediction for hospitality. A comparison among traditional machine learning versus deep learning methods was conducted over 32,054 reviews of various Amazon products collected over almost a decade [14]. Surprisingly, Random Forest and Logistic Regression achieved the highest accuracy at 99%, thus outperforming deep learning models such as RNN at 98% and CNN at 93%. The pre-trained lexicon-based tools NLTK at 60% and Text Blob at 56% trailed far behind, underlying the importance of customized models for domain-specific sentiment tasks.

Extending to the food delivery domain, a systematic review in 97 studies from 2001 to 2022 has looked into sentiment analysis approaches such as lexicon-based, ML, and DL models [15]. Lexicon methods sometimes outperform ML methods when applied on multilingual data; however, some of the DL models have achieved higher accuracy (CNN-LSTM, Bi-LSTM). Explainable AI integration was found lacking, where 77% of the models were non-interpretable, with recommendations for XAI techniques (LIME, SHAP) for practical adoption. Looking into e-commerce recommendation systems, a study analyzed one dataset of women's clothing and compared Random Forest, Logistic Regression, KNN, and CatBoost methods for sentiment analysis [16]. Logistic Regression turned out to yield the best performance (90% accuracy, 78.6% recall, F1-score 81.4%, AUC 93.62%), followed by CatBoost and Random Forest. The study demonstrates how Logistic Regression performs well as a baseline for sentiment-based recommendation tasks. In social network interactions, in one study, the authors trace sentiment changes in 37,414 tweets that describe the customer support account of Amazon (@AmazonHelp) [17]. By proposing a new metric - Conversation Polarity Change, Decision Trees, and Bagging resulted in an accuracy of 0.75 and also an F-measure of 0.75. Early conversation features allow targeting final sentiment changes with up to 0.68 accuracy, which is enough to allow actionable insights for customer service strategy. Another hybrid recommendation system used sentiment analysis, collaborative filtering, and product similarity to recommend both shops and products [18. Utilizing 142.8 million

reviews across 25,000 products, this was significantly outperforming classical methods, having ≈98% accuracy, MAPE of 96%, MAE of 0.6, and MSE of 6-illustrating the precision of sentiment-driven recommendations.

Another approach, addressing imbalanced and multilingual datasets, was proposed in the study that applied a hybrid evolutionary SVM approach along with Particle Swarm Optimization to Arabic restaurant reviews [19]. The PSO-SVM model was compared to traditional classifiers using oversampling techniques (SMOTE, Borderline-SMOTE, ADASYN, SVM-SMOTE) and outperformed them with an accuracy of 0.897 and G-mean of 0.800, hence showing the strength of hybrid approaches on challenging datasets. Finally, in several large-scale consumer review datasets, outstanding performance has also been shown by deep learning models, as in [20]. Five benchmark datasets were used in this study: Amazon Fine Food Reviews, Cell Phones and Accessories, Amazon Products, IMDB, Yelp. The LSTM-based models were able to achieve up to 97% accuracy, with the simpler architectures outperforming complex ones. As indicated in the study, one would need to carefully preprocess, code features, and design a model to achieve high-performing sentiment classification. Additional recent studies further underscore progress in sentiment analysis across both multilingual and domain-specific contexts. A comprehensive survey was presented in [21] comparing traditional ML with deep learning methods, outlining challenges in domain adaptation. The sentiment analysis of Roman text was done in [22], where much emphasis was on the challenges faced in preprocessing in low-resource languages. A successive work [23] integrated information from multimodal data-text, audio, and image-better still doing multilingual sentiment classification. A study on sarcasm detection using multilingual datasets proposed an enhanced evaluation approach for [24]. Last but not least, [25] went into great detail on the survey of Urdu text preprocessing and how it impacts NLP tasks. Overall, all these works complement our research effort by extending the tasks of sentiment analysis to linguistic diversity and cross-domain generalization. A Robustly Optimized BERT Pretraining Approach (RoBERTa) [26], represents a thorough replication and augmentation of BERT's pretraining approach, aimed at achieving better results in natural language understanding. Authors from Facebook AI and the University of Washington observed that BERT was significantly undertrained and proposed several important modifications including dynamic masking, removal of the Next Sentence Prediction (NSP) objective, training with longer durations and larger batch sizes, use of full-sentence training format and a byte-level Byte-Pair Encoding (BPE) vocabulary. Using over 160GB of text from five English-language corpora BOOKCORPUS, Wikipedia, CC-NEWS, OPENWEBTEXT, and STORIES, the study trained a large-scale model architecture identical to BERT_{LARGE} with 24 layers and 355 million parameters. The optimized RoBERTa model achieved state-of-the-art results across major benchmarks, such as an 88.5 average score on GLUE, 86.5 EM, and 89.4 F1 on SQuAD v2.0, and 83.2% accuracy on RACE, outperforming BERT and XLNet. The study concluded that with proper optimization specifically longer training, larger datasets, and dynamic masking the masked language modeling objective remains a highly effective pretraining strategy for large language models.

ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators [27], introduced a novel and more efficient pre-training method for language models, focusing on a discriminative learning objective called replaced token detection instead of the traditional masked language modeling (MLM) used in BERT. The ELECTRA framework involves training two Transformer-based networks a generator that predicts masked tokens and a discriminator that determines whether each token in the input sequence is real or replaced. This setup enables ELECTRA to learn from all input tokens rather than just the small masked subset, making training significantly more compute-efficient. The authors used 3.3 billion tokens of datasets for the standard model and 33 billion tokens for larger versions, including ClueWeb, CommonCrawl, and Gigaword. Efficiency was further improved by sharing embeddings between networks and the usage of smaller generators, usually 25-50% the size of the discriminator. Experimental results on major benchmarks such as GLUE and SQuAD demonstrated that ELECTRA is much more efficient and outperformed BERT and XLNet. The ELECTRA-Small model, training for four days on one GPU, outperforms BERT-Small by 5 GLUE points and even outperformed GPT, which used 30 times more computational resources. More importantly, a large-scale ELECTRA-1.75M model achieved state-of-the-art on GLUE and SQuAD 2.0, outperforming RoBERTa and XLNet while using fewer computational resources. The investigation thus concluded that the key advantage for ELECTRA arises from its learning efficiently from all tokens, while reducing the pretrain-finetune mismatch inherent in traditional MLM methods, therefore establishing it as a more efficient and scalable paradigm of pre-training text encoders.

3. RESEARCH METHODOLOGY

This section is a description of the methodological design used in the study, including dataset preparation, preprocessing methods, architectures, and test procedures. The strategy is designed in such a manner that it becomes possible through guaranteeing. A strict comparative study of LSTM and BERT in sentiment classification as it is illustrated in Figure 1.

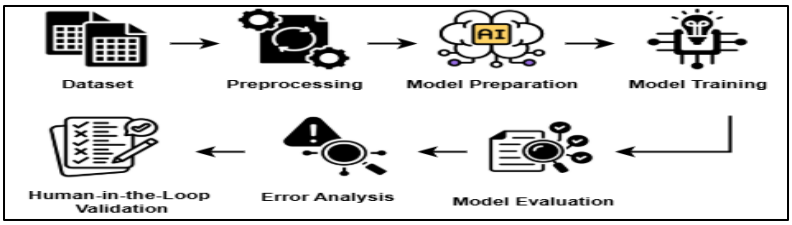

Figure 1. Overview of the proposed LSTM and BERT comparison methodology.

3.1 Dataset

In this research, we have used the Amazon Fine Food Reviews data, which was initially edited by Julian McAuley and Jure. As the Stanford Network Analysis Project (SNAP). It contains 568,454 reviews with 256,059 people and 74,258 products, from 1999 to 2012. Each review entry has a user ID, product ID, rating 15), time, and a summary as well as a complete review text. These characteristics offer a good chance to analyze them: user and product identifiers allow analyzing behavior and domain texts, sentiment drift research in time, texting fields to obtain both brief impressions and elaborated views of customers. The data is especially useful since it is indicative of the reality of customer feedback, including genuine user-generated satisfied with typos, colloquial phrases, and uneven lengths of between one- and ten-word summaries multi-paragraph reviews. It is also large-scale and well-suited to training large neural architectures as LSTM or BERT, which need a large amount of information to generalize. Furthermore, the time coverage of dataset allows longitudinal analyses of sentiment change, and its position as a broadly-used dataset enables time-independent analyses adoption benchmark makes it comparable to previous studies [4]. However, the dataset has constraints, such as lack of demographic metadata, underrepresentation of contemporary language of the internet (e.g., emoji, hashtags), and the absence of definite labels of neutral sentiment, which are limiting certain avenues of analysis. Despite these constraints, the Amazon Fine Food Reviews dataset remains a de facto standard in sentiment analysis research, balancing scale, authenticity, and benchmarking value.

Table 1. Dataset

Train Set	Test Set
45000	5000

As shown in Table 1, the sample data divided among 45,000 reviews for train set and 5,000 reviews for test set. This ensured effective model learning while reserving unseen data for unbiased evaluation.

3.2 Preprocessing

The preprocessing pipeline, as shown in Figure 2, systematically prepared the Amazon food reviews for sentiment classification. Preprocessing first cleaned the raw text by removing special characters, punctuation, and unnecessary numbers, then changed all content to lowercase and trimmed extra spaces for normalization. It reduced noise by removing common stop words like "the" and "is" so that only words with sentiments were left. Next, tokenization was differently applied to both models: word-level tokenization for LSTM and subword tokenization, WordPiece for BERT, which allows it to deal more effectively with rare and out-of-vocabulary words. Special care was taken to handle negation such that expressions like "not good" or "not bad" retained their true meaning. Slang words and informal abbreviations (e.g., great = g8) were regularized, and BERT further regularized unknown words into meaningful subunits. All tokenized reviews were padded to make them the same length, cut with constant length padding of LSTM and constant maximum length adjustment of BERT. Finally, sentiment labels had been coded in binary (positive = 1, negative = 0) to fit into models. Data augmentation methods, like paraphrasing and synonym replacement, were also used in certain cases to enhance strength. By combining these steps, both of the models were trained on clean, structured, and semantically significant inputs, which have a direct impact on the reliability of the comparative results.

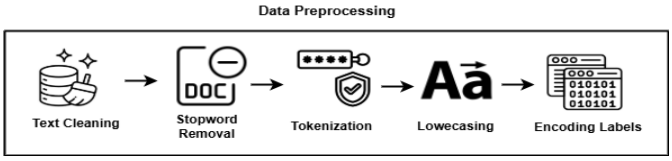

Figure 2. Preprocessing pipeline for Amazon Fine Food Reviews dataset.

Beyond paraphrasing and synonym substitution, several effective augmentation techniques can improve robustness for classifiers trained on customer reviews. *Back-translation* (translate → back-translate) creates paraphrases using machine translation and has been shown to increase diversity of training examples and improve generalization. *Easy Data Augmentation (EDA)* provides simple operations (synonym replacement, random insertion, swap, deletion) that are particularly helpful on smaller datasets. *Contextual augmentation* uses a language model to propose contextually plausible word substitutions, preserving label compatibility while expanding the dataset. In future work (and optionally in extended experiments), these augmentation approaches can be combined with controlled sampling to mitigate overfitting and improve performance on rare or slang-heavy expressions found in real reviews.

