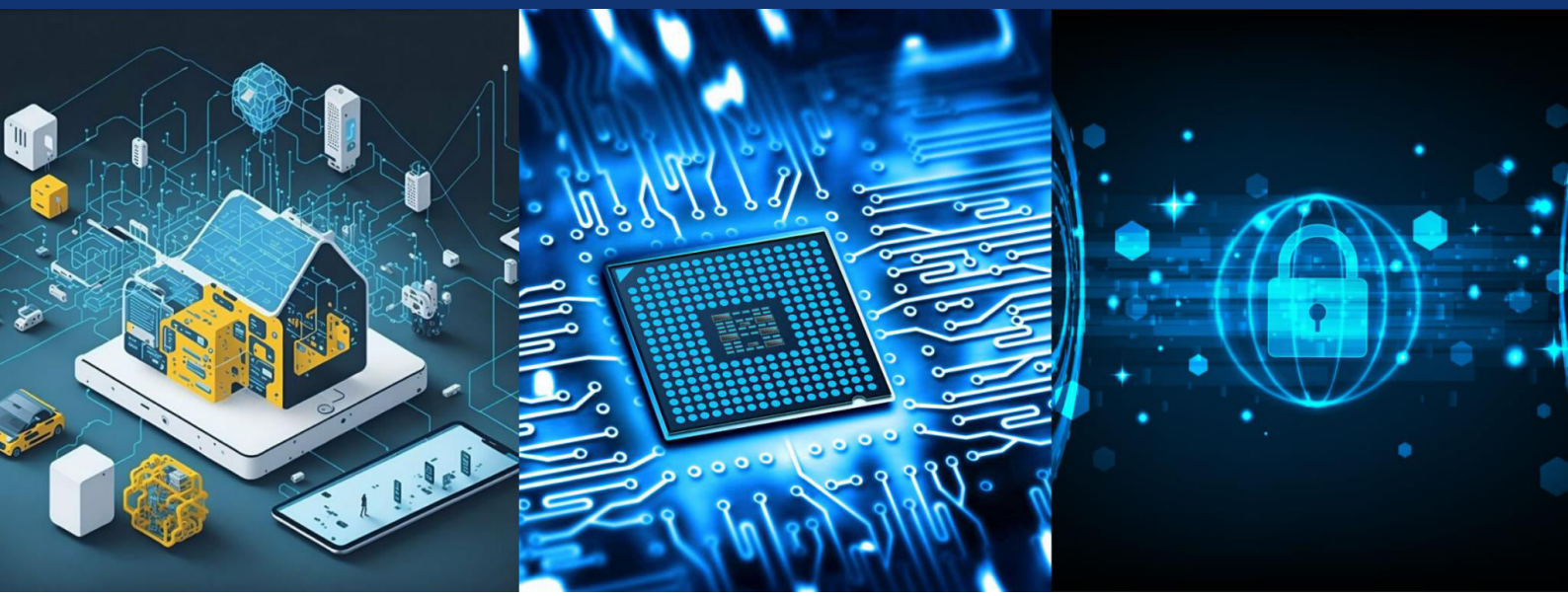
IJIRCCE

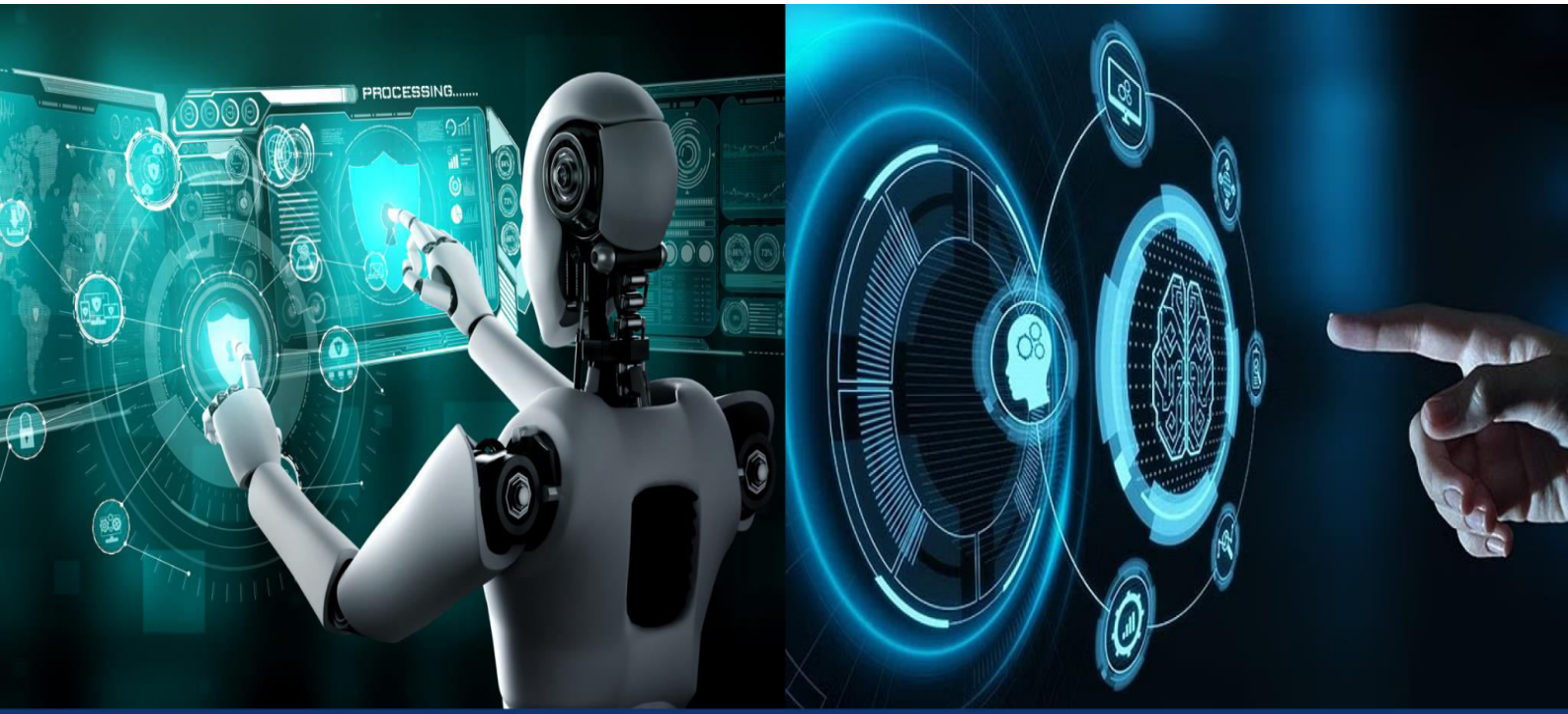
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 4, April 2025

© 2025 IJIRCCE | Volume 13, Issue 4, April 2025| DOI:10.15680/IJIRCCE.2025.1304288

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

E-Commerce Trust Score System using AI – Validation

T Usha Rani[1], V Pavan Kumar[2], V Monika Pavani[3], V Sivasankar[4], V Sambhavi[5], V Venkata Aditya[6]

Assistant Professor, Department of CSE, Sir C R Reddy College of Engineering, India[1]

B. Tech Student, Sir C R Reddy College of Engineering, India[2-6]

ABSTRACT: The proliferation of e-commerce platforms has redefined consumer purchasing behavior by offering instant access to a vast array of products and services. A crucial element influencing online purchase decisions is user-generated product reviews. However, the increasing prevalence of fake or manipulated reviews poses a significant threat to consumer trust and platform integrity. This paper presents the *E-Commerce Trust Score System*, an AI-driven solution designed to enhance transparency and credibility in online shopping. The system integrates a web-based interface developed using HTML, CSS, and JavaScript with a Flask-based Python backend. It employs Term Frequency-Inverse Document Frequency (TF-IDF) vectorization and a Multinomial Naive Bayes classifier to detect review authenticity, augmented by sentiment analysis using TextBlob. The output is a trust score that visually represents the proportion of real versus fake reviews along with sentiment polarity. Experimental results demonstrate the system's effectiveness in classifying reviews and providing trust metrics. The proposed model offers a scalable foundation for future enhancements, including product-specific trust scoring and integration with commercial e-commerce platforms.

KEYWORDS: E-commerce, fake reviews, trust score, machine learning, sentiment analysis, web application, Naive Bayes, TF-IDF.

