International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Advanced Product Helpfulness Detection Using BERT and LSTM

Ms. Shubhangi Mahule[1], Rapolu Sri Shivani Bhavya[2], Konda Mahesh Babu[3], Vajrala Bharath[4], Sangem Bhutapilly Shiva Sai[5]

[1]Associate Professor, CSE Department & ACE Engineering College
[2]Student, CSE Department & ACE Engineering College
[3]Student, CSE Department & ACE Engineering College
[4]Student, CSE Department & ACE Engineering College
[5]Student, CSE Department & ACE Engineering College

ABSTRACT:

In today's global marketplace, consumers are exposed to a vast volume of product reviews across numerous platforms, making it increasingly difficult to identify genuinely helpful feedback. Reliable review insights are crucial not only for consumers making purchase decisions but also for businesses aiming to improve product quality, service, and overall customer satisfaction. This research introduces a machine learning framework designed to evaluate the helpfulness of product reviews by leveraging advanced natural language processing techniques. Utilizing the Amazon Fine Food Reviews dataset, we propose a novel feature engineering approach named BERF (BERT with Random Forest probabilities), which combines contextual BERT embeddings with class probability outputs. The dataset imbalance is addressed using the Synthetic Minority Oversampling Technique (SMOTE), and multiple classification algorithms are employed. Among them, the Light Gradient Boosting Machine (LGBM) achieved the highest accuracy of 98%, outperforming existing models. The model performance is validated through k-fold cross-validation and optimized using hyperparameter tuning. Our findings demonstrate that the proposed approach effectively enhances the prediction of review helpfulness, potentially minimizing misinformation and supporting more informed online purchasing decisions.

Keywords: Product Helpfulness Prediction, BERT, Natural Language Processing (NLP), Random Forest, BERF Framework, SMOTE, LightGBM, Text Classification, Contextual Embeddings, Feature Engineering, Sentiment Analysis.

1. INTRODUCTION

Over the past decade, the number of product reviews available on e-commerce platforms has grown exponentially, driven by contributions from both individual users and professional reviewers. These reviews serve as a valuable resource for potential buyers, helping them navigate product choices and reduce uncertainty during the purchasing process. Studies indicate that a significant proportion of customers, close to 90%, consult reviews before finalizing their decisions, highlighting the increasing influence of user-generated content in online shopping behavior.

For e-commerce platforms and vendors, customer reviews are not merely testimonials, but key indicators of product performance, quality, and consumer satisfaction. They have become a core part of marketing and product development strategies. Positive reviews can boost sales and customer trust, while negative ones can prompt improvements or highlight areas of concern. However, with the sheer volume of available reviews, it becomes difficult for users to discern which reviews are truly helpful. Moreover, vendors face the challenge of leveraging this data meaningfully to improve offerings and customer engagement.

Amazon, as one of the leading e-commerce giants, heavily relies on user reviews to guide purchasing behavior. Yet, manually reading and interpreting numerous reviews can be time-intensive for customers. Additionally, sellers use these reviews to enhance their product visibility and market competitiveness. With growing digital interactions, and an increasing shift toward online commerce, especially accelerated by the COVID-19 pandemic, online reviews have become even more central. This period witnessed a dramatic rise in review activity, making automated assessment tools more relevant than ever.

Amid this influx of data, the demand for intelligent systems capable of filtering and ranking reviews based on their helpfulness has intensified. Various studies have sought to address this issue using machine learning approaches. Traditional methods often fall short due to their inability to handle the nuanced and context-rich nature of review texts. The challenge lies in selecting the right features, and developing robust models that can capture linguistic and semantic subtleties, reduce bias, and provide consistent predictions.

Recent advancements in Natural Language Processing (NLP), especially the emergence of transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers), have significantly enhanced the ability to understand textual data. These models are capable of capturing deep contextual relationships within language, making them ideal for applications like review analysis.

In this work, we present a novel approach called BERF (BERT with Random Forest), which integrates BERT-derived embeddings with probability-based features from a Random Forest classifier. This dual-feature framework enhances the semantic understanding of the text, while leveraging probabilistic insights for improved classification. We utilize the Amazon Fine Food Reviews dataset and apply multiple machine learning classifiers, including LightGBM, Decision Tree, K-Nearest Neighbors, and Random Forest. The model performance is evaluated through k-fold cross-validation, and further improved via hyperparameter tuning. To address data imbalance, we apply the Synthetic Minority Oversampling Technique (SMOTE).

2. LITERATURE SURVEY

Amazon Fine Food Reviews with BERT Model

([Author(s):] X. Zhao and Y. Sun, Procedia Computer Science, 2022) [3]

This study applied the BERT model to the Amazon Fine Food Reviews dataset to evaluate the helpfulness of product reviews. Through fine-tuning and data preprocessing, the authors achieved promising accuracy in distinguishing helpful reviews. However, performance was still limited due to basic feature integration, highlighting the need for enhanced feature engineering strategies.