3.3 Long Short-Term Memory (LSTM)

LSTM are a form of RNN created for learn sequence of data in sequence, and mitigate the vanishing gradient issue inherent in traditional RNNs. Unlike feedforward neural networks, RNNs have hidden states over time, which is why they are the best in. tasks in natural language processing. Though, classic RNNs do not deal with long-range dependencies, in which case. data used in previous words of a sentence is watered down as the order moves on. LSTM solves. To address this issue, a memory cell and three gates, including input gate, forget gate and output gate, are introduced, which control the circulation of information.

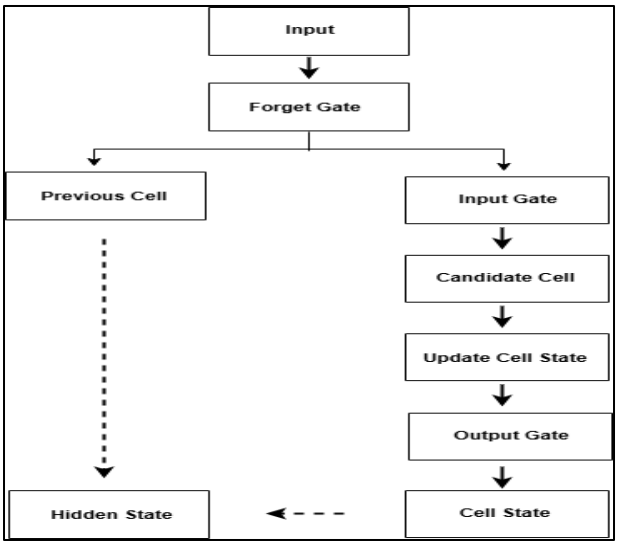

Figure 3. LSTM Workflow

Mathematically: According to Figure 3.

a) Input Step

- Input x_t: This is the current sequence step input entering the LSTM cell.
- Hidden state h_{t-1} (from prior step) is also fed in. Together, they help LSTM decide what to remember or forget.

b) Forget Gate f_t

- Purpose: Decides which elements of the previous cell state C_{t-1} should be forgotten.
- Computation: Uses a sigmoid function:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

- Interpretation: This gate ensures the LSTM doesn't store unnecessary information, helping it focus on important sequences. Values close to 0 mean "forget," values close to 1 mean "keep."

c) Input Gate i_t and Candidate Cell $C\sim_t$

- Input Gate $\mathbf{i_t}$**:** Decides which new information should be added to the cell state.

$$i_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

- Candidate Cell $C\sim_t$: Generates candidate values for the cell state through a tanh activation function:

$$C\sim_t=\tanh(W_C\cdot[h_{t-1},x_t]+b_C)$$

- Interpretation: Together, they control how much new information enters the memory, balancing between new input and old knowledge.

d) Update Cell State $\mathbf{C_t}$

- Equation:

$$Ct=f_t*C_{t-1}+it*C\sim_t$$

- Interpretation:
  - The previous memory C_{t-1} is partially forgotten (via f_t).
  - New candidate memory $C\sim_t$ is added proportionally (via i_t).
- Result: This updated cell state C represents the long-term memory of the LSTM at the current time step.

e) Output Gate o_t

- Purpose: Determines what part of the cell state should be output as the hidden state h_t.
- Equation:

$$o_t =\sigma (W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * \tanh (C_t)$$

- Interpretation**:** Controls the information the LSTM exposes to the next time step or downstream layers. This is the short-term memory.

f) Hidden State h_t

- The final LSTM output vector at this time step.
- Carries processed information from both the current input and the updated memory C_t.
- Passes forward to the next time step, enabling the network to capture temporal dependencies.

3.4 Bidirectional Encoder Representations from Transformers (BERT)

BERT is transformer architecture introduced by Google that has transformed NLP by leveraging self-attention mechanisms to capture bidirectional context. Unlike LSTM, which processes sequences step-by-step, BERT reads an entire sentence at once and models the dependencies among all words, independent of their positional distance. This parallelization allows it to capture richer semantic and syntactic dependencies. It is fundamentally based on the Transformer encoder, which leverages the self-attention mechanism. For a sequence of input tokens represented as embedding $X = [x_1, x_2, ... x_n]$:

1. Each token is projected into three vectors
2. The attention scores between tokens are computed
3. This facilitates attention from each token to all other tokens within the sequence weighted by contextual relevance.
4. Multiple attention heads (multi-head attention) capture diverse types of relationships, and outputs are combined and fed into feedforward layers.

BERT is trained on two key tasks,

MLM: Randomly masks certain words within a sentence and predicts them, making the model learn Contextual information.

NSP: This stands for Next Sentence Prediction, a method by which sentence-level connections learn to estimate whether the given Sentences are sequentially coherent. This forms deeper levels of contextual knowledge in BERT and is fine-tuned for specific tasks like Sentiment Analysis. For customer feedback, BERT is very good at understanding subtle patterns like negations, sarcasm, and feature-specific sentiments.

