International Journal of Engineering Research and Development
e- ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 22, Issue 11 (November 2026), PP 124-128

Verifai: AI Review Intelligence and Trust Analyzer for Fake Review Detection Using Machine Learning

Soham Suthar, Yash Patel, Yagnik Gajera
Department of Computer Science and Engineering, Parul University, Vadodara, Gujarat, India

ABSTRACT
Online reviews influence purchasing decisions but can also contain misleading or suspicious content. This paper presents Verifai, a review intelligence system that extends a fake-review classifier with supporting, interpretable signals. Its authenticity model uses TF-IDF features with unigrams and bigrams and Logistic Regression, selected after comparison with Multinomial Naive Bayes and Random Forest using five-fold cross-validation on weighted F1. The labelled corpus was cleaned from 381,734 raw entries to 28,530 reviews and divided into training and held-out sets, with 5,706 reviews reserved for testing. On that test set, the confusion matrix yields 72.59% accuracy, 72.49% fake-class precision, 73.45% fake-class recall, and 72.97% fake-class F1; ROC-AUC was reported as 80.59%. Beyond classification, Verifai analyzes suspicious writing patterns, sentiment, exact duplicates, rating-text consistency, and product aspects, then presents an indicative 0-100 reliability score. A Flask API serves a React and TypeScript interface; all 20 reported application regression cases passed. The score is a decision support signal, not proof that a review or seller is fraudulent.
Keywords: *Fake review detection, natural language processing, machine learning, TF-IDF, Logistic Regression, sentiment analysis, review intelligence.*

Date of Submission: 02-11-2026 Date of acceptance: 13-11-2026

I. INTRODUCTION

Online reviews function as social proof in e-commerce, yet fabricated praise can inflate a product's standing and fabricated criticism can harm a competitor. Because deceptive text may imitate ordinary customer language, manual review at scale is difficult. Early work framed opinion spam as a supervised classification task [1], while deceptive hotel-review research showed that linguistic cues can support automated detection [2]. Later approaches considered reviewer behavior, sentiment, and relationships among users, products, and reviews [3-7]. Such behavioral and network signals require metadata that may not be available when a system receives review text alone.

Verifai was redesigned from a single-label Fake Review Detector into a review intelligence and trust analyzer. It pairs a supervised text classifier with signals that users can inspect: suspicious writing patterns, sentiment, exact-duplicate detection, rating-text consistency, and aspect-level summaries. The current implementation uses a lightweight TF-IDF and Logistic Regression pipeline and does not claim live access to marketplace review feeds or reviewer histories.

The design reflects a practical constraint in review analysis: language alone rarely settles whether a review is authentic, while behavioral and graph-based methods depend on platform metadata. Verifai therefore reports the model verdict with related text signals and keeps the product-level reliability score explicitly indicative. Its contributions are a cross-validated TF-IDF classifier, an explainable review-intelligence layer, and a full-stack implementation checked with a functional regression suite.

II. OBJECTIVES

This study aims to select a practical authenticity classifier by comparing Logistic Regression, Multinomial Naive Bayes, and Random Forest under five-fold cross-validation. It also aims to extend a single Fake or Genuine prediction with inspectable signals for writing patterns, sentiment, exact duplicates, rating-text consistency, and product aspects. The final objective is to integrate these capabilities into a usable review and batch analysis application and to evaluate predictive performance separately from application behavior.

III. STUDY SCOPE

Verifai is a review-centric software system rather than a geographically bounded field study. A user can submit one review or a batch, with an optional star rating. The application returns a Fake or Genuine indication, confidence information, supporting signals, and a batch summary. The reported build includes a

Python and Flask backend with a React and TypeScript frontend. The paper does not claim live URL-based scraping, database-backed authentication, or production-scale deployment.

IV. DATA SOURCES

The authenticity classifier was developed from a labelled review corpus. The source paper reports 381,734 raw entries before removing duplicates, empty records, and excessively short reviews. The remaining 28,530 reviews comprise 14,162 labelled Genuine and 14,368 labelled Fake examples. A stratified split reserved 5,706 reviews for held-out testing; the remaining 22,824 formed the training portion. The source paper does not name the dataset provider or platforms and does not describe how labels were assigned. These details are therefore not inferred here.