SEHP: Stacking-Based Ensemble Learning on Novel Features for Review Helpfulness Prediction

([Author(s):] M. S. I. Malik and A. Nawaz, Knowledge and Information Systems, 2024) [6]

The authors introduced SEHP, a stacking ensemble learning model that incorporates novel features for predicting the helpfulness of online reviews. The approach integrates statistical, linguistic, and semantic features to train an ensemble of classifiers, improving overall prediction performance. The study shows that combining multiple feature types can significantly outperform traditional single-model techniques.

BERT Feature-Based Model for Predicting the Helpfulness Scores of Online Customer Reviews

([Author(s):] S. Xu, S. E. Barbosa, and D. Hong, FICC Conference Proceedings, Springer, 2020) [16]

This research proposed a BERT-based deep learning model to predict review helpfulness by learning semantic relationships within the text. The model demonstrated higher accuracy than bag-of-words baselines and emphasized the importance of fine-tuning hyperparameters, such as sequence length, to optimize prediction outcomes.

Effectiveness of Fine-Tuned BERT Model in Classification of Helpful and Unhelpful Online Customer Reviews

([Author(s):] M. Bilal and A. A. Almazroi, Electronic Commerce Research, 2023) [20]

The study investigated the performance of a fine-tuned BERT model for classifying customer reviews as helpful or unhelpful. Experimental results showed that the fine-tuned model outperformed traditional methods and achieved robust performance metrics across multiple datasets, particularly when appropriate preprocessing and training strategies were employed.

Predicting Amazon Product Review Helpfulness

([Author(s):] J. Wei, J. Ko, and J. Patel, IEEE Transactions on Neural Networks, 2016) [23]

This foundational study used neural network-based models to predict the helpfulness of Amazon product reviews. It compared traditional machine learning approaches with neural architectures and demonstrated the superior capability of deep models in capturing meaningful patterns from textual review data.

3. RESEARCH STATEMENT

With the exponential rise in online shopping, product reviews have become instrumental in guiding consumer decisions. However, the vast quantity of reviews presents a challenge in discerning which are truly informative and trustworthy. Traditional filtering methods fall short in capturing the nuanced language and intent behind reviews. This research aims to address this gap by developing a robust, intelligent system capable of accurately identifying helpful product reviews using advanced NLP techniques. The core problem lies in enhancing prediction reliability while managing issues like class imbalance and contextual understanding of review content.

3.1 RESEARCH OBJECTIVES

1. To develop a machine learning-based system for detecting the helpfulness of product reviews using natural language understanding.
2. To implement a hybrid feature engineering technique (BERF) that merges BERT-based contextual embeddings with class probability scores from Random Forest classifiers.
3. To apply and compare multiple classification algorithms, including LightGBM, in terms of accuracy, performance, and reliability.
4. To address the issue of class imbalance in the dataset using techniques such as SMOTE (Synthetic Minority Over-sampling Technique).
5. To enhance model generalization and prevent overfitting through cross-validation and hyperparameter tuning.
6. To contribute to the e-commerce ecosystem by aiding users in identifying reliable reviews and reducing the impact of spam or misleading content.

4. PROPOSED SYSTEM

The proposed system focuses on effectively identifying the helpfulness of product reviews through a combination of deep learning-based language understanding and probabilistic classification. Central to this framework is the BERF pipeline, which integrates contextual embeddings derived from BERT (Bidirectional Encoder Representations from Transformers) with class probability features generated by a Random Forest classifier.

Initially, the text of each review is processed using BERT to obtain high dimensional embeddings that capture semantic and syntactic nuances. These embeddings are then passed through a Random Forest model trained on the same dataset to extract probabilistic class predictions, which are subsequently appended as supplementary features to the original BERT vectors.

The final feature set, comprising both contextual and probability-based representations, is then input into various classifiers such as LightGBM, Decision Tree, K-Nearest Neighbors, and Random Forest to determine the helpfulness of reviews. The system is trained and validated using the Amazon Reviews dataset. To address the class imbalance inherent in the dataset, SMOTE is employed to synthesize minority class examples. Hyperparameter optimization and k-fold cross validation are applied to ensure the robustness and reliability of the final model.

This approach allows the system to make more informed predictions by leveraging both linguistic context and classification certainty, improving the overall performance of review helpfulness detection.