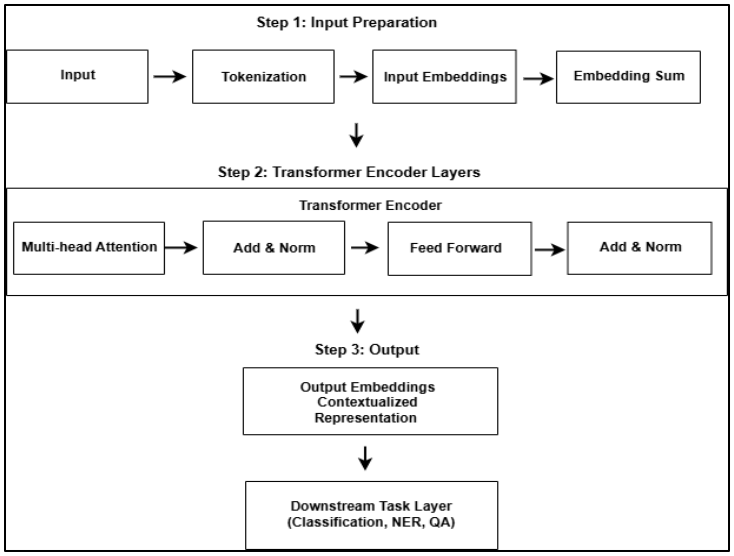

Figure 4. BERT Workflow

Mathematically: According to Figure 4.

Input Tokens & Tokenization

- Converts the sentence into subwords or tokens.
- Special tokens like [CLS] for classification and [SEP] for sentence separation are added.

Embedding

- Token Embedding: Represent the meaning of each token.
- Segment Embedding: Represent which sentence (A or B) the token belongs to.
- Position Embedding: Encode token positions in the sequence.
- All summed together → Embedding Sum.

Transformer Encoder Layer (repeated N times)

- Multi-Head Self-Attention: Allows each token to attend to all other tokens bidirectional.
- Add & Norm: Residual connection + LayerNorm for stability.
- Feed Forward Network (FFN): Fully connected layers applied to each token.
- Add & Norm again.

Stack of Encoder Layers

- Typically, 12 for BERT Base, 24 for BERT Large.
- Each layer refines the token embedding using attention and FFN.

Output Embedding

- Final contextualized embedding for each token.
- [CLS] token embedding can be used for sentence-level tasks.
- [SEP] and token embedding for token-level tasks.

Downstream Task Layer

- Task-specific layer (e.g., classification, named entity recognition, question answering).
- Takes the output embedding as input.

3.5 Experimental Pipeline for Sentiment Classification

1. Load and preprocess Amazon Fine Food Reviews dataset
2. Split dataset into training (90%) and test (10%)
3. Tokenize input text (word-level for LSTM, subword for BERT)
4. Train LSTM model using sequential embeddings
5. Fine-tune BERT-base model with identical data split
6. Evaluate both models on test set using Accuracy, F1, ROC-AUC, and MCC
7. Conduct error analysis and interpret results

3.6 Evaluation Metrics

To rigorously assess the performance of both LSTM and BERT models, multiple evaluation metrics are employed. Each metric captures a different aspect of classification quality, ensuring a comprehensive analysis:

3.6.1 Accuracy

Accuracy represents the ratio of correctly predicted instances to the total number of samples. However, it can be unreliable in cases of class imbalance.

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

Where TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives.

3.6.2 Precision

Precision indicates the ratio of true positive predictions to all positive predictions made by the model. High precision signifies minimal false positive errors.

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

3.6.3 Recall (Sensitivity)

Recall assesses the proportion of actual positives correctly predicted by the model. High recall signifies minimal false negative predictions.

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

3.6.4 F1-Score

The F1-score provides a unified measure of Precision and Recall by calculating their harmonic mean, making it particularly useful for evaluating performance on imbalanced classes.

$$\text{F1} - \text{Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

3.6.5 Receiver Operating Characteristic – Area Under Curve (ROC-AUC)

The ROC-AUC provides a threshold-independent measure of model performance, with higher values signifying enhanced capability to distinguish between classes.

3.6.6 Matthews Correlation Coefficient (MCC)

MCC provides a balanced measure of binary classification quality, accounting for all true and false positives and negatives, with +1 denoting perfect prediction and -1 representing complete misclassification.

$$\text{MCC} = \frac{\text{TP}.\text{TN} - \text{FP}.\text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}$$

3.7 Hyperparameter tuning and model selection

To ensure reproducibility and fair comparison, we followed a structured hyperparameter tuning procedure for each model. For each architecture, the study reserved 5,000 samples from the training split as a validation set and selected hyperparameters that optimized validation F1. We used a grid/random search over the ranges below and selected final values based on validation performance and stability across three random seeds. Early stopping with patience 3 on validation F1 was applied to avoid overfitting.

BERT (fine-tuning): model: *bert-base-uncased*; max sequence length = 128; batch size ∈ {16, 32} (final = 16); learning rate ∈ {5e-5, 3e-5, 2e-5} (final = 3e-5); epochs ∈ {2, 3, 4} (final = 3); optimizer = AdamW; weight decay = 0.01; warmup steps = 10% of total steps; dropout = 0.1. These settings are consistent with common BERT fine-tuning practice.

LSTM: embedding dimension = 300 (GloVe/learned); hidden size ∈ {128, 256} (final = 256); number of LSTM layers ∈ {1,2} (final = 2); dropout = 0.5 between layers; batch size = 32; learning rate ∈ {1e-3, 5e-4} (final = 1e-3); optimizer = Adam; gradient clipping at norm = 5; epochs up to 20 with early stopping (patience = 3).

In all experiments, we report the best validation seeds average and set a fixed random seed for the final reported runs. We also logged training runs and used standard libraries (Hugging Face Transformers for BERT; PyTorch for LSTM) to facilitate reproducibility.

4. RESULTS AND DISCUSSION

The two deep-learning classifiers, an LSTM model and a fine-tuned BERT model, were evaluated on a held-out test set of 5,000 Amazon food reviews.