Table 1: Dataset composition and split.

Dataset item	Reported value	Use in the study
Raw review entries	381,734	Starting corpus before cleaning
Clean labelled corpus	28,530	Model-development dataset after cleaning
Genuine / Fake labels	14,162 / 14,368	Approximately balanced binary classes
Held-out test set	5,706	Stratified evaluation set
Training portion	22,824	Model fitting and candidate comparison

Data availability note: *platform provenance, language coverage, sampling method, and label-generation protocol are not stated in the source manuscript. Their addition would improve reproducibility and help interpret cross-platform generalization.*

V. METHODOLOGY

A. Data preparation

Duplicate, empty, and very short entries were removed before modeling. The final dataset was split stratifiably so that the Genuine and Fake classes remained proportionally represented in the held-out test set. The source reports a 5,706-review test set from 28,530 cleaned reviews.

B. Text preprocessing and feature extraction

Review text was lowercased, cleaned of markup and extraneous punctuation, and tokenized. Two TF-IDF representations were considered: unigrams alone and unigrams plus bigrams. The combined representation was selected based on cross-validation performance. A separate rule-based feature set supports the review-intelligence layer and includes text-length measures, punctuation and capitalization ratios, sentiment polarity, normalized duplicate keys, and rating-to-sentiment comparison.

C. Candidate model selection

Logistic Regression, Multinomial Naive Bayes, and Random Forest were compared with the optimized TF-IDF features using five-fold cross-validation and mean weighted F1. Logistic Regression achieved the highest weighted F1 among the candidates and was retained. The fitted vectorizer, classifier, and training metadata are persisted as artifacts for inference by the Flask backend. Fold-level scores were not reported in the source paper.

V. METHODOLOGY (CONTINUED)

D. Authenticity model and inference

For a TF-IDF vector x, the Logistic Regression model maps a learned linear score to a probability for the Fake class:

$$P(Fake \mid x) = 1 / (1 + \exp(-(w^T x + b)))$$

where w is the learned feature-weight vector and b is the bias. At inference, the backend cleans and validates submitted text, applies the persisted vectorizer and classifier, and returns a class indication with confidence information. The paper reports evaluation with accuracy, precision, recall, F1, and ROC-AUC; the held-out results are summarized in Section VI.

E. Review-intelligence signals and aggregation

The supporting layer computes suspicious-pattern indicators from simple writing features, estimates sentiment, detects exact repeats after normalization, and checks whether the text sentiment conflicts with an optional star rating. Aspect analysis groups supported product themes such as Delivery, Performance, Quality, Price, Packaging, and Durability, then summarizes mention counts, dominant sentiment, and representative

keywords. For a batch, the dashboard aggregates review counts, Genuine and suspicious indications, duplicate-involved reviews, sentiment distribution, and aspect summaries.

These signals are presented as evidence around the classifier rather than as independent proof. The 0-100 reliability score aggregates available signals and is explicitly indicative; the source paper does not describe it as a calibrated probability.