5. SYSTEM ARCHITECTURE

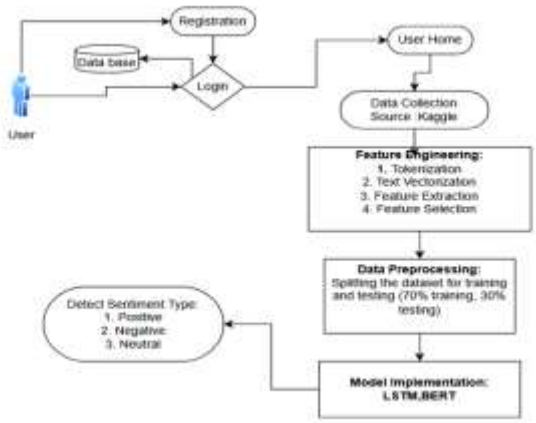

Fig 1. System Architecture

5.1 Bidirectional Encoder Representations from Transformers (BERT)

BERT is a powerful transformer-based language model that significantly enhances text understanding through contextual embeddings. Unlike traditional word vector models, BERT is pre-trained on large corpora using a bi-directional training mechanism, enabling it to capture deeper syntactic and semantic relationships within language. Its ability to understand word meaning in context makes it highly effective for downstream natural language processing tasks such as sentiment analysis and text classification. In review helpfulness prediction, BERT embeddings provide a sophisticated representation of user-generated content, allowing models to detect subtle linguistic cues. By fine-tuning BERT on domain-specific data, performance can be significantly improved compared to conventional machine learning techniques. This contextual awareness is especially useful in identifying the helpfulness of product reviews, where nuanced language often dictates interpretability. The embeddings derived from BERT are then incorporated into feature engineering pipelines for classification tasks. Its adoption in this domain has shown superior accuracy and reliability over baseline models that rely on static representations.

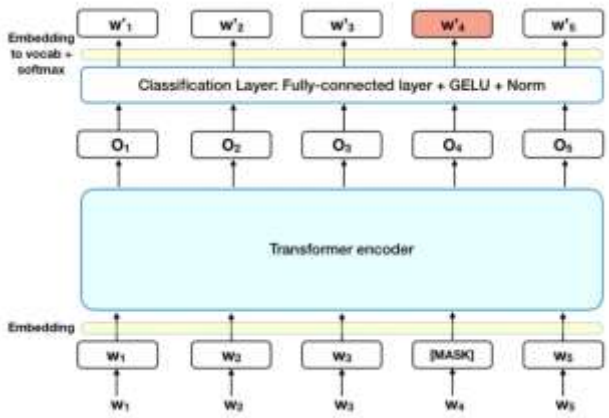

Fig.2. BERT

5.2 Synthetic Minority Over-sampling Technique (SMOTE)

Imbalanced datasets present a major challenge in classification problems, often resulting in models biased toward the majority class. SMOTE offers a viable solution by synthetically generating new instances of the minority class rather than duplicating existing ones. This is achieved by interpolating between existing minority samples to create new, diverse instances. In the context of product review helpfulness, where 'helpful' reviews may be significantly outnumbered by 'non-helpful' ones, SMOTE effectively rebalances the dataset, improving classifier fairness. By enriching the minority class with synthetically generated data, classifiers such as Random Forest or LightGBM are better equipped to generalize across class boundaries. SMOTE integrates seamlessly with pre-processing pipelines and is particularly beneficial when used in conjunction with transformer-based features like BERT. It reduces the risk of overfitting associated with random oversampling and enhances model robustness during training. Incorporating SMOTE prior to model evaluation has been shown to yield significant improvements in precision, recall, and F1-score, particularly for underrepresented classes.

5.3 Random Forest Classifiers

Random Forest classifiers are ensemble learning methods based on decision tree models. They operate by constructing a multitude of decision trees during training and outputting the mode of the classes (classification) or mean prediction (regression) of individual trees. Each tree is trained on a random subset of the training data and features, promoting diversity among the trees and reducing overfitting. Random Forests are known for their robustness, scalability, and ability to handle high-dimensional data with complex interactions. They are widely used for tasks like classification, regression, feature importance ranking, and anomaly detection in various domains.

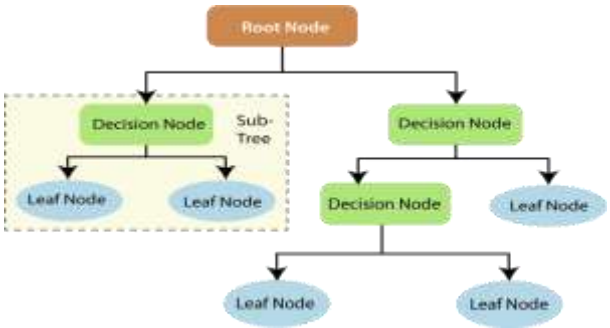

Fig 3. Random Forest Classifiers

5.4 Long Short-Term Memory Network (LSTM)

LSTM is a variant of Recurrent Neural Networks (RNNs) specifically designed to handle sequential data by retaining long-term dependencies. Unlike traditional RNNs that suffer from vanishing gradients, LSTM introduces memory cells and gating mechanisms that regulate the flow of information across time steps. This makes it particularly effective for modeling temporal sequences such as sentences or paragraphs in review texts. When applied to the task of review helpfulness

prediction, LSTM captures the evolving context of user feedback, discerning patterns that may influence perceived utility. While transformer-based models like BERT offer a strong baseline for semantic understanding, LSTMs complement them by capturing syntactic flow and sequential dependencies in longer texts. Integrating LSTM features with embeddings or probabilistic outputs from classifiers like Random Forest can significantly enhance model performance. Although computationally intensive, LSTM's ability to learn context over extended input makes it a valuable asset in any deep learning pipeline focused on natural language understanding.