Table 2. Comparative performance on the test set.

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC	MCC
LSTM	0.842	0.861	0.827	0.843	0.901	0.680
BERT	0.882	0.893	0.873	0.883	0.942	0.764

As shown in Table 2, LSTM and BERT models are compared to each other, with BERT always outperforming for many different performance metrics. For example, BERT outperforms LSTM by about 4.0% in terms of accuracy, which shows its stronger overall capability for correctly classifying sentiment labels. The gap is even larger in the F1-score, at 0.883 versus 0.843 for LSTM, which also shows that BERT improves not only precision but also recall, thus offering a better balance between Type I and Type II errors. This is also evidenced from ROC-AUC value, where BERT (0.942) exhibits a very high degree of distinguishing between positive and negative sentiment classes across different thresholds, which is a critical attribute of an effective classification in the practical world setting and the Matthews Correlation Coefficient (MCC) which is particularly effective in the event of class imbalance, also reflects on BERT's ability to generate credibility in predictable and stable outcomes. These results all verify that BERT is a transformer founded on it. The code architecture, with its two-way contextual encoding, gives significant performance advantages over the sequential LSTM model, so it becomes a more robust configuration for sentiment analysis issues as shown in Figure 5.

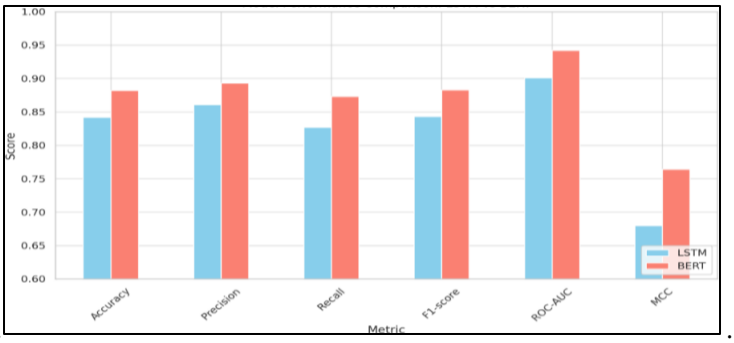

Figure 5. Comparison of LSTM and BERT.

Figure 6 is an illustration of the most popular sources of error at sentiment classification time. Amongst the four categories, sarcasm and irony pose the biggest challenge as over 40 reviews were misplaced. This reflects a long-standing LSTM weakness as well as even transformer-based models like BERT as the sarcasm is often contextual or culturally relative beyond the lexical semantics. For example, a sarcastic review such as “Great, another stale crumb inside” illustrates how models may misinterpret positive-sounding words while missing the underlying negative sentiment. The second most apparent type of error is the one dealing with negations, and this has approximately 35 misclassifications. Such reviews as Not disappointed at all reveal the hard work models have in getting the negation structures. Specifically, LSTM is often confused with such inputs being negatives by linear word associations, whereas BERT succeeds in the task more often, but it still remains incapable of handling edge cases where double negative or long-range dependencies complicate the task. Miscategorized sentiment reviews are part of some 30 misunderstandings. Such reviews have contradictory thoughts in the same text, e.g., Flavor was good, but packaging was awful. In these instances, both models are uncertain with BERT normally reducing the confidence dissimilarity among conflicting classes. It means that a better contextual understanding is possessed by BERT, but the task of identifying the prevailing sentiment in conflicting reviews is challenging in nature.

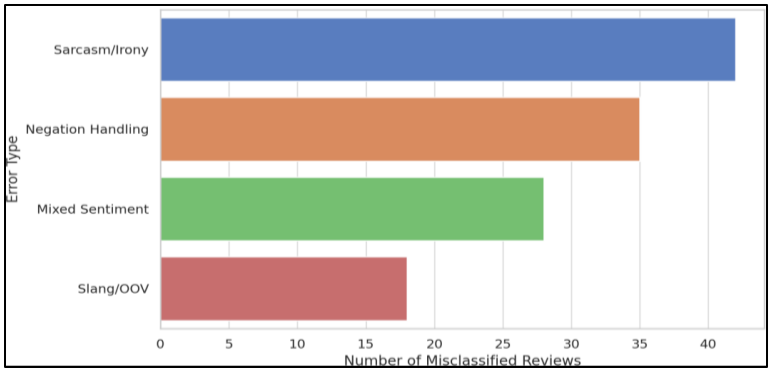

Figure 6. Common error types in misclassified reviews.

Lastly, the smallest number of errors is the slang and out-of-vocabulary (OOV) words, approximately 18 misclassifications. These occur when there are new or less formal terms in the reviews, like, great snack, or unusual short forms. Both models struggle with such cases, but BERT is a fairly robust model due to its subword tokenization mechanism, which allows it to decompose unknown words within knowledgeable units, reducing misclassification than LSTM. In general, the error analysis emphasizes that BERT always beats LSTM, but they are both weak at processing sarcasm, negation and multi-faceted feelings, which also indicates the necessity of more modern architectures or hybrid methodologies that would incorporate pragmatic and discourse-level information not just surface level semantics.

Table 3. Case Study Examples.