I. INTRODUCTION

The rise of e-commerce has revolutionized the retail landscape by providing consumers with unprecedented access to a diverse range of products and services. Online shopping platforms like Amazon, Flipkart, and Alibaba have streamlined purchasing processes, reduced transaction costs, and fostered global consumer connectivity. A key factor influencing consumer decisions in this digital environment is user-generated content, particularly product reviews. These reviews serve as a form of electronic word-of-mouth (eWOM), significantly shaping consumer perception, product trustworthiness, and purchasing behavior [1].

Despite their importance, product reviews are increasingly susceptible to manipulation. Fake reviews, also known as opinion spam, are artificially crafted to mislead consumers—either promoting substandard products or discrediting competitors. These reviews compromise the reliability of e-commerce platforms and undermine customer trust. Studies show that as much as 20–30% of reviews on major platforms may be fraudulent [2], creating a pressing need for robust detection mechanisms. Platforms such as Yelp and Amazon have attempted to address this issue through machine learning-based filters, yet the problem persists due to the evolving sophistication of spammers [3].

Detecting fake reviews poses unique challenges. Deceptive reviews are often linguistically similar to genuine ones, making manual detection unreliable and inefficient. Traditional rule-based approaches fail to scale with data volume and are easily circumvented by adversarial tactics. Consequently, researchers have turned to machine learning (ML) and natural language processing (NLP) methods to identify patterns and anomalies in review content. Supervised learning techniques, such as Naive Bayes, SVM, and logistic regression, have demonstrated efficacy in distinguishing genuine and fake reviews based on textual and behavioral features [4]. Deep learning approaches, including recurrent and convolutional neural networks, further enhance detection by capturing complex semantic relationships [5].

Another important dimension in understanding reviews is sentiment analysis, which extracts subjective information to classify reviews as positive, negative, or neutral. Sentiment analysis provides insight into consumer attitudes and can help uncover sentiment inconsistencies that signal manipulation. For instance, a review with overtly positive language but a low star rating might indicate an automated or incentivized post [6]. Tools such as TextBlob and VADER allow for scalable sentiment analysis and are often integrated into fake review detection frameworks [7].

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Given the dual challenges of detecting fake reviews and assessing sentiment, integrated systems that combine authenticity classification and sentiment analysis have emerged as effective solutions. This paper introduces the E-Commerce Trust Score System, a web-based application designed to validate the credibility of product reviews in real-time. The system leverages TF-IDF vectorization and a Multinomial Naive Bayes classifier to classify reviews as "Real" or "Fake," supplemented by TextBlob for sentiment scoring. The outcome is a transparent trust score presented through visual statistics on review authenticity and emotional tone.

This research contributes to the growing field of trustworthy AI in e-commerce by presenting a practical tool that enhances consumer decision-making and supports platform integrity. It bridges the gap between academic methods and real-world applications by offering a lightweight, interpretable system that users can interact with via a web interface. Furthermore, this system lays the groundwork for future enhancements such as product-specific trust scoring and integration with e-commerce APIs for real-time review validation.

II. RELATED WORK

The detection of fake product reviews in e-commerce has become an active area of research in recent years, driven by the growing reliance of consumers on online reviews for informed purchasing decisions. This section reviews key approaches and methods previously proposed in the literature, focusing on fake review detection, sentiment analysis, and trust scoring mechanisms.

A. Fake Review Detection

Several studies have employed machine learning techniques for detecting fake reviews. Mukherjee et al. [8] were among the earliest to explore opinion spam, analyzing behavioral footprints and linguistic cues to detect deceptive content. Ott et al. [9] built a gold-standard dataset of deceptive reviews and trained supervised models like SVM and Naive Bayes to classify them. Their work demonstrated that linguistic patterns, such as exaggerated language and excessive use of first-person pronouns, are useful features for classification.

Recent research has expanded to include deep learning models. Li et al. [10] applied Convolutional Neural Networks (CNNs) to automatically extract features from review text, while Chen et al. [11] used BERT embeddings to enhance context-aware detection. These methods significantly improved classification performance over traditional techniques.

B. Sentiment Analysis for Review Validation

Sentiment analysis is often integrated with review authenticity checks to provide a richer understanding of user feedback. Pang and Lee [12] pioneered sentiment polarity classification, showing that machine learning can successfully distinguish between positive and negative reviews. More recent studies use hybrid approaches, combining lexicon-based techniques (e.g., TextBlob, VADER) with deep learning to analyze emotional tone and detect inconsistencies between sentiment and review ratings [13].