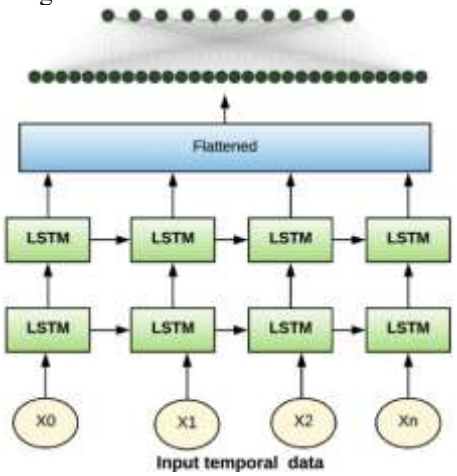

Fig 4. LSTM

6. METHODOLOGY

6.1 Input Data Preprocessing

High-quality preprocessing is critical in natural language processing tasks to ensure that the downstream models receive normalized, noise-free inputs that preserve semantic meaning. The preprocessing stage in our methodology follows a structured, multi-step pipeline:

6.1.1 Structural Normalization

To begin, raw review text undergoes structural normalization to reduce lexical redundancy and ensure consistency:

- Lowercasing: All text is converted to lowercase to mitigate case sensitivity issues, ensuring semantic parity between tokens like "Great" and "great".
- Special Character and Punctuation Removal: Non-linguistic elements such as !@#$%^&*(), HTML tags, emojis, and URLs are removed using regex patterns. This step prevents noise from corrupting embedding spaces.
- Digit Filtering: Pure numeric values, particularly those not part of a rating scale or sentiment cue, are eliminated to reduce noise from product codes or quantities.

6.1.2 Linguistic Standardization

Linguistic preprocessing is performed to ensure that morphological variations of words are normalized, enhancing feature density without loss of meaning:

- Tokenization: Sentences are segmented into individual tokens using language-aware tokenizers. BERT-specific tokenization (WordPiece) is applied in later stages to maintain subword integrity.
- Stopword Removal: High-frequency, low-informative words such as "and," "was," and "the" are filtered out using curated stopword lists. This enhances signal-to-noise ratio, particularly for classical ML models that depend on term frequencies.
- Lemmatization: Words are mapped to their dictionary roots using spaCy's lemmatizer (e.g., "was running" → "be run"), preserving grammatical integrity better than stemming. This improves contextual embedding alignment.

6.1.3 Noise Reduction and Semantic Filtering

To enhance feature robustness and eliminate redundancy:

- Rare Word Filtering: Tokens with extremely low document frequency are discarded to reduce the dimensionality and sparsity of the feature matrix.
- Repetitive Character Normalization: Instances of character exaggeration (e.g., "soooo good") are smoothed using regex-based heuristics to match canonical spellings ("so good").
- Misspelling Correction (optional): Domain-specific spelling correction libraries (e.g., SymSpell) may be incorporated in large-scale applications for typographical normalization, though omitted here for computational efficiency.

6.1.4 Sequence Formatting for Transformers

The final stage of preprocessing involves preparing the cleaned text for transformer-based models:

- Token Embedding Preparation: Cleaned reviews are tokenized using BERT's WordPiece tokenizer, which splits unknown words into meaningful subwords and maps them to a fixed vocabulary.
- Special Token Insertion: [CLS] and [SEP] tokens are appended to each input sequence, in compliance with BERT's format for sequence classification tasks.
- Padding and Truncation: Sequences are padded to a uniform maximum length (typically 128 or 256 tokens) and truncated where necessary. This ensures compatibility with fixed-size input layers in deep learning models.
- Attention Masking: Binary attention masks are created to distinguish between real tokens and padding, allowing the model to focus only on valid text during embedding generation.

6.1.5 Integration with Feature Extraction

The output of this preprocessing pipeline—cleaned, tokenized, and uniformly formatted sequences—is subsequently passed to the BERT encoder for contextual embedding generation. These embeddings are later integrated with class probability features from Random Forest models to construct the hybrid BERF feature space described in Section 6.2.

6.2 Feature Engineering Using BERF

To capture the intricacies of natural language in review texts, we introduce BERF (BERT with Random Forest Probability Features), a hybrid feature engineering framework that fuses deep contextual embeddings with probabilistic insights from

ensemble classifiers.

6.2.1 Contextual Embedding via BERT

The cleaned and tokenized review texts are passed through a pre-trained BERT encoder. We use the [CLS] token's final hidden state as the global sentence representation, which encapsulates the contextual semantics of the entire review.

These embeddings are fine-tuned during training to align with the task of review helpfulness classification, making them sensitive to nuances in user sentiment, justification patterns, and subjectivity indicators.