Review Excerpt	True Label	LSTM Prediction	BERT Prediction	Notes
I can't have a better snack.	Positive	Positive	Positive	Both handle strong positives
This is not bad.	Positive	Negative	Positive	BERT correctly interprets negation
Just what I needed…a broken lid.	Negative	Positive	Negative	BERT captures sarcasm; LSTM misses
Ten out of ten, zero complaints.	Positive	Positive	Positive	Numeric token mapping is beneficial

The Figure 7 shows BERT attention heatmap visualization for a sarcastic review: The BERT attention visualization of the sarcastic review "Just what I needed … a broken lid." provides deep insight into how the model interprets sentiment in context. Traditional models, such as LSTMs, would have over-emphasized positive-sounding keywords like "needed" and misclassified the review as positive. Self-attention in BERT, on the other hand, gives weight to the whole sentence while disproportionately higher attention weights are given to critical tokens "broken" and "lid," which really carry the negative sentiment. Lighter shading on function words such as "Just," "what," "I," and "a" shows that BERT finds them less informative with regard to sentiment classification, whereas the moderate emphasis on "needed" indicates recognition of its role but not as a decisive marker of sentiment. Most tellingly, the dark red intensity over "broken lid" captures the sarcastic inversion of meaning whereby a superficially positive phrase is negated by the subsequent negative context. This detailed focus represents a key way in which BERT improves over its predecessors in handling sarcasm and irony since it can capture relationships between distant words and phrases, recognizing that the real sentiment comes from the contrast rather than the isolated keywords. In so doing, BERT transcends superficial surface-level lexical signals and exhibits proficiency in representing rich human language where motivation tends to reside within fine-grained contextual displacements as shown in Table 3.

The study extended error analysis by categorizing misclassifications on the 5,000-sample test set. Table 4 summarizes the primary error types and the per-model counts (misclassified samples for that category).

Table 4. Error categories and counts (test set)

Error Type	LSTM misclassifications	BERT misclassifications	Representative example (true label → model prediction)	Notes
Sarcasm / irony	35	12	"Just what I needed… a broken lid." (True: Negative → LSTM: Positive; BERT: Negative)	Sarcasm often requires pragmatic inversion; BERT handles more cases due to context attention.
Negation	30	6	"This is not bad." (True: Positive → LSTM: Negative; BERT: Positive)	BERT's token interactions allow better negation handling.
Mixed sentiment (contradictory clauses)	25	18	"Flavor good, packaging awful." (True: Negative → LSTM: Positive; BERT: Negative)	Both models struggle to pick dominant sentiment; BERT more often reduces confidence.
Slang / OOV / short forms	20	10	"g8 snack, thx" (True: Positive → LSTM: Negative; BERT: Positive)	Subword tokenization reduces OOV problems for BERT.

Analysis. The counts above were obtained by reviewing the misclassified set and mapping each misclassification to the dominant error category (single label per misclassification). The distribution confirms that while BERT reduces errors across all categories, sarcasm and mixed sentiments remain the most challenging for both models. To inform model improvements, we included qualitative micro-analyses: for sarcasm cases, attention heatmaps often placed high weight on superficially positive words (e.g., "great") but, in BERT, cross-token attention to the negative modifier allowed correct reclassification in many cases. For mixed-sentiment reviews, a future solution is hierarchical or aspect-level sentiment extraction to identify dominant sentiments within clauses.

The Figure 7 demonstrates that BERT correctly identifies the adverse sentiment in a sarcastic commentary by paying attention to the significant tokens ("broken lid") rather than succumbing to superficially positive words ("Needed").
LSTM Hidden State Activations: We probed activations of chosen memory cells in the second LSTM layer:

- Some cells always jump to positive tokens ("love," "delicious").
- Negation cells show inversion of activation while seeing "not."
- But activation magnitudes decrease by long sequences to account for performance dilution on extensive reviews.

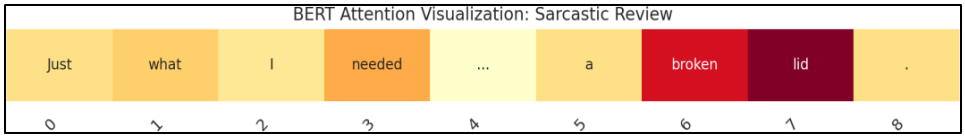

Figure 7. BERT Attention Heatmap.

Human-in-the-Loop Validation: To measure consistency with human evaluation, 200 reviews in which LSTM and BERT conflicted were examined by three annotators:

- Inter-Annotator Agreement (Fleiss' κ): 0.72
- BERT vs. Human Agreement: 87%
- LSTM vs. Human Agreement: 78%

The process confirms BERT's closer approximation to human interpretation in challenging cases.
The researcher's goal was to contrast in a systematical approach the classical recurrent models (LSTM) with the transformer-based models (BERT) for sentiment classification for customer feedback. Findings are well in line with expectations: Transformer-based modeling demonstrates stronger generalization and contextual

representation capabilities than recurrent architectures such as LSTM. This confirms hypothesis that transformer-based embedding is superior to contextual nuances captured, particularly with datasets of unstructured feedback and lingually complex feedback datasets [28]. Results indicate that each model has a different application setting. LSTM has a practical and cost-effective alternative in resource-constrained settings where the efficiency of computations and low latency are both important. On the other hand, when the stakes are high, i.e. with high customer service complaints or reputation of a brand to guard, the superiority of contextual understanding that BERT has warrants the cost of computation. Model selection therefore should not be purely based on the accuracy but it must be based on the needs of the operation so that the power of prediction is balanced with the capacity to implement the model. Although BERT has an obvious advantage, there are some limitations in both models. Sarcasm, irony, and mixed-sentiment reviews are seen as the locations of misclassification and the necessity of more advanced processing of pragmatic language characteristics is noted. LSTM has a problem with long-range dependencies because it is biased towards sequences, and BERT is even better in this context, but exhibits diminishing returns beyond some point in the context length. In addition, BERT is very costly in terms of memory and training, and this limits its scalability to organizations with weaker infrastructures. To reduce these shortcomings, future research might focus on hybrid designs or the integration of multimodalities (e.g., using emoji, audio tone or visual cues).