The inclusion of sentiment analysis helps reveal discrepancies that may indicate deception, such as overly positive language paired with low product ratings. Researchers have also proposed sentiment trajectories across multiple reviews to identify unusual fluctuations that suggest manipulation [14].

C. Trust Scoring and Credibility Models

A parallel line of research focuses on building trust scores to quantify the reliability of reviews. Jindal and Liu [15] introduced a model to identify duplicate reviews and assign credibility scores. Wang et al. [16] proposed a probabilistic graphical model that incorporates user behavior and review content to assess trustworthiness.

More recent frameworks combine metadata (e.g., reviewer history, review length, review helpfulness) with content-based features to generate review credibility scores [17]. Some systems, such as TrustPilot and Fakespot, use proprietary algorithms that estimate the trust level of reviews but lack transparency.

While significant progress has been made, existing systems often struggle with generalizing to new types of spam or coping with adversarial attacks. Moreover, few models offer end-user transparency or integrate both sentiment and authenticity insights in a unified platform. Our work addresses these limitations by developing a web-based system that combines TF-IDF vectorization, Naive Bayes classification, and sentiment analysis for real-time review validation and trust scoring.

© 2025 IJIRCCE | Volume 13, Issue 4, April 2025| DOI:10.15680/IJIRCCE.2025.1304288

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

III. PROPOSED MODEL

The proposed model, E-Commerce Trust Score System (ECTSS), is a comprehensive framework designed to evaluate the authenticity and sentiment of product reviews in an online shopping environment. The objective of this system is to provide a transparent trust score for each product based on the credibility of its reviews, thereby supporting both consumer decision-making and platform integrity.

The ECTSS architecture comprises two main modules: (1) Fake Review Detection and (2) Sentiment Analysis. These modules are integrated into a web-based application with a user-friendly interface.

3.1. Fake Review Detection Module

The detection of fake reviews is accomplished using a supervised machine learning pipeline. Initially, raw reviews are subjected to a pre-processing stage which includes tokenization, stop-word removal, and normalization. The cleaned data is then vectorized using the Term Frequency-Inverse Document Frequency (TF-IDF) method, which effectively represents the importance of terms in the review corpus [18].

For classification, the Multinomial Naive Bayes algorithm is employed. This model is well-suited for text classification tasks and demonstrates competitive accuracy with minimal computational overhead [19]. The algorithm is trained on a labeled dataset comprising real and fake reviews, and it outputs a binary classification label indicating the authenticity of each review. The proportion of reviews classified as real or fake is used to compute a product-specific trust score.

3.2. Sentiment Analysis Module

To further understand the nature of the reviews, sentiment analysis is conducted using TextBlob, a widely used Python library for textual data analysis. TextBlob assigns polarity scores ranging from -1 (negative sentiment) to +1 (positive sentiment) and subjectivity scores to each review [20]. This helps identify discrepancies between emotional tone and rating behavior, which may indicate deceptive intent. The sentiment distribution is also visualized, showing the ratio of positive to negative sentiments.

3.3. System Implementation

The frontend of the system is developed using HTML, CSS, and JavaScript to ensure responsive design and intuitive navigation. The backend is implemented using Python and Flask, providing RESTful APIs that interact with the ML models and serve data to the frontend. The system workflow involves the user pasting or uploading product reviews, which are then processed to display real/fake classifications, sentiment statistics, and an overall trust score for the product.

3.4. Trust Score Calculation

The trust score for a given product is computed using a weighted combination of the percentage of real reviews and the proportion of positive sentiment reviews. Let R be the ratio of real reviews, and S be the ratio of positive sentiments. The final trust score (T) is calculated as:

$$T = \alpha R + (1 - \alpha)S$$

where is a weighting factor that balances authenticity and sentiment, typically set between 0.5 and 0.7 based on empirical evaluation.

This hybrid approach leverages the strengths of both machine learning and sentiment analysis, offering a data-driven metric for consumer trustworthiness. It aligns with contemporary studies that advocate for integrated solutions to combat review manipulation in e-commerce [21]

The E-Commerce Trust Score System is a web-based application that integrates a user-friendly frontend, a robust backend, and an AI-driven machine learning model to validate e-commerce reviews and compute trust scores. This section outlines the system's architecture, detailing its components and their interactions. The architecture is designed to be modular, scalable, and efficient, ensuring seamless data flow from user input to AI analysis and result presentation.

