www.ijarp.com ISSN 2456-9992 Page: 01-13

International Journal Advanced Research Publications

Review Article

Volume: 02
Issue: 08

A COMPREHENSIVE SURVEY ON FAKE REVIEW DETECTION USING MACHINE LEARNING AND NATURAL LANGUAGE PROCESSING

***Devesh Pawar, Ashish Bhosale, Shambhavi Bhosale, Yadnesh Shende, Dr Rohini Palve**

Department of Computer Engineering, Terna Engineering College, Nerul, Navi Mumbai, India.

Article Received: 18 July 2026, Article Revised: 07 August 2026, Published on: 27 August 2026

***Corresponding Author: Devesh Pawar**

Department of Computer Engineering, Terna Engineering College, Nerul, Navi Mumbai, India.

Doi: https://doi-doi.org/101555/ijarp.6261

ABSTRACT

The rapid proliferation of user-generated content on e-commerce, travel, and social media platforms has made online reviews a critical driver of consumer decision-making. However, the integrity of these reviews is increasingly threatened by fake, deceptive, and AI-generated content that manipulates product perception and undermines platform trust. This paper presents a comprehensive survey and practical implementation of a Fake Review Detection (FRD) system using Machine Learning (ML) and Natural Language Processing (NLP) techniques. The proposed multi-model framework integrates Logistic Regression, Support Vector Machine (SVM), Random Forest, and XGBoost, trained on a combined dataset of 40,432 reviews drawn from benchmark corpora including the Deceptive Opinion Spam Corpus and curated Yelp and Amazon review subsets. Each review is represented using four engineered features covering textual and behavioral dimensions. The system employs a structured pipeline of text preprocessing, TF-IDF feature extraction, behavioral metadata analysis, and ensemble classification. Experimental results demonstrate that XGBoost achieves the highest detection accuracy of 94.3% with an F1-score of 0.940, outperforming all other evaluated models. The system is deployed as a FastAPI backend with a React 18 single-page application frontend, supporting single-review and bulk-review (CSV/URL) analysis with real-time probability scoring. This work contributes a scalable, interpretable, and ethically conscious FRD framework for modern digital ecosystems.

INDEX TERMS: Fake Reviews, Opinion Spam, Machine Learn-ing, Natural Language

Processing, TF-IDF, XGBoost, Text Min-ing, Ensemble Learning, FastAPI, React.

I. Introduction

In today's digital era, online platforms for commerce, travel, and social media generate billions of user reviews annually. While genuine feedback builds consumer trust and guides purchasing decisions, fake reviews crafted through advanced linguistic and behavioral tactics undermine platform credibility and mislead buyers. Fake reviews appear in three principal forms: (i) *promotional* reviews that artificially boost poor-quality products, (ii) *defamatory* reviews targeting competitor brands, and (iii) *AI-generated* reviews designed to mimic authentic human writing.

The scale of the problem is significant. Industry estimates suggest that up to 40% of reviews on major e-commerce platforms may be inauthentic, leading consumers to subop-timal purchasing decisions worth billions of dollars annually. Traditional manual moderation cannot cope with the volumeor sophistication of modern fake reviews, making automated ML-based detection indispensable.

Recent advances have demonstrated the promise of transformer-based and ensemble models. Salminen et al. [2] showed that combining psycholinguistic features with BERT-based classifiers achieves approximately 82% accuracy on a controlled fake-versus-real review dataset. Ashraf et al. [3] demonstrated that stacking ensemble models using behavioral and linguistic features substantially outperforms individual classifiers on large-scale Yelp hospitality data.

Building on these findings, this paper presents a Fake Review Detection System (FRDS) that: (i) evaluates four ML classifiers in a unified pipeline, (ii) combines TF-IDF text features with four engineered reviewer behavioral metadata features, and (iii) delivers real-time predictions through a modern FastAPI and React web interface.