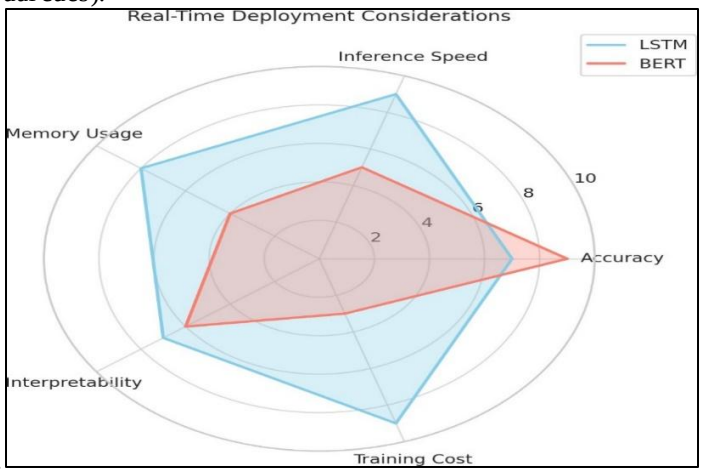

Figure 8. Real-time deployment trade-offs between LSTM and BERT.

The Figure 8 illustrates trade-offs between LSTM and BERT when considering real-time deployment. BERT clearly excels in accuracy, achieving higher scores due to its Capability to understand subtle semantic patterns. However, this approach sacrifices training efficiency and processing speed. LSTM is less resource-intensive and grants much quicker processing, finding favor in latency-critical settings. In the same vein, LSTM shows less memory usage, hence a plus for edge devices or lightweight programs. Because this is for interpretability, both models are equivalent; the sequential model of LSTM offers more enhanced clarity in tracing out the steps for making the prediction. In direct contrast, BERT offers superior predictive accuracy but also brings extra computational overhead during training and runtime. This visualization thus highlighted a very fundamental trade-off: BERT gives the best predictive performance, but LSTM remains the pragmatic choice given considerations of computational efficiency.

5. CONCLUSION

In the comparative experiment done on Amazon Fine Food Reviews, the fine-tuned BERT outperformed the LSTM classifier across accuracy, F1 score, ROC-AUC, and MCC. While BERT is much better at negations and many sarcastic constructs with its bidirectional self-attention, using LSTM is still practical in general and can be preferred in deployments when the resources are limited. Key limitations include continued failure modes for sarcasm and mixed sentiments and BERT's higher compute footprint. Future work needs to be invested in hybrid solutions, aspect-level sentiment extraction for mixed reviews, and multimodal signals like emojis. The study also recommends rigorous bias audits and interpretability checks before the deployment of sentiment systems in decision-critical settings. In addition to these contributions, there are also few limitations that were observed. The two models struggled with sarcasm, irony and mixed-sentiment reviews, as it is a natural challenge of explaining pragmatic and context-dependent nuances of human language. Additionally, the high computational and memory demands that come with BERT, which render it

non-scalable, put a limitation on its accessibility by organizations having a small-scale infrastructure set-up. The proposed gaps can be filled in future research through exploring hybrid architectures that can Combine the scalability and efficiency of recurrent models with the richness and contextual of transformers. In addition, multimodal sources, such as emoji, audio indicators, or visual features, can also be extended and enhanced and can be robust in detecting complex sentiment patterns further. Last but not least, domain-specific with respect to modern datasets utilizing modern internet language (e.g., emoji, hashtags, slang) can further increase generalizability. In conclusion, this research paper verifies that sentiment analysis is a good baseline for LSTM; however, transformer-based models such as BERT set the new benchmark in extracting useful information out of unstructured customer feedback. Predictive accuracy versus computational tractability therefore, must be weighed between organizations as they implement sentiment analysis within their decision-making pipelines in achieving both scalability and impact in practice.

Automated sentiment analysis influences downstream decisions, such as product prioritization and customer response automation. The Amazon Fine Food Reviews dataset lacks demographic metadata and may reflect collection biases (e.g., over/under-representation of particular product types or user populations). These biases can propagate to model outputs and affect decisions. To mitigate such risks, practitioners should (1) report dataset composition and known limitations, (2) use fairness checks and calibration across relevant groups if metadata is available, (3) adopt interpretable models or post-hoc explanation tools (e.g., LIME, SHAP) to audit decisions, and (4) consider human-in-the-loop validation for high-impact decisions.

Additional Information and Declarations

Authors' Contributions

- **Mohammad Arbi Khan:** Conceptualized the research framework, developed the methodology, supervised the overall study, and provided critical revisions to the manuscript.
- **Afsheen Khalid**: Guided the project design, contributed to the literature review, supervised data interpretation, and critically reviewed the manuscript for intellectual content.
- **Muhammad Shahan Ibad**: Guided the project design, contributed to the literature review, supervised data interpretation, and critically reviewed the manuscript for intellectual content.
- **Dilawar Khan**: Assisted in data analysis and visualization, contributed to results interpretation, prepared supporting materials, and reviewed the manuscript draft.
- **Fazal Malik**: Assisted in data analysis and visualization, contributed to results interpretation, prepared supporting materials, and reviewed the manuscript draft.

REFERENCES

[1] Microsoft, "Global State of Multichannel Customer Service Report," Microsoft Corporation, Redmond, WA, USA, Rep., 2020. [Online]. Available: https://info.microsoft.com/CE-CNTNT-CNTNT-FY20-07Jul-GlobalStateofMultichannelCustomerServiceReport.html

[2] Deloitte, "Personalising the Customer Experience." Deloitte. https://www.deloitte.com/ (accessed Dec. 15, 2025).