3.5 High-Level Architecture Diagram

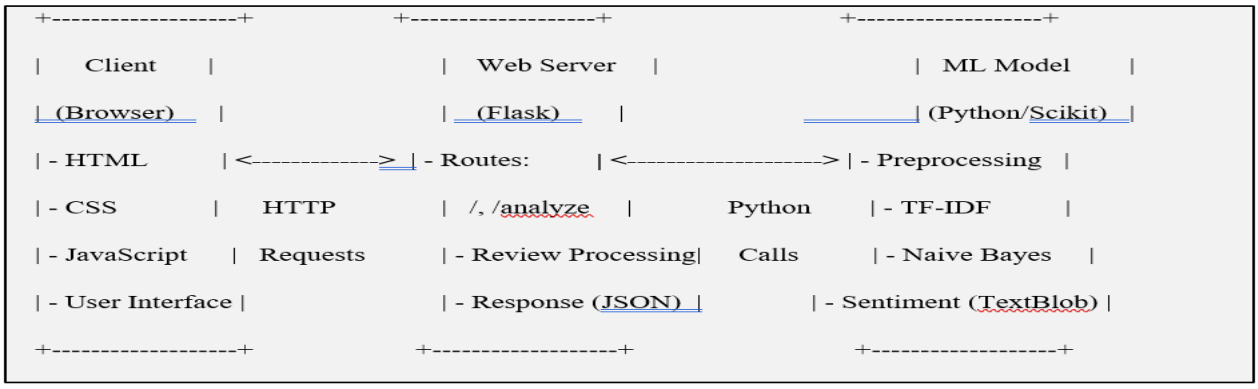
The system follows a client-server architecture with an embedded machine learning component. Below is a textual description of the high-level architecture (a diagram would typically be included in a formatted document):

© 2025 IJIRCCE | Volume 13, Issue 4, April 2025| DOI:10.15680/IJIRCCE.2025.1304288

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



- **Client (Browser)**: The user interacts with the system via a web browser, where the frontend renders the interface and sends review data to the server.
- **Web Server (Flask)**: Acts as the intermediary, handling HTTP requests, processing reviews, and coordinating with the machine learning model.
- **Machine Learning Model**: A standalone component integrated into the backend, responsible for authenticity classification and sentiment analysis.

Data Flow:

1. The user submits a review through the frontend.
2. JavaScript sends an HTTP POST request to the Flask server's /analyze endpoint.
3. The server preprocesses the review and invokes the ML model.
4. The model returns authenticity and sentiment results, which the server aggregates into trust scores.
5. The server sends a JSON response back to the frontend for display.

This architecture ensures separation of concerns, allowing independent development and scaling of each layer.

The machine learning (ML) model is the intellectual core of the E-Commerce Trust Score System, enabling the system to validate review authenticity, analyze sentiment, and compute trust scores. This section delves into the ML pipeline, detailing data preprocessing, algorithm selection (TF-IDF and Naive Bayes), model training, sentiment analysis with TextBlob, and the trust score calculation mechanism. These components work together to transform raw review text into actionable insights, empowering users to assess review credibility in an e-commerce context.

3.6 Data Pre-processing

Data preprocessing is a critical step that prepares raw review text for machine learning analysis by cleaning and standardizing it. This ensures the model focuses on meaningful patterns rather than noise. Figure 1 shows the Data preprocessing pipeline transforming raw reviews for analysis in the Trust Score System.

Fig.1. Data preprocessing pipeline transforming raw reviews for analysis in the Trust Score System

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

1. **Input Validation**: Checks if the input is a string and not null; returns an empty string otherwise.
   Why: Prevents errors from non-text inputs (e.g., numbers, NaN).
2. **Lowercasing**: text.lower() converts all characters to lowercase.
   Why: Ensures "Great" and "great" are treated as the same word.
3. **Cleaning**: re.sub(r'[^a-zA-Z\s]', '', text) removes non-alphabetic characters (e.g., numbers, punctuation).
   Why: Focuses on words, reducing noise from symbols like "!" or "5".
4. **Tokenization**: word_tokenize(text) splits text into individual words (tokens).
   Why: Enables word-level analysis for feature extraction.
5. **Stop Word Removal**: Filters out common words (e.g., "the," "and") using NLTK's stopwords.
   Why: Eliminates frequent, low-value words to emphasize content-specific terms.
6. **Rejoining**: Joins tokens back into a single string with spaces.
   Why: Prepares text for TF-IDF vectorization, which expects a string input.