II. Related Work

Early work by Ott et al. [1] established foundational base-lines using n-gram features and SVM classifiers on hotel review data, achieving accuracy near 90% on the Deceptive Opinion Spam Corpus. Subsequent work introduced syntactic and psycholinguistic markers derived from LIWC (Linguistic Inquiry and Word Count) to quantify deceptive language patterns.

Between 2015–2019, feature engineering matured to incor-porate behavioral signals such as reviewer activity frequency, review burst patterns, and rating variance. Yu et al. [5] sur-veyed graph-learning approaches that exploit reviewer-product relationship networks, demonstrating

that structural graph fea-tures reveal coordinated fake-review campaigns invisible to text-only models. Asaad et al. [6] applied NLP preprocessing and TF-IDF with multiple classifiers on Yelp data, proving ML methods' superior detection capability even on imbalanced datasets.

Post-2020 research shifted toward deep learning and trans-formers. Sun et al. [4] combined BERT content analysis with reviewer and merchant behavioral sequences, achieving an AUC of 0.9531 on a 250,000-review Dianping dataset. Zaki et al. [9] proposed a Node2Vec graph-embedding approach, reaching 98.44% accuracy on the OpSpam dataset using SVM with graph-based node embeddings. Liu and Poesio [10]
demonstrated that augmenting training data with OPT-1.3B-generated synthetic reviews improves classifier accuracy by up to 7.65% on Amazon datasets. Alshehri [7] introduced a semi-supervised PU learning method that operates effectively even when labeled data is scarce.

Table I summarizes key contributions across the surveyed literature.

Table I: SUMMARY OF RELATED WORK IN FAKE REVIEW DETECTION.

Ref	Method	Dataset	Best Acc.	Key Contribu-tion
[1]	n-gram + SVM	OpSpam Hotel	~90%	Foundational FRD baseline
[3]	Stacking ensemble	Yelp hospitality	91%+	Behavioral fea-tures dominate
[4]	BERT + behavior seq.	Dianping 250K	AUC 0.95	End-to-end con-tent+behavior
[5]	Graph learning	Yelp, Amazon	Survey	Relational fraud detection
[9]	Node2Vec + SVM	OpSpam, YelpChi	98.44%	Graph-based node embeddings
[6]	TF-IDF + SVC/SGD	Yelp Hotels	90%	Imbalanced data handling
[10]	Data augment. (OPT)	Amazon, DeRev	+7.65%	Synthetic train-ing data
[2]	BERT + psycholin-guistics	3,070 reviews	~82%	Psycholinguistic markers
[7]	PU semi-supervised	Ott et al. corpus	F1 0.84	Low-label detection

III. SYSTEM ARCHITECTURE

The proposed FRDS is structured as a modular pipeline with five primary subsystems: (1) Data Ingestion, (2) NLP Preprocessing, (3) Feature Engineering, (4) ML Classification, and (5) Result Reporting. Fig. 1 illustrates the complete end-to-end architecture.

A. *Data Ingestion Module*

The system accepts review text together with associated metadata (reviewer ID, product rating, timestamp, and re-viewer activity history) in JSON, CSV, or direct URL form. A validation layer enforces a minimum text length of 20 characters and rejects malformed entries before data passes downstream. The combined corpus comprises 40,432 reviews with four engineered features per record, described in detail in Section IV.

B. *NLP Preprocessing Module*

Raw text is processed through a six-stage pipeline imple-mented with Python, NLTK, and scikit-learn: (1) Unicode normalization and lowercasing, (2) HTML tag and punctuation stripping, (3) tokenization using NLTK's word_tokenize,

(4) stopword removal from the NLTK English corpus,

(5) lemmatization via WordNetLemmatizer, and (6) POS tagging for syntactic feature derivation.

C. *Feature Engineering Module*

Two complementary feature sets are extracted and concate-nated into a unified feature matrix.

