Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 03 (2026)

Design of an Intelligent Model Selection Framework for Unified Review Authenticity Detection with Real-Time E-Commerce Integration

Swijel Dmello[1]*, Dr. Nilesh Deotale[2]

1Student, Department of Computer Engineering, St. John College of Engineering & Management (SJCEM), Palghar, Mumbai-401404, India.

2Professor, Department of Computer Engineering, St. John College of Engineering & Management (SJCEM), Palghar, Mumbai-401404, India.

Abstract: Objectives: Develop a machine learning system with online classification ability for detecting fake reviews of hotels using a web-based application. **Methods**: The analysis was done on 11,912 samples of hotel reviews. Preprocessing of the data set and extraction of the features were done using the TF-IDF method, after which seven machine learning algorithms including Decision Tree, K-Nearest Neighbor (KNN), Gaussian Naive Bayes, Multinomial Naive Bayes, Logistic Regression, Stochastic Gradient Descent, and Support Vector Machine (SVM) were trained using various kernels. The most efficient model was implemented using API on a demo hotel website to provide real-time prediction. **Findings**: An SVM model using Radial Basis Function (RBF) kernel yielded the highest prediction accuracy of 88% among the evaluated classifiers such as Decision Tree (63%), KNN (73%), Gaussian Naive Bayes (74%), Multinomial Naive Bayes (79%), Logistic Regression (81%) and SGD (83%). It can be inferred from the results that the RBF kernel is very effective in handling non-linearity present in the data set. Contrary to offline models, the proposed system provides real-time detection of fake hotel reviews using API. **Novelty**: This research has presented a unified framework that involves multi-model comparison, SVM kernel optimization, and real-time API-based deployment, enabling the practical implementation of machine learning algorithms for fake reviews detection applicable in online applications.

Keywords: Fake Reviews, Machine Learning, Support Vector Machine, Text Classification, Real-Time Detection.

1. Introduction

Online reviews have emerged as a dominant driver in decision-making processes in recent times, not only in areas like e-commerce but also in hospitality and many others. Consumers rely heavily on these ratings to decide whether to buy something or not. Nevertheless, the sudden increase in the number of user-generated content has also witnessed an increase in the number of deceptive or fraudulent reviews [1,2,3]. Fake reviews are intentionally generated to deceive the consumer and benefit certain business establishments at the expense of others. There has been a pressing need for detecting such reviews effectively.

The latest advances in the detection of fake reviews have used multiple approaches in the process. Linguistic features analysis, behavioral pattern extraction, machine learning, and neural network-based solutions, including transformers, have been applied to tackle this problem [4,5,6]. Recent studies have further highlighted the challenge posed by AI-generated reviews. Additionally, advanced multimodal frameworks that integrate textual and visual features have been proposed to enhance detection accuracy[7,8]. Promising results have been achieved in this regard, especially in terms of classification precision and detection rate [9]. Sentiment analysis models and feature engineering have played an instrumental role in detecting deception patterns in the text, whereas deep learning has helped improve detection on larger scale datasets [1,10,11].

However, certain constraints continue to exist despite all these improvements. Existing literature concentrates more on the offline analysis of models based on benchmark data without any application in actual time situations. Additionally, many approaches emphasize either complex models or feature extraction techniques without addressing deployment challenges[12]. There is also limited work that performs a unified comparison of multiple traditional machine learning algorithms within the same framework, especially with detailed analysis of Support Vector Machine (SVM) kernel behavior for text classification tasks [8,13,14]. As a result, a gap exists between high-performing models and their practical implementation in real-world environments.

From Table 1, it is evident that existing studies primarily focus on model accuracy and feature engineering, while lacking real-time deployment, system integration, and user interaction. The present research addresses these gaps through a unified, deployable framework.