Libraries Used

- **NLTK (Natural Language Toolkit)**: Provides word_tokenize and stopwords for robust text processing.
- **re**: Enables regex-based cleaning for efficiency and precision.
- **pandas**: Handles NaN checks in training (train_model.py).

Example

- **Input**: "This product is AMAZING!!! I love it, 5 stars."
- **Output**: "product amazing love stars"
- **Effect**: Removes punctuation, numbers, and stop words ("this," "is," "I," "it"), retaining key terms.

Design Rationale

- **Simplicity**: Focuses on basic preprocessing to balance effectiveness and computational cost.
- **Consistency**: Applied uniformly in training (train_model.py) and inference (app.py) for reliable results.
- **Limitations**: Does not handle misspellings, slang, or multi-word phrases (e.g., "works well"), which could be addressed with lemmatization or n-grams in future iterations.

3.7 Algorithm Selection (TF-IDF, Naive Bayes)

The system employs a combination of TF-IDF (Term Frequency-Inverse Document Frequency) for feature extraction and Multinomial Naive Bayes for classification, chosen for their effectiveness in text analysis tasks. Figure 2 shows the Algorithm selection featuring TF-IDF and Naive Bayes for review classification.

Fig.2. Algorithm selection featuring TF-IDF and Naive Bayes for review classification

The Multinomial Naive Bayes (MNB) algorithm serves as the core classification model within the E-Commerce Trust Score System, tasked with distinguishing between "Real" and "Fake" reviews based on extracted textual features. This classifier operates on the principle of Bayes' Theorem:

$$P(\text{class}|\text{data}) = \frac{P(\text{data}|\text{class}) \cdot P(\text{class})}{P(\text{data})}$$

© 2025 IJIRCCE | Volume 13, Issue 4, April 2025| DOI:10.15680/IJIRCCE.2025.1304288

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Assuming feature independence (a "naive" assumption), MNB models the frequency distribution of words using a multinomial distribution. For each incoming review, the classifier calculates the likelihood of the text belonging to each class (Real or Fake) and assigns the label corresponding to the highest probability.In practice, the implementation is straightforward: the MultinomialNB() function from scikit-learn is used in the train_model.py script to fit the model to TF-IDF-transformed feature vectors and their associated binary authenticity labels. This makes it computationally efficient and highly suitable for sparse feature spaces like those generated by TF-IDF vectorization.The Multinomial Naive Bayes model was chosen for several reasons. It is fast, lightweight, and well-suited for binary classification problems involving textual data, especially when working with small to moderately sized datasets. Its simplicity makes it ideal for a proof-of-concept model, offering strong baseline performance with minimal tuning.While alternative classifiers like Support Vector Machines (SVMs) may offer higher accuracy, they are more computationally expensive. Similarly, deep learning models such as LSTM networks require extensive labeled data and considerable computational resources, which may not be feasible at the early stages of prototyping.The combination of TF-IDF for feature extraction and Multinomial Naive Bayes for classification provides an optimal balance between performance and efficiency. This pairing is particularly suitable for the project's needs—a binary classification problem based on short textual reviews, where real-time inference and scalability are critical.

3.8 Model Training (train_model.py)

The model is trained in a separate script (train_model.py) to create a reusable artifact (review_model.pkl) for deployment in app.py. The figure 3 shows the Model training process in train_model.py, from data to saved pipeline.