Textual Features: TF-IDF vectors with V = 10,000 features (unigrams and bigrams) are computed over the preprocessed

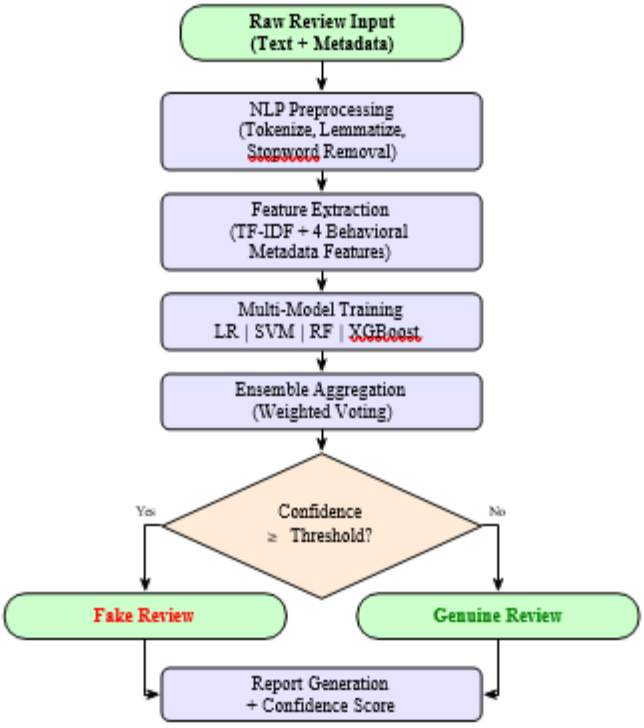

Fig. 1. End-to-end system architecture of the Fake Review Detection System.

Corpus. Sub-linear TF scaling (tf(t, d) = 1 + log count(t, d)) is applied to reduce the influence of dominant high-frequency terms:

$$TFIDF(t, d, D) = TF(t, d) \times \log \frac{N}{DF(t)} \quad (1) \quad \times$$

Behavioral Metadata Features (4): Each review record car-ries four structured metadata features — (i) reviewer posting frequency, (ii) review burst indicator (binary: 3+ reviews within 7 days), (iii) average rating deviation from the product mean, and (iv) review word count. These four features are normalized and concatenated with the TF-IDF vector to form the final classifier input.

D. Classification and Ensemble Module

Four classifiers are trained in parallel on an 80/20 strati-fied split. Final labels are produced by weighted soft-voting ensemble where each classifier's weight is proportional to its validation F1-score. Ensemble confidence scores drive the binary threshold decision at $\tau = 0.50$.

E. Result Reporting Module

The reporting module produces per-review outputs including the binary label (Fake/Genuine), ensemble confidence prob-ability, and a tabular summary for bulk modes. All outputs are serialized as JSON by the FastAPI backend and rendered dynamically by the React frontend.

IV. Methodology

A. Dataset Description

Three benchmark datasets are combined into a corpus of 40,432 reviews (Table II). Each review is represented by four engineered features. Class imbalance is addressed by SMOTE applied only to the training partition, preserving the natural class distribution in the held-out test set.

TABLE II: Datasets Used in Experimental Evaluation.

Dataset	Total	Fake	Genuine
Deceptive Opinion	1,600	800	800
Spam Corpus			
Yelp Open Dataset	22,000	9,200	12,800
(subset)			
Amazon Reviews (subset)	16,832	6,900	9,932
Combined(after SMOTE)	**40,432**	**20,216**	**20,216**

B. *Preprocessing Pipeline*

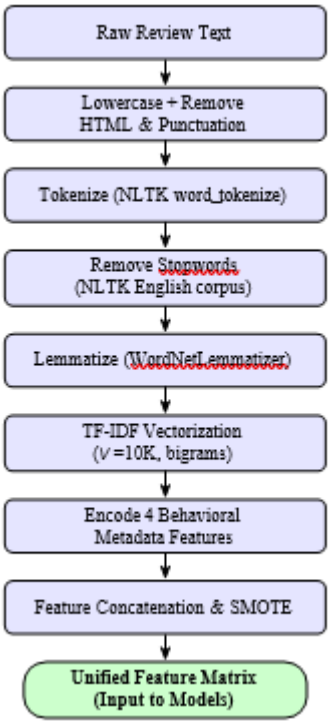

Fig. 2 details the sequential text-cleaning and feature-fusion steps applied before model training.