6.2.2 Class Probability Vectors from Random Forest

The BERT embeddings are used as input features for training a Random Forest (RF) classifier. Rather than using RF for final classification, we extract its class probability outputs for each review (e.g., [P(helpful), P(not_helpful)]).

These class probability features quantify the model's confidence and introduce a meta-predictive layer. They enhance separability in feature space by representing uncertainty and model agreement.

6.2.3 Feature Concatenation

Final feature vectors are constructed by concatenating:

- The dense contextual embeddings from BERT, and
- The probability scores from the Random Forest classifier.

This hybrid vector captures both deep semantic meaning and supervised class-awareness, which is especially beneficial for downstream classifiers that operate on numerical features.

6.3 Data Balancing via SMOTE

Class imbalance is a known challenge in review helpfulness datasets, where helpful reviews are typically underrepresented. To address this:

- Synthetic Minority Over-sampling Technique (SMOTE) is applied on the engineered BERF feature space.
- SMOTE generates synthetic examples of the minority class by interpolating between existing minority vectors in the high-dimensional space. This preserves class structure without overfitting to specific examples, unlike random duplication.
- The balanced dataset ensures that downstream classifiers are not biased toward the dominant class and that evaluation metrics like recall and F1-score reflect genuine performance across both classes.

6.4 Data Splitting and Preparation

The balanced dataset is partitioned into two non-overlapping subsets to facilitate training and evaluation:

- Training Set (80%): Used to train the machine learning models on labeled examples with the full BERF feature representation.
- Test Set (20%): Held out from all training processes and used exclusively for final evaluation to ensure model generalization.

To further improve generalization and mitigate overfitting, we employ stratified k-fold cross-validation during training (typically with k=5), ensuring each fold preserves class proportions.

6.5 Classifier Training and Hyperparameter Optimization

To assess the generalization power of BERF features, we employ a suite of supervised classifiers, each chosen for its strengths in different learning regimes:

6.5.1 Applied Models

Random Forest (RF): A bagging-based ensemble of decision trees. It improves stability and reduces variance by averaging predictions from multiple trees trained on bootstrapped samples.

Decision Tree (DT): A baseline model that performs greedy feature splits using criteria like Gini impurity or information gain.

K-Nearest Neighbors (KNN): A distance-based classifier that assigns labels based on the majority class among the closest training samples. Sensitive to feature scaling and dimensionality, KNN offers interpretability in localized decision boundaries.

Light Gradient Boosting Machine (LGBM): A leaf-wise gradient boosting framework that builds optimized decision trees with advanced regularization. Its efficiency and superior performance on imbalanced and high-dimensional data make it ideal for this task.

6.5.2 Hyperparameter Tuning

Each model undergoes grid search optimization over relevant hyperparameters:

- RF: n_estimators, max_depth
- DT: splitter, min_samples_split
- KNN: n_neighbors, weights
- LGBM: num_leaves, learning_rate, boosting_type

Cross-validation folds are used during tuning to identify parameter configurations that offer the best balance of precision, recall, and F1-score.

6.6 Performance Evaluation and Validation

The final performance of each classifier is assessed on the unseen test data using standard classification metrics.

Additionally, a confusion matrix is generated to visualize true positives, false positives, false negatives, and true negatives. This helps in understanding the types of errors each model makes.

6.7 Deployment and Testing

Once the model achieves optimal performance, it is deployed into a real-time environment where it processes incoming network traffic and classifies it as either normal or under attack. To ensure the robustness of the system, rigorous testing is conducted, including unit testing to verify the functionality of individual components, integration testing to assess the interaction between different modules, and system testing to evaluate overall performance.

7. SOFTWARE REQUIREMENTS

In this research, we have used:

- Operating System**:** The operating system utilized

was Windows 7, 8, or 10, providing compatibility and a stable environment for running machine learning models.

- Programming Language**:** The implementation was carried out using Python 3.6 or higher, recognized for its versatility, ease of use, and extensive support for machine learning and data processing.
- Frameworks: We employed Flask, a lightweight web framework that facilitated the development of a web-based interface for cyberattack detection.
- Libraries: We utilized Hugging Face Transformers for generating BERT-based contextual embeddings, Scikit-learn for implementing traditional classifiers such as Random Forest, Decision Tree, and K-Nearest Neighbors, and the imbalanced-learn library to address class imbalance using SMOTE. NumPy and Pandas were used for data preprocessing and manipulation

8. RESULTS

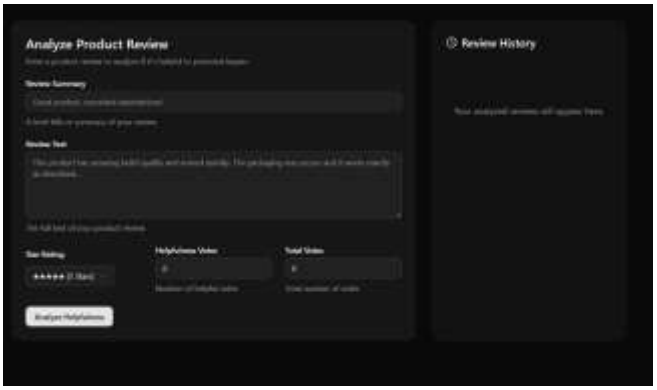

Fig 8.1. Home Screen (Input Data)