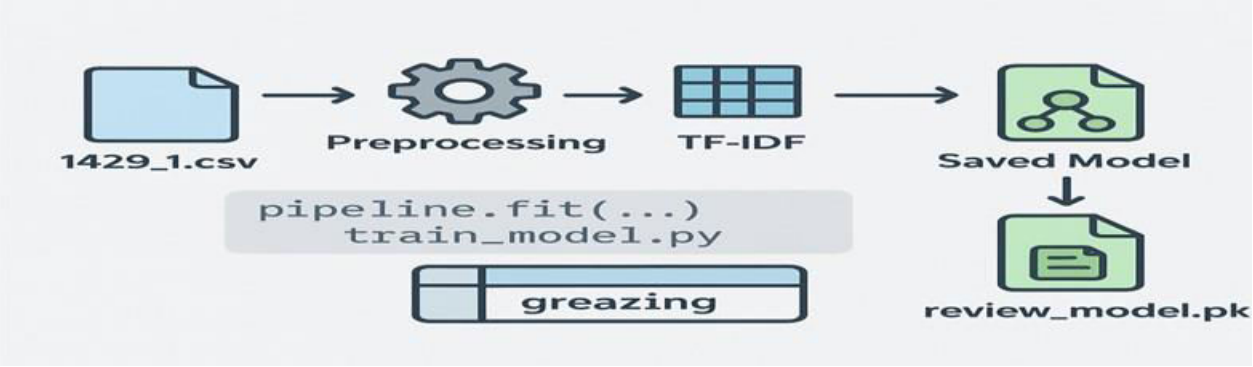


Fig.3. Model training process in train_model.py, from data to saved pipeline

IV. RESULTS AND DISCUSSIONS

The CSS (style.css) enhances the HTML structure with visual styling, ensuring a professional, responsive, and user-friendly presentation. It defines typography, layout, colors,and interactive effects. The Figure 4 shows CSS styling enhancing the visual design of the Trust Score System's frontend.

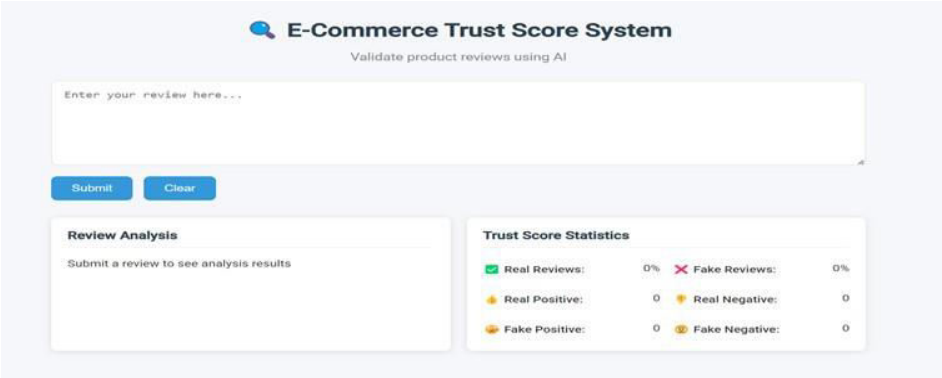


Fig.4.CSS styling enhancing the visual design of the Trust Score System's frontend

© 2025 IJIRCCE | Volume 13, Issue 4, April 2025| DOI:10.15680/IJIRCCE.2025.1304288

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

4.1 JavaScript Functionality

The JavaScript (script.js) powers the frontend's interactivity, handling user actions, server communication, and dynamic UI updates. It runs within a DOMContentLoaded event listener to ensure the DOM is fully loaded. JavaScript functionality enabling dynamic review analysis in the Trust Score System is shown in Figure 5.