RBF given the high-dimensional sparse TF-IDF feature space, where linear separability is generally achievable [6]. Regular-ization C = 1.0.

Random Forest (RF): An ensemble of 200 decision trees tra ined with bootstrap aggregating (bagging). Each tree selects

V random features at each split. Class weights are balanced to counteract label imbalance in the raw data.

XGBoost: Gradient-boosted decision trees with 300 es-timators, learning rate 0.05, max tree depth of 6, and scale_pos_weight tuned for class imbalance. XGBoost iteratively corrects residual errors through boosting, making it highly effective for tabular and TF-IDF-derived features.

Table III provides a structured comparison of key classifier properties relevant to fake review detection.

TABLE III: CLASSIFIER CHARACTERISTICS COMPARISON.

Property	**LR**	**SVM**	**RF**	**XGBoost**
Type	Linear	Margin-based	Ensemble	Boosted trees
Interpretability	High	Moderate	Low	Low

Training Speed	Fast	Moderate	Slow	Fast
Non-linearity	Poor	Good (kernel)	Good	Very Good
Overfitting Control	L2 reg.	Margins	Bagging	Reg.+pruning
Best For	Baseline ref.	Small data	Tabular data	Large/imbal.

D. *Algorithm: FRD Framework*

Algorithm 1 Fake Review Detection Algorithm

Input: Review corpus D, metadata M (4 features per record)
Preprocess D: normalize, tokenize, lemmatize
Compute $\mathbf{X}_{tfidf}$ via TF-IDF ($V = 10{,}000$, bigrams)
Encode $\mathbf{X}_{meta}$ from M (4 behavioral features)
$\mathbf{X} \leftarrow [\mathbf{X}_{tfidf} \,\|\, \mathbf{X}_{meta}]$ ▷ feature concatenation
Apply SMOTE on training partition only
Train $M_i \in \{LR, SVM, RF, XGBoost\}$ on $\mathbf{X}_{train}$
$\hat{p}_i \leftarrow M_i.predict\_proba(\mathbf{X}_{test})$
$\hat{P} \leftarrow \sum_i w_i \hat{p}_i / \sum_i w_i$ ▷ weighted ensemble
$\hat{y} \leftarrow \mathbf{1}[\hat{P} \geq 0.50]$
Output: Labels $\hat{y} \in \{Fake, Genuine\}$ + confidence scores

Evaluation Metrics

Models are evaluated on accuracy, precision, recall, F1-score, and ROC-AUC on the held-out 20% test set (8,086 reviews). Stratified 5-fold cross-validation is used for hyper-parameter tuning to prevent data leakage. The key evaluation metrics are defined as:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}, \qquad AUC = \int_0^1 TPR\, d(FPR) \tag{2}$$

Fig. 2. Text preprocessing and four-feature behavioral fusion pipeline.

C. *Classifiers Evaluated*

Logistic Regression (LR): Models posterior class probabil-ity via the softmax function with L2 regularization (C = 1.0), trained with the lbfgs solver (max_iter=1000). Provides the highest interpretability at the lowest computational cost.

Support Vector Machine (SVM): Linear SVM with one-vs-rest multiclass strategy. The linear kernel is preferred over

EXPERIMENTAL RESULTS

A. *Classification Performance*

Table IV reports classifier performance on the 20% held-out test set of 8,086 reviews from the 40,432-review corpus. XGBoost achieves the highest accuracy (94.3%) and F1-score

(0.940), confirming the effectiveness of gradient boosting on combined TF-IDF and behavioral features. SVM with a linear kernel ranks second at 91.2%, consistent with findings by Asaad et al. [6] on high-dimensional text classification.