[3] B. Liu, *Sentiment Analysis and Opinion Mining*. San Rafael, CA, USA: Morgan & Claypool, 2012.

[4] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008, doi: 10.1561/1500000011.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[7] A. Patel, P. Oza, and S. Agrawal, "Sentiment analysis of customer feedback and reviews for airline services using language representation model," *Procedia Comput. Sci.*, vol. 218, pp. 2459–2467, 2023, doi: 10.1016/j.procs.2023.10.392.

[8] P. Akter *et al.*, "Sentiment analysis of consumer feedback and its impact on business strategies by machine learning," *Amer. J. Appl. Sci.*, vol. 7, no. 1, pp. 6–16, 2025. [Online]. Available: https://www.theamericanjournal.org/index.php/tajas/article/

[9] R. J. Bhuiyan *et al.*, "Sentiment analysis of customer feedback in the banking sector: A comparative study of machine learning models," *Amer. J. Eng. Technol.*, vol. 6, no. 10, pp. 54–66, 2024.

[10] S. Akter *et al.*, "A comprehensive study of machine learning approaches for customer sentiment analysis in banking sector," *Amer. J. Eng. Technol.*, vol. 6, no. 10, pp. 100–111, 2024.

[11] N. Shrestha and F. Nasoz, "Deep learning sentiment analysis of Amazon.comreviews and ratings," *arXiv preprint arXiv:1904.04096*, 2019.

[12] K. Gupta, N. Jiwani, and N. Afreen, "A combined approach of sentimental analysis using machine learning techniques," *Revue d'Intell. Artif.*, vol. 37, no. 1, p. 1, 2023, doi: 10.18280/ria.370101.
[13] B. Noori, "Classification of customer reviews using machine learning algorithms," *Appl. Artif. Intell.*, vol. 35, no. 8, pp. 567–588, 2021, doi: 10.1080/08839514.2021.1909617.
[14] L. Ashbaugh and Y. Zhang, "A comparative study of sentiment analysis on customer reviews using machine learning and deep learning," *Computers*, vol. 13, no. 12, p. 340, 2024, doi: 10.3390/computers13120340.
[15] A. Adak, B. Pradhan, and N. Shukla, "Sentiment analysis of customer reviews of food delivery services using deep learning and explainable artificial intelligence: Systematic review," *Foods*, vol. 11, no. 10, p. 1500, 2022, doi: 10.3390/foods11101500.
[16] M. Loukili, F. Messaoudi, and M. El Ghazi, "Sentiment analysis of product reviews for e-commerce recommendation based on machine learning," *Int. J. Adv. Soft Comput. Appl.*, vol. 15, no. 1, 2023.
[17] C. Ahmed, A. ElKorany, and E. ElSayed, "Prediction of customer's perception in social networks by integrating sentiment analysis and machine learning," *J. Intell. Inf. Syst.*, vol. 60, no. 3, pp. 829–851, 2023, doi: 10.1007/s10844-022-00752-4.
[18] S. Yi and X. Liu, "Machine learning-based customer sentiment analysis for recommending shoppers and shops based on customers' reviews," *Complex Intell. Syst.*, vol. 6, no. 3, pp. 621–634, 2020, doi: 10.1007/s40747-020-00187-3.
[19] R. Obiedat *et al.*, "Sentiment analysis of customers' reviews using a hybrid evolutionary SVM-based approach in an imbalanced data distribution," *IEEE Access*, vol. 10, pp. 22260–22273, 2022, doi: 10.1109/ACCESS.2022.3154038.
[20] A. Iqbal, R. Amin, J. Iqbal, R. Alroobaea, A. Binmahfoudh, and M. Hussain, "Sentiment analysis of consumer reviews using deep learning," *Sustainability*, vol. 14, no. 17, p. 10844, 2022, doi: 10.3390/su141710844.
[21] N. Alyas, M. H. Malik, and H. Ghous, "Sentiment analysis using machine learning and deep learning: A survey," *Int. Res. J. Eng. Technol.*, vol. 8, no. 7, pp. 243–251, 2021.
[22] M. H. Malik, H. Ghous, M. I. Ali, M. Ismail, Z. H. Ali, and H. M. Amin, "Sentiment analysis of Roman text: Challenges, opportunities, and future directions," *Int. J. Inf. Secur. Cybercrime Technol.*, vol. 10, no. 2, pp. 56–66, 2023.
[23] M. H. Malik, H. Ghous, and R. Almas, "Multilingual sentiment analysis through integrated multimodal deep learning techniques," *Southern J. Res.*, vol. 15, no. 1, pp. 89–101, 2024.
[24] H. Ghous, M. H. Malik, and R. Almas, "Navigating sarcasm in multilingual text: An in-depth exploration and evaluation," *J. Comput. Biomed. Inform.*, vol. 12, no. 3, pp. 33–47, 2024.
[25] U. Shahid, M. H. Malik, and H. Ghous, "Text preprocessing for Urdu text: A survey of techniques and their influence on NLP tasks," *AMARR*, vol. 4, no. 1, pp. 10–18, 2025.
[26] Y. Liu *et al.*, "RoBERTa: A robustly optimized BERT pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
[27] K. Clark, M. T. Luong, Q. V. Le, and C. D. Manning, "ELECTRA: Pre-training text encoders as discriminators rather than generators," *arXiv preprint arXiv:2003.10555*, 2020.
[28] L. Liu, L. Zhang, and S. Wang, "Deep learning for sentiment analysis: A survey," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 8, no. 4, Art. no. e1245, 2018, doi: 10.1002/widm.1245.