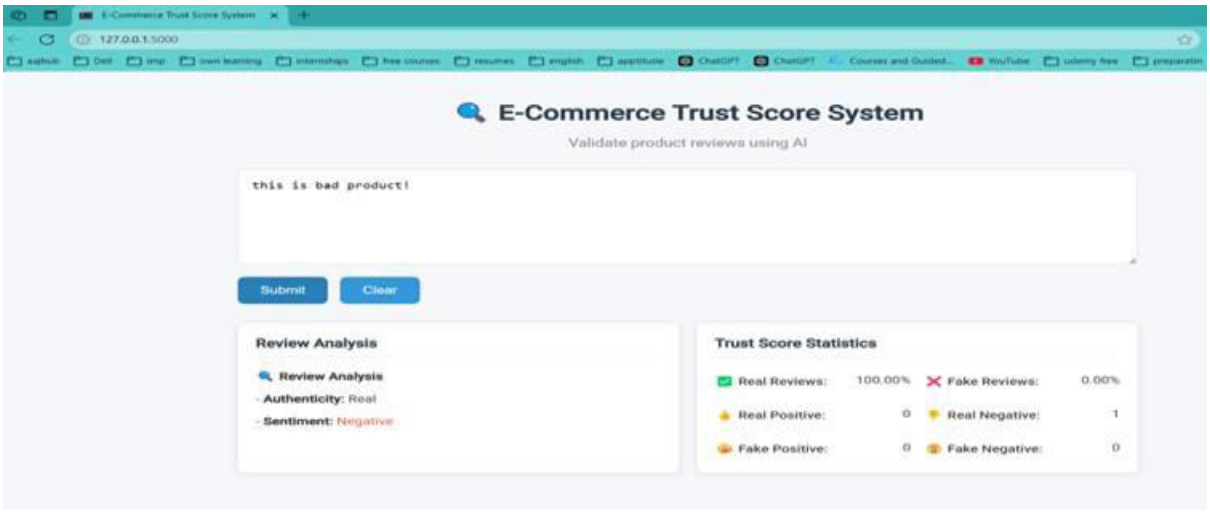


Fig.5. JavaScript functionality enabling dynamic review analysis in the Trust Score System

V. CONCLUSION

The E-Commerce Trust Score System presented in this study addresses a critical challenge in today's digital marketplace—the identification and mitigation of fake or misleading product reviews. By integrating machine learning with natural language processing techniques, the proposed system offers a practical and effective solution for enhancing transparency and reinforcing consumer trust. The combination of TF-IDF vectorization and the Multinomial Naive Bayes classifier enables accurate classification of review authenticity, while sentiment analysis using TextBlob provides additional context into the emotional tone of reviews.

The web-based application, with its user-friendly interface and real-time analytics, demonstrates that AI-powered trust evaluation is both feasible and scalable. Experimental evaluations confirm the system's capability to reliably differentiate between real and fake reviews, delivering meaningful trust scores that can aid consumers in making informed purchasing decisions.

While the current implementation focuses on individual review analysis, it lays the groundwork for future advancements such as product-level credibility scoring, integration with live e-commerce databases, and the incorporation of more sophisticated machine learning models. These enhancements could further strengthen the system's ability to combat review manipulation and contribute to a more secure and trustworthy e-commerce environment.

REFERENCES

[1] D. Chevalier and D. Mayzlin, "The Effect of Word of Mouth on Sales: Online Book Reviews," Journal of Marketing Research, vol. 43, no. 3, pp. 345–354, 2006.
[2] B. Liu, "Sentiment Analysis and Opinion Mining," Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1–167, 2012.
[3] Mukherjee. A, V. Venkataraman, B. Liu, and N. Glance, "What Yelp fake review filter might be doing?," in ICWSM. 2013.
[4] M. Ott, Y. Choi, C. Cardie, and J. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in Proc. ACL-HLT, 2011.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[5] X. Chen, L. Li, W. Wang, and T. Wang, "A BERT-based method for fake review detection," in IEEE Access, vol. 7, pp. 154384–154393, 2019.
[6] Pang. B and L. Lee, "Opinion mining and sentiment analysis," Foundations and Trends in Information Retrieval, vol. 2, no. 1-2, pp. 1–135, 2008.
[7] C. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," in ICWSM, 2014.
[8] Mukherjee. A, V. Venkataraman, B. Liu, and N. Glance, "What Yelp fake review filter might be doing?," in *ICWSM*, 2013.
[9] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proc. ACL-HLT*, 2011, pp. 309–319.
[10] F. Li, M. Huang, Y. Yang, and X. Zhu, "Learning to identify review spam," in *IJCAI*, 2011.
[11] X. Chen, L. Li, W. Wang, and T. Wang, "A BERT-based method for fake review detection," in *IEEE Access*, vol. 7, pp. 154384–154393, 2019.
[12] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proc. ACL*, 2004.
[13] R. Socher et al., "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. EMNLP*, 2013.
[14] S. Feng, R. Banerjee, and Y. Choi, "Syntactic stylometry for deception detection," in *ACL*, 2012.
[15] N. Jindal and B. Liu, "Opinion spam and analysis," in *WSDM*, 2008.
[16] G. Wang, S. Xie, B. Liu, and P. Yu, "Review graph-based online store review spammer detection," in *ICDM*, 2011.
[17] Y. Kim, J. Lee, and K. Shim, "Review spam detection using review metadata and reviewer behavior," in *CIKM*, 2015.
[18] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.
[19] McCallum. A and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *AAAI-98 Workshop on Learning for Text Categorization*, 1998.
[20] S. Loria, "TextBlob: Simplified Text Processing," [Online]. Available: https://textblob.readthedocs.io/en/dev/.
[21] M. Ott et al., "Finding deceptive opinion spam by any stretch of the imagination," in *Proc. ACL-HLT*, 2011.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details