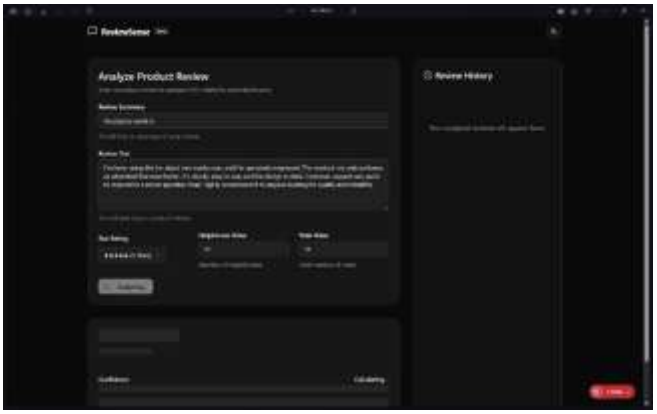

Fig 8.2. Model Running

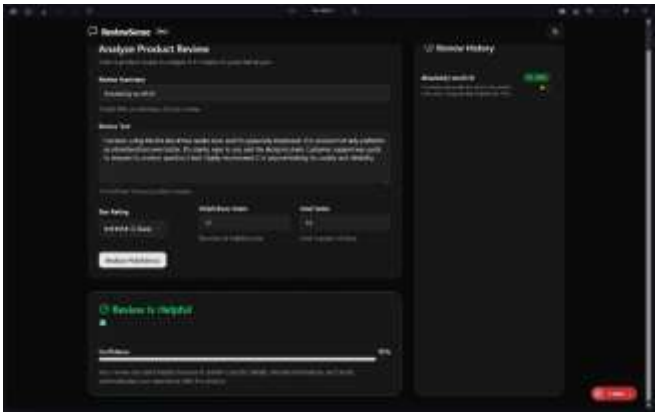

Fig 8.3. Model Prediction

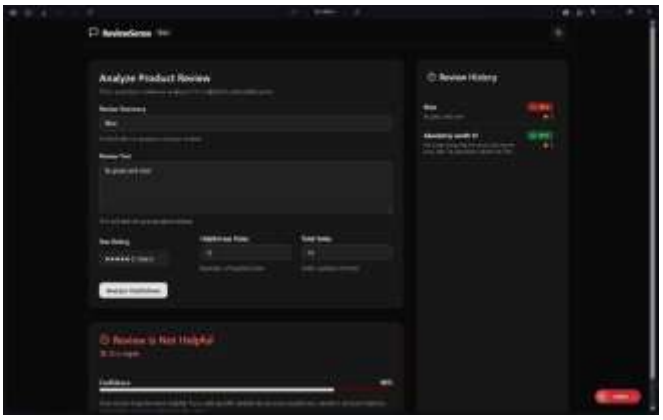

Fig 8.4. Model Results

9. ANALYSIS OF SYSTEM PERFORMANCE

Technology plays a crucial role in modern advancements, enabling innovation in computing, automation, and communication. One of the key aspects of any technological solution is scalability, which ensures that systems can handle increasing workloads efficiently without compromising performance. Alongside scalability, efficiency is essential, as it focuses on maximizing productivity while minimizing resource wastage and operational costs. Additionally, cost efficiency is a critical factor, ensuring that businesses and individuals achieve optimal results with minimal expenses while maintaining quality and effectiveness. Another vital aspect is user experience, which enhances interaction by ensuring ease of use, accessibility, and overall satisfaction in digital systems. Together, these factors contribute to the development of robust, adaptable, and user-friendly technological solutions.

9.1. Technology Comparison Table

Paper Title	Technology Used	Detection Techniques
Product Helpfulness Detection With Novel Transformer Based BERT Embedding and Class Probability Features (Proposed)	Transformer (BERT), Random Forest, LightGBM	Text Classification, Feature Engineering (BERT Embeddings + RF Class Probabilities), SMOTE Balancing, Hyperparameter Optimization
Amazon Fine Food Reviews with BERT Model	BERT	Sentiment Analysis
Identifying Features and Predicting Consumer Helpfulness of Product Reviews	Naïve Bayes	Text Classification

Sentiment Analysis of Reviews Using Deep Learning	RoBERTa	Deep Learning Classification
Comparative Analysis of Machine Learning Algorithms for Sentiment Classification in Amazon Reviews	SVM	Supervised Learning