Table 1. Comparison between selected recent studies to demonstrate the research gaps that exist

Metric	[5] (2020)	[1](2022)	[14](2023)	[2](2024)	[12](2025)	Present Research
Multi-Algorithm Comparison	Yes	Yes	Yes	Partial	Partial	Yes
Traditional ML Models	Yes	Yes	Yes	Yes	No	Yes
Deep Learning Models	No	No	No	Yes	Yes	No
Feature Engineering	Partial	Partial	Partial	No	Yes	Yes
SVM Kernel Analysis	No	No	No	No	No	Yes
Real-Time Deployment	No	No	No	Partial	No	Yes
API Integration	No	No	No	No	No	Yes
Web-Based System	No	No	No	No	No	Yes
Interpretability	Low	Moderate	Low	Moderate	Moderate	Moderate
Practical Usability	Low	Moderate	Moderate	Moderate	Low	High

In order to deal with these problems, the proposed approach in this research work involves a framework for detecting fake reviews using machine learning. In this context, a set of reviews, consisting of 11,912 hotel reviews is used and different kinds of classifiers, such as Decision Tree, KNN, Naïve Bayes, Logistic Regression, Stochastic Gradient Descent, and SVM, are compared [15,16,17]. Particularly, special attention is paid to the selection of the best kernels for SVM to achieve better results. The best model among the studied models is then deployed via an API for use in real time. This research aims to fill the performance-gap between models and their usability in practice through the creation of a framework that will not only be accurate but also able to give instantaneous and understandable output to the end user. Unlike many existing studies that primarily focus on either model performance or deployment aspects separately, this paper will cover both aspects, thus making it relevant to real-life applications.

2. Methodology

The proposed methodology follows a structured pipeline [6], as shown in Fig. 1, covering data collection, preprocessing, feature extraction, model training, and evaluation.

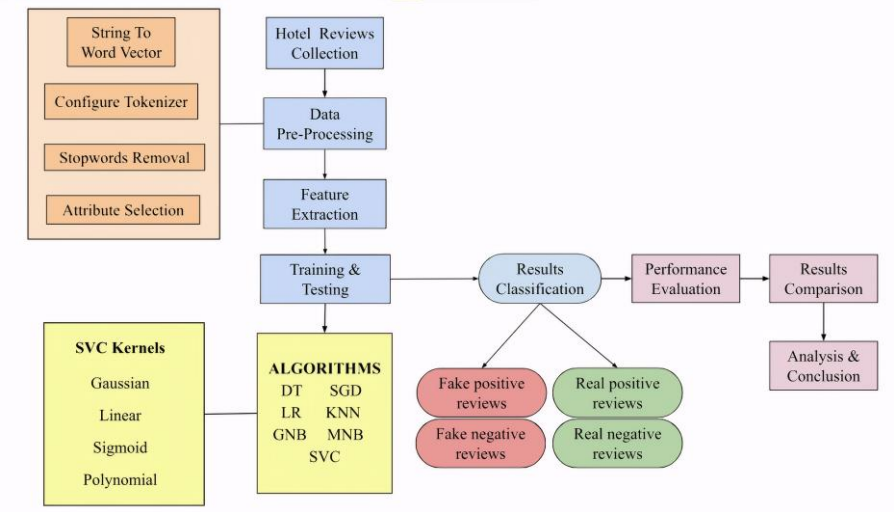

Figure 1. Proposed Methodology

2.1 Dataset Description

The dataset used in this study consists of 11,912 hotel reviews from publicly available datasets extracted through Kaggle. The dataset consists of both genuine and fabricated reviews based on the labels provided in the original datasets, without performing any additional manual annotation. Reviews were combined and standardized into a unified dataset after removing redundant entries. In order to ensure a balanced data, equal numbers of authentic reviews and false reviews totaling 5,956 were selected. Each data point includes the review content together with context related to hotel user reviews [16].

The reviews in the dataset differ considerably in terms of textual length and writing patterns. On average, the number of words in one review equals 60 while varying from 1 word up to 784 words. Data preprocessing was carried out before the start of model training to eliminate duplications in the data [14]. Duplicate reviews were removed prior to the train-test split in order to avoid possible data leakage between training and testing sets.

Thus, the dataset provided by Kaggle sources is diverse enough in terms of length, style, and patterns of reviews so that machine learning models could be trained to detect fake reviews. Nevertheless, since the dataset comes from publicly available online resources, there might be some peculiarities in its texts affecting the model training process.

2.2 Data Preprocessing

Pre-processing was done to prepare and normalize the textual data before modeling [18]. In the process of pre-processing, the text reviews were converted to lower case, punctuation and special characters removed, numerical patterns normalized, stop words removed with NLTK package, text was tokenized, and stemming was done through Snowball Stemmer method [6,14]. Regular-expression based pre-processing was also done for normalization of texts and removal of unwanted symbols.