F. Application architecture

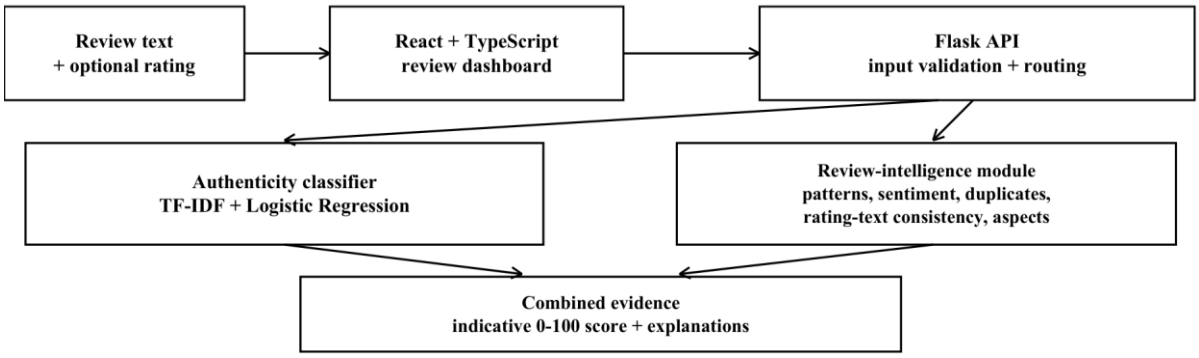


Fig. 1: Verifai processing architecture.

A React and TypeScript interface sends review text and an optional rating to a Flask API. The API runs the authenticity model and review-intelligence module, then returns a combined response for dashboard presentation.

The application uses Python, Flask, scikit-learn, Pandas, NumPy, and Joblib on the backend and React, TypeScript, HTML5, and CSS3 on the frontend. The reported implementation runs on standard CPU hardware; no GPU is required for the current TF-IDF classifier.

Table 2: Minimum and recommended local execution environment.

Component	Minimum	Recommended
Processor	Dual-core, 2.0 GHz	Quad-core, 2.5 GHz or higher
Memory	4 GB RAM	8-16 GB RAM
Storage	10 GB free space	20 GB or more on SSD
Operating system	Windows 10 or Ubuntu 20.04	Windows 11 or Ubuntu 22.04; current browser

VI. RESULTS

The selected Logistic Regression classifier was evaluated on the 5,706-review held-out test set. The source paper reports 72.59% accuracy and 80.59% ROC-AUC. Its confusion matrix contains 2,111 true Fake detections, 801 false Fake indications, 763 missed Fake reviews, and 2,031 correctly identified Genuine reviews.

Table 3: Held-out test metrics.

Metric	Value	Basis
Accuracy	72.59%	Computed from all four confusion-matrix cells
Precision (Fake class)	72.49%	2,111 / (2,111 + 801)
Recall (Fake class)	73.45%	2,111 / (2,111 + 763)
F1-score (Fake class)	72.97%	Recomputed from Fake precision and recall
Specificity (Genuine class)	71.72%	2,031 / (2,031 + 801)
ROC-AUC	80.59%	Reported in the source paper

Metric reconciliation: *the source paper's summary table lists the same 72.59% for accuracy, precision, recall, and F1, while its sensitivity value also conflicts with the confusion matrix. Class-wise metrics above are recalculated from the published counts with Fake as the positive class; ROC-AUC is retained as reported.*

Table 4: Confusion matrix on the held-out test set.

Actual class	Predicted Genuine	Predicted Fake
Genuine	2,031 (TN)	801 (FP)
Fake	763 (FN)	2,111 (TP)

Application-level verification

The integrated application was checked against a 20-case regression suite, and all reported cases passed. Coverage included ordinary and suspicious text, empty input, rating-text combinations, batch and duplicate handling, aspect analysis, review filters, history save and restore, and backend error handling. These checks verify application behavior; they do not establish classifier accuracy beyond the separate held-out evaluation.

In one reported seven-review batch run, the interface showed six Genuine indications, one suspicious indication, two duplicate-involved reviews (29%), and an indicative reliability score of 81/100. This is an illustrative application output, not a general performance estimate.

Outputs surfaced to users

Table 5: Verifai signals and outputs.

Signal	Displayed information
Authenticity	Fake or Genuine indication and confidence
Suspicious patterns	Triggered writing-pattern indicators
Sentiment and consistency	Text tone and optional rating-text alignment
Duplicates and aspects	Exact-duplicate counts; theme mentions, sentiment, and keywords
Batch reliability	Review counts, distributions, and indicative 0-100 score

VII. DISCUSSION

The held-out results indicate useful but imperfect separation between the two labels: the reported ROC-AUC is 80.59%, while the confusion matrix shows both false positive and false negative errors. These results support treating the classifier as a triage signal rather than a definitive verdict. Verifai's layered presentation is valuable because users can inspect duplicate, sentiment, writing, rating-consistency, and aspect evidence alongside the model output.