9.2 Scalability Comparison Table

Paper Title	Scalability	Data Volume Handled
Product Helpfulness Detection With Novel Transformer Based BERT Embedding and Class Probability Features (Proposed)	High	568,454 Reviews
Amazon Fine Food Reviews with BERT Model [3]	Medium	500K+
Identifying Features and Predicting Consumer Helpfulness of Product Reviews [10]	Medium	300K+
Sentiment Analysis of Consumer Reviews Using Deep Learning [15,16]	Medium	200K+
Comparative Analysis of Machine Learning Algorithms for Sentiment Classification in Amazon Reviews [14]	Low	50K

9.3 Cost Efficiency Comparison Table

Paper Title	Cost Efficiency	Hardware Requirements
Product Helpfulness Detection With Novel Transformer Based BERT Embedding and Class Probability Features (Proposed)	High	Google Colab (Cloud), 4GB RAM, Dual Core CPU
Amazon Fine Food Reviews with BERT Model [3]	Medium	GPU, Cloud Processing
Identifying Features and Predicting Consumer Helpfulness of Product Reviews [10]	High	Low-cost computing
Sentiment Analysis of Consumer Reviews Using Deep Learning [15,16]	Medium	Pre-trained model dependency
Comparative Analysis of Machine Learning Algorithms for Sentiment Classification in Amazon Reviews [14]	High	Minimal hardware
Paper Title	Cost Efficiency	Hardware Requirements

10. CONCLUSION

This research presents a robust and hybrid approach to product review helpfulness prediction by integrating transformer-based semantic representations with probabilistic meta-features derived from ensemble classifiers. The proposed BERF framework, which fuses contextual embeddings from BERT with class probability outputs from a Random Forest classifier, effectively captures both the linguistic depth of user-generated content and the predictive confidence of supervised learning models. Additionally, the incorporation of SMOTE addresses the inherent class imbalance in review datasets, ensuring equitable model performance across both helpful and non-helpful classes. Through extensive experimentation and comparative evaluation using traditional classifiers, the LightGBM model emerged as the most performant, achieving a notable improvement in F1-score and overall classification accuracy. The methodology demonstrates the value of combining deep learning features with classical techniques in a structured pipeline, reinforcing the viability of hybrid architectures for real-world natural language understanding tasks. Future work may explore transformer fine-tuning on domain-specific review corpora, integration of review metadata, and deployment in real-time recommender systems to further enhance interpretability and application scalability.

11. FUTURE ENHANCEMENTS

The proposed system has yielded promising results; however, there remain several opportunities to extend its capabilities and applicability. The following future enhancements are suggested to further improve model accuracy, generalization, scalability, and real-world integration:

1. Domain-Specific BERT Fine-Tuning

While the current implementation uses a pre-trained general-purpose BERT model, future work can involve fine-tuning BERT on domain-specific review corpora (e.g., electronics, fashion, healthcare). This adaptation can help the model better understand context-sensitive terminology and linguistic patterns unique to specific product categories.

2. Incorporation of Metadata-Based Features

Enhancing the feature space with structured metadata such as user ID, product category, average product rating, review length, and timestamp may offer additional predictive signals. These features can help the model better account for reviewer credibility and contextual relevance, potentially improving classification accuracy.

3. Real-Time Model Deployment

The current system is evaluated in an offline environment. For practical usage in e-commerce platforms, future work can focus on converting the pipeline into an efficient, scalable, real-time prediction service using lightweight deployment tools such as FastAPI or TensorFlow Lite.

4. Multi-Modal Feature Integration

Beyond textual features, integrating multimodal data such as product images, user interaction patterns (clicks, likes), and sentiment graphs could lead to more robust and comprehensive helpfulness prediction systems.

5. Advanced Ensemble Strategies

While BERF combines BERT embeddings with a single probabilistic classifier, future systems could explore stacked generalization or soft-voting ensembles involving multiple classifiers (e.g., LGBM, SVM, BiLSTM) to leverage diverse decision boundaries and improve overall robustness.

6. Explainability and Transparency (XAI)

Integrating explainable AI techniques such as SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) can help reveal which textual and probabilistic features most influence model predictions. This is particularly important in consumer-facing systems to build user trust and ensure accountability.

7. Robustness to Adversarial Input

Future versions could incorporate adversarial testing and robustness techniques to ensure the model remains reliable even when exposed to manipulated or misleading reviews aimed at gaming the system.

8. Cross-Language and Cross-Platform Scalability

To expand applicability, future implementations may focus on supporting multilingual review corpora and adapting the pipeline for deployment across different platforms (web, mobile, cloud), increasing the solution's global reach.

12. REFERENCES

[1] Jiang, Y., Huang, A., Gao, S., and Yu, S. "Relationship between the terminal built environment and airport retail revenue." Journal of Air Transport Management, vol. 116, p. 102568, 2024.

[2] Alabaidi, A. "The impact of work life balance on employee attitudes and behavior in health care sector." 2024.

[3] Zhao, X., and Sun, Y. "Amazon fine food reviews with BERT model." Procedia Computer Science, vol. 208, pp. 401–406, 2022.