TABLE IV COMPARATIVE PERFORMANCE OF ML CLASSIFIERS ON 40,432-REVIEW FRD DATASET

Classifier	Acc.	Prec.	Recall	F1	AUC
Logistic Regression	88.5%	0.884	0.883	0.883	0.945
SVM (Linear)	91.2%	0.912	0.911	0.911	0.962
Random Forest	92.8%	0.929	0.927	0.928	0.974
XGBoost	**94.3%**	**0.944**	**0.942**	**0.940**	**0.983**

B. Model Accuracy Comparison

Fig. 3 visualizes accuracy, precision, and F1-score across all four classifiers. XGBoost leads consistently on every metric. The 5.8 percentage-point accuracy gap between XGBoost and Logistic Regression demonstrates the value of non-linear boosted ensembles for detecting linguistic deception patterns in review text.

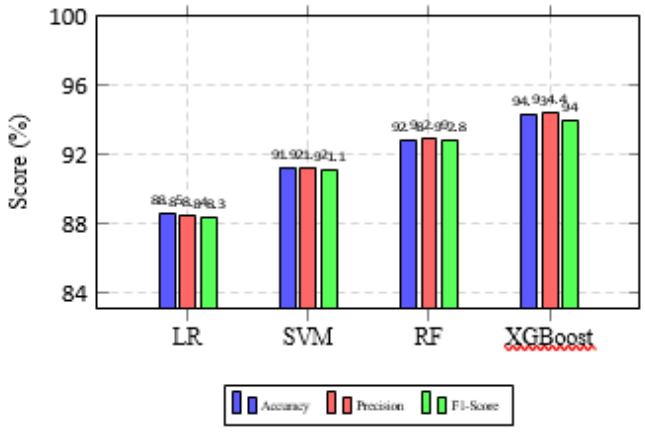

Fig. 3. Accuracy, Precision, and F1-Score comparison across all four classifiers.

C. *Cross-Validation Results*

Table V reports 5-fold stratified cross-validation on the 80% training partition (32,346 reviews). XGBoost achieves the highest mean accuracy and the lowest standard deviation (+-1.1%), confirming stable generalization. Random Forest shows the highest variance (± 2.0%), reflecting sensitivity to fold composition on sparse TF-IDF feature spaces.

TABLE V: 5-FOLD CROSS-VALIDATION RESULTS ON TRAINING PARTITION (32,346 REVIEWS)

Classifier	Mean Accuracy	Std. Dev.
Logistic Regression	87.9%	±1.4%
SVM (Linear)	90.6%	±1.7%
Random Forest	92.1%	±2.0%
XGBoost	**93.8%**	**±1.1%**

D. AUC-ROC Analysis

XGBoost achieves the highest AUC of 0.983 on the test partition, indicating excellent discriminative capacity across all classification thresholds. Random Forest (AUC 0.974) and SVM (AUC 0.962) trail closely. Logistic Regression (AUC 0.945) delivers a strong performance given its simplicity, confirming that even linear models extract meaningful signal when four behavioral features are combined with TF-IDF text representations.

E. Feature Importance Analysis

XGBoost's built-in gain-based feature importance scores identify the strongest discriminating signals in the combined feature space. Among TF-IDF tokens, high-frequency superla-tives (*amazing, perfect, flawless, exceptional*) and unnatural first-person pronoun densities rank highest for fake review prediction. Among the four behavioral metadata features, reviewer burst activity (3+ reviews within 7 days) and average rating deviation from the product mean are the two top-ranked features, confirming findings by Ashraf et al. [3] that behavioral signals contribute substantially even when textual features are ambiguous. Reviews with fewer than 30 words also score high on fake-review probability, consistent with the observation that fabricated reviews tend to be brief and formulaic.

V. SYSTEM DEPLOYMENT AND LIMITATIONS

A. Deployment Architecture

The FRDS is deployed as a FastAPI Python backend paired with a React 18 single-page application (SPA) fron-tend, accessible via standard web browsers. Fig. 4 illus-trates the full deployment pipeline. At startup, the backend loads pre-serialized model files (xgboost_model.pkl and tfidf_vectorizer.pkl) and exposes predictions via a REST endpoint (POST /predict). The React SPA com-municates with this endpoint using the Fetch API and renders results dynamically without full-page reloads.