The model-selection procedure compares three conventional classifiers under five-fold cross-validation, but the source paper does not provide per-fold scores or an external validation set. Results should be interpreted within the reported corpus and split. The lack of dataset provenance and label-generation details limits independent reproduction and makes it difficult to assess possible class or platform bias.

A. Technical and methodological limitations

TF-IDF with Logistic Regression captures useful lexical patterns but has limited contextual understanding compared with contextual language models. Performance may change across platforms, domains, languages, or review styles. Exact-duplicate detection does not identify paraphrases or coordinated campaigns. Rule-based suspiciousness indicators can flag legitimate writing, rating-text checks need a supplied rating, and aspect analysis is limited to the themes and keywords implemented in the system.

The 0-100 reliability score is an aggregation of available signals, not a calibrated probability that a review, product, or seller is genuine. Likewise, a passed application regression suite demonstrates expected software behavior, not the correctness of each authenticity prediction. The current build does not claim live review scraping, database-backed authentication, or production-scale marketplace use.

B. Responsible interpretation

A false Fake indication can unfairly cast doubt on an authentic reviewer, while a missed deceptive review can preserve misleading content. The interface should therefore support human inspection and further verification. The source paper frames the reliability score as indicative, and that distinction should remain clear in any deployment.

C. Future work

Further evaluation should document the corpus source and label protocol, test across independent platforms and languages, and report class-wise metrics and calibration. Future versions could compare BERT or RoBERTa with the current baseline, detect near-duplicates and coordinated behavior when reliable metadata is available, add feature-level explanations, and create a retraining and monitoring process. Verified review APIs could be considered subject to platform access. The proposed customer-evidence layer described in the source paper could provide an additional transparent signal if participation is voluntary and consent-based.

VIII. CONCLUSION

This paper presents Verifai, a review intelligence system that combines a TF-IDF and Logistic Regression authenticity classifier with interpretable signals for suspicious writing, sentiment, exact duplication, rating-text consistency, and product aspects. The held-out confusion matrix corresponds to 72.59% accuracy, 72.49% Fake-class precision, 73.45% Fake-class recall, and 72.97% Fake-class F1, with an 80.59% ROC-AUC reported by the source paper. All 20 reported application regression cases passed. Together, these results show a lightweight, CPU-efficient decision-support design, while its dataset provenance, label process, cross-platform generalization, and score calibration remain open issues. Verifai's output is best interpreted as structured evidence for review rather than as a conclusive fraud judgment.

REFERENCES

[1]. N. Jindal and B. Liu, "Opinion Spam and Analysis," in *Proceedings of the First ACM International Conference on Web Search and Data Mining*, pp. 219-230, 2008.

[2]. M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding Deceptive Opinion Spam by Any Stretch of the Imagination," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, Portland, Oregon, pp. 309-319, 2011.

[3]. A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "What Yelp Fake Review Filter Might Be Doing," in *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, 2013.

[4]. J. K. Rout, A. Dalmia, K. K. Choo, S. Bakshi, and S. K. Jena, "Revisiting Semi-Supervised Learning for Online Deceptive Review Detection," *IEEE Access*, vol. 5, pp. 1319-1327, 2017.

[5]. J. Fontanarava, G. Pasi, and M. Viviani, "Feature Analysis for Fake Review Detection through Supervised Classification," in *Proceedings of the IEEE International Conference on Data Science and Advanced Analytics*, pp. 658-666, 2017.

[6]. S. Rayana and L. Akoglu, "Collective Opinion Spam Detection: Bridging Review Networks and Metadata," in *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 985-994, 2015.

[7]. G. Wang, S. Xie, B. Liu, and P. S. Yu, "Review Graph Based Online Store Review Spammer Detection," in *Proceedings of the 11th IEEE International Conference on Data Mining*, pp. 1242-1247, 2012.

[8]. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.