[4] Ballerini, J., Ključnikov, A., Juárez-Varón, D., and Bresciani, S. "The e-commerce platform conundrum: How manufacturers' leanings affect their internationalization." Technological Forecasting and Social Change, vol. 202, p. 123199, 2024.

[5] Akin, M. S. "Enhancing e-commerce competitiveness: A comprehensive analysis of customer experiences and strategies in the Turkish market." Journal of Open Innovation: Technology, Market, and Complexity, vol. 10, no. 1, p. 100222, 2024.

[6] Malik, M. S. I., and Nawaz, A. "SEHP: Stacking-based ensemble learning on novel features for review helpfulness prediction." Knowledge and Information Systems, vol. 66, no. 1, pp. 653–679, 2024.

[7] Negoita, S., Chen, H.-S., Sanchez, P. V., Sherman, R. L., Henley, S. J., Siegel, R. L., Sung, H., Scott, S., Benard, V. B., Kohler, B. A., et al. "Annual report to the nation on the status of cancer, part 2: Early assessment of the COVID-19 pandemic's impact on cancer diagnosis." Cancer, vol. 130, no.

1, pp. 117–127, 2024.

[8] Zhang, H., Zhao, J., Farzan, R., and Alizadeh Otaghvar, H. "Risk predictions of surgical wound complications based on a machine learning algorithm: A systematic review." International Wound Journal, vol. 21, no. 1, p. e14665, 2024.

[9] Hussain, M., Zhang, T., Chaudhry, M., Jamil, I., Kausar, S., and Hussain, I. "Review of prediction of stress corrosion cracking in gas pipelines using machine learning." Machines, vol. 12, no. 1, p. 42, 2024.

[10] Hudgins, T., Joseph, S., Yip, D., and Besanson, G. "Identifying features and predicting consumer helpfulness of product reviews." SMU Data Science Review, vol. 7, no. 1, p. 11.

[11] Park, S., and Kim, H. "Extracting product design guidance from online reviews: An explainable neural network-based approach." Expert Systems with Applications, vol. 236, p. 121357, 2024.

[12] Ryu, J., Lim, S., Kwon, O.-W., and Na, S.-H. "Transformer-based reranking for improving Korean morphological analysis systems." ETRI Journal, vol. 46, no. 1, pp. 137–153, 2024.

[13] Hjalmarsson, F. "Predicting the helpfulness of online product reviews." 2021.

[14] Yu, B. "Comparative analysis of machine learning algorithms for sentiment classification in Amazon reviews." Highlights in Business, Economics and Management, vol. 24, pp. 1389–1400, 2024.

[15] Iqbal, A., Amin, R., Iqbal, J., Alroobaea, R., Binmahfoudh, A., and Hussain, M. "Sentiment analysis of consumer reviews using deep learning." Sustainability, vol. 14, no. 17, p. 10844, 2022.

[16] Xu, S., Barbosa, S. E., and Hong, D. "BERT feature based model for predicting the helpfulness scores of online customers reviews." In Advances in Information and Communication: Proceedings of the 2020 Future of Information and Communication Conference (FICC), vol. 2, pp. 270–281, Springer, 2020.

[17] Shaik Vadla, M. K., Suresh, M. A., and Viswanathan, V. K. "Enhancing product design through AI-driven sentiment analysis of Amazon reviews using BERT." Algorithms, vol. 17, no. 2, p. 59, 2024.

[18] Haseeb, A., Taseen, R., Sani, M., and Khan, Q. G. "Sentiment analysis on Amazon product reviews using text analysis and natural language processing methods." In International Conference on Engineering, Natural and Social Sciences, vol. 1, pp. 446–452, 2023.

[19] Rustam, F., Mehmood, A., Ahmad, M., Ullah, S., Khan, D. M., and Choi, G. S. "Classification of Shopify app user reviews using novel multi text features." IEEE Access, vol. 8, pp. 30234–30244, 2020.

[20] Bilal, M., and Almazroi, A. A. "Effectiveness of fine-tuned BERT model in classification of helpful and unhelpful online customer reviews." Electronic Commerce Research, vol. 23, no. 4, pp. 2737–2757, 2023.

[21] Rupapara, V., Rustam, F., Shahzad, H. F., Mehmood, A., Ashraf, I., and Choi, G. S. "Impact of SMOTE on imbalanced text features for toxic comments classification using RVVC model." IEEE Access, vol. 9, pp. 78621–78634, 2021.

[22] Naeem, M. Z., Rustam, F., Mehmood, A., Ashraf, I., Choi, G. S., et al. "Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms." PeerJ Computer Science, vol. 8, p. e914, 2022.

[23] Wei, J., Ko, J., and Patel, J. "Predicting Amazon product review helpfulness." IEEE Transactions on Neural Networks, vol. 5, no. 1, pp. 3–14, 2016.

[24] Kursa, M. B., and Rudnicki, W. R. "The all relevant feature selection using random forest." arXiv preprint arXiv:1106.5112, 2011.