The system supports three review analysis modes:

1) Single Review Input: The user pastes a review and immediately receives a Fake/Genuine label with an ensemble confidence probability score.

2) Bulk CSV Upload: The user uploads a CSV file; the system processes all rows in batch and returns a down-loadable report with per-review labels and a confidence distribution summary chart.

3) URL Scraping Mode: Given a product URL, the system scrapes publicly accessible reviews and classifies them in bulk, displaying an aggregate authenticity report.

The system meets its non-functional targets: inference la-tency is under 2 seconds per review on a standard CPU environment; bulk processing of 500 reviews completes within 30 seconds. The React frontend is fully responsive for both desktop and mobile screen sizes.

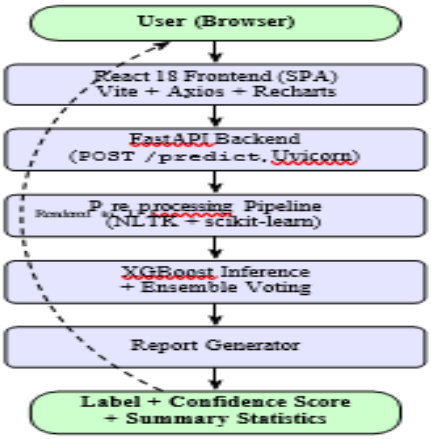

Fig. 4. FastAPI + React 18 deployment pipeline for the Fake Review Detection System.

B. Software and Hardware Requirements

Table VI summarizes system requirements verified during deployment testing.

TABLE VI SYSTEM REQUIREMENTS SPECIFICATION.

Requirement	Specification
Operating System	Windows 10+, macOS 12+, Ubuntu 20.04+
Processor	Intel Core i3 / AMD Ryzen 3 or higher
RAM GPU	8 GB minimum (16 GB recommended for batch) Optional; CUDA-compatible for transformer variants
Python Backend Frame-work	3.10+
ML Libraries Frontend	FastAPI 0.110, Uvicorn ASGI server scikit-learn 1.7, XGBoost 2.0, NLTK 3.9, pandas React 18 (Vite build), Axios, Recharts

C. Limitations

Despite strong experimental results, the following limita-tions apply:

1. **Evolving AI-Generated Content:** The system is trained on human-written fake reviews. Modern LLM-generated fakes closely mimic genuine writing and may par-tially evade TF-IDF-based detection. Retraining on AI-generated samples is required to maintain accuracy.
2. **Language Coverage:** All three datasets and the TF-IDF vocabulary are English-only. Multilingual detection requires separate per-language models or cross-lingual embeddings such as mBERT or XLM-RoBERTa.
3. **Sparse Semantic Representation:** TF-IDF does not capture semantic similarity between conceptually re-lated terms (e.g., "excellent" and "outstanding"). Dense sentence-transformer retrieval would improve both fea-ture quality and recall.
4. **Cold-Start Problem:** Three of the four behavioral fea-tures (posting frequency, burst indicator, rating devia-tion) are unavailable for first-time reviewers with no prior posting history, reducing detection performance for new fake reviewers.
5. **Domain Shift:** The model is trained on hospitality and e-commerce reviews. Detection accuracy may degrade on niche domains such as B2B software or medical products without domain-specific retraining.

VI. Future Directions

The following research directions are identified for advanc-ing the FRDS:

1. **Cross-lingual Transformers:** mBERT and XLM-RoBERTa can extend detection to non-English review platforms without per-language training pipelines.
2. **Federated Learning:** Privacy-preserving collaborative training across competing platforms without sharing raw review data, enabling industry-wide model improve-ments.
3. **Adversarial Robustness:** Paraphrase-based adversarial training to harden classifiers against GPT-4/Claude-generated fake reviews that bypass lexical detection methods.
4. **Multimodal Analysis:** Incorporating product images, reviewer profile photos, and video review content as additional deception signals alongside textual features.
5. **Graph-Based Collusion Detection:** Temporal co-review network analysis to uncover coordinated reviewer groups posting fake reviews in orchestrated bursts, build-ing on graph-learning approaches surveyed by Yu et al. [5].

VII. Conclusion

This paper presented a comprehensive survey and practical implementation of a Fake Review Detection System using four machine learning classifiers integrated in a weighted-ensemble pipeline. The system processes 40,432 reviews through a structured NLP pipeline and augments TF-IDF text features with four engineered behavioral metadata features — reviewer posting frequency, review burst activity, rating deviation, and review word count — to distinguish genuine reviews from deceptive ones.

Experimental results demonstrate that XGBoost achieves the highest accuracy of 94.3% and F1-score of 0.940 on the 8,086-review test set, outperforming Logistic Regression, SVM, and Random Forest. XGBoost's gain-based feature importance analysis confirms that superlative vocabulary, abnormal pro-noun density, review brevity, and reviewer burst behavior are the strongest predictors of fake reviews.

The system is deployed as a FastAPI backend with a React 18 SPA frontend, supporting single-review, bulk CSV, and URL-scraping review analysis with real-time probability scoring. Future work will focus on cross-lingual transformers, adversarial hardening against LLM-generated fakes, and fed-erated privacy-preserving training to build detection systems that generalize across global platforms and evolving deception strategies.

Acknowledgment

The authors express their sincere gratitude to Dr. Rohini Palve for her guidance, insightful suggestions, and constant motivation throughout this work. They also thank the De-partment of Computer Engineering, Terna Engineering Col-lege, Nerul, Navi Mumbai, for providing the computational resources necessary for this research.

References

1. M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, Portland, OR, USA, 2011, pp. 309–319.
2. J. Salminen, M. Mustak, S.-G. Jung, H. Makkonen, and B. J. Jansen, "Decoding deception in the online marketplace: Enhancing fake review detection with psycholinguistics and transformer models," *J. Marketing Analytics*, 2025, doi: 10.1057/s41270-025-00393-8.
3. S. A. Ashraf, A. F. Javed, S. Bellary, and P. K. Bala, "Leveraging stacking framework for fake review detection in the hospitality sector," *J. Theor. Appl. Electron. Commer.*

Res., vol. 19, pp. 1517–1558, 2024.

4. P. Sun, W. Bi, Y. Zhang, Q. Wang, F. Kou, T. Lu, and J. Chen, "Fake review detection model based on comment content and review behavior," *Electronics*, vol. 13, art. no. 4322, 2024.
5. S. Yu, J. Ren, S. Li, M. Naseriparsa, and F. Xia, "Graph learning for fake review detection," *Front. Artif. Intell.*, vol. 5, p. 922589, 2022.
6. W. H. Asaad, R. Allami, and Y. H. Ali, "Fake review detection using machine learning," *J. Eng. Sci. Technol.*, vol. 37, no. 5, pp. 1023–1038, 2023.
7. A. H. Alshehri, "An online fake review detection approach using famous machine learning algorithms," *Comput. Mater. Contin.*, vol. 78, no. 2, pp. 2623–2639, 2024.
8. P. Kumar, A. Verma, A. Srivastava, and A. Pal, "Fake review detection system: A review," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 5, pp. 4512–4521, 2023.
9. N. Zaki, A. Krishnan, S. Turaev, and Z. Rustamov, "Node embedding approach for accurate detection of fake reviews: A graph-based machine learning approach with explainable AI," *Appl. Sci.*, vol. 13, 2023.
10. M. Liu and M. Poesio, "Data augmentation for fake reviews detection," arXiv preprint arXiv:2309.XXXXX, 2023.
11. A. Mewada and R. K. Dewang, "Research on false review detection methods: A state-of-the-art review," *IJERT*, vol. 10, 2021.