After the pre-processing step, the dataset was divided using an 80:20 train-test split with random_state = 5. Pre-processing and feature extraction were done only on the training dataset and then applied to the test dataset to avoid any information leak.

2.3 Feature Extraction

The feature extraction was achieved using TF-IDF vectorization technique, which assists in converting the textual data to vectors of features. The configuration of the TF-IDF vectorizer was performed using unigram features with the help of parameters ngram_range (1,1), min_df = 1, max_df = 1.0, and L2 normalization. The tokenization of data was done through the default tokenizer with parameter token_pattern "(?u)\b\w\w+\b". The stop words removal and normalization of text data were also performed as part of preprocessing before vectorization.The vectorizer was fitted only on the training data and subsequently applied to the testing data in order to avoid information leakage [5,6].

2.4 Model Configuration and Training

A total of seven classification models have been used for the purpose of training in this study. These include Decision Tree , K-Nearest Neighbor, Gaussian Naive Bayes, Multinomial Naive Bayes, Logistic Regression, Stochastic Gradient Descent, and Support Vector Machine [17,19].

TF-IDF vector representation was obtained using unigram representation with the use of ngram_range = (1,1), min_df = 1, max_df = 1.0 and L2 normalization. These vectors were utilized as features in all the classification models[2,5,20].

Default configuration of the SVM classifier included the use of Scikit-learn package along with the default setting of the RBF kernel with C = 1.0, gamma = 'scale' and class_weight = None. In addition to the above kernel setting, linear, polynomial, and sigmoid kernels were also considered for comparison purposes. Max_depth = 4 has been considered as the maximum depth for the Decision Tree model. Neighbors values such as k = 3, 5 and 7 have been considered for KNN models. The remaining classifiers have been configured using default configurations [9,14].

2.5 Evaluation Metrics

Evaluation metrics were used to assess the performance of the proposed classification models in this paper. Such evaluation metrics may include accuracy, precision, recall, and F1-score. These metrics provide useful insights into the classification process of genuine and fake reviews. Confusion matrix may also be used for classification evaluation purposes[20,21].

2.6 System Implementation

For better functionality, the best performing model was implemented in a real-time web application via an API-oriented structure. This application was designed based on the use of the Django framework for backend, and React.js, HTML and CSS for the front end.

The implemented application enables customers to enter hotel reviews from the web portal, and then the entered text is sent to the customized Predict API as JSON objects. The API performs preprocessing, TF-IDF vectorization, and classification using the trained machine learning model before returning the prediction result along with the associated probability score. The classified output is then formatted and displayed on the website in real time.

In addition to single-review analysis, the system also includes a simulated e-commerce review portal where submitted reviews are stored through a database integration workflow. The implementation workflow of the proposed system is illustrated in Fig. 2. The reported inference latency includes preprocessing, feature extraction, model prediction, and API response generation under local deployment conditions.

2.7 Algorithms Implemented

Seven ML models have been chosen to solve the problem of classifying reviews into authentic and fake categories. These seven algorithms include Decision Tree(DT), K-nearest Neighbor (KNN), Gaussian Naïve Bayes (GNB), Multinomial Naïve Bayes (MNB), Logistic Regression (LR), Stochastic Gradient Descent (SGD), and Support Vector Machine (SVM) [17,19].

Decision Tree is a non-parametric learning model which learns decision rules from data for classification. It has the ability to learn non-linear relationships between the features and the target label[21]. KNN classifies samples based on similarity with neighbouring data points, while Gaussian and Multinomial Naïve Bayes are probabilistic classifiers based on Bayesian theory. Logistic regression and SGD are linear classification methods that are efficient and have been widely employed in applications of text classification. Support Vector Machine uses a strategy that maximizes the margin between classes through the identification of the best separating hyperplane [17].

Various SVM kernel types such as linear, polynomial, sigmoid, and radial basis function have been analyzed to determine the effect on classification accuracy. Default parameters of Scikit-learn were considered for almost all classifiers except for KNN and Decision tree where KNN was tested with multiple neighbors while the max depth for Decision Tree was fixed to 4.

2.8 SVM Kernels Implemented

A. Linear Kernel

The linear kernel is used when the data is linearly separable and it is efficient for high-dimensional text data. It computes the dot product between feature vectors, enabling direct separation in the original feature space.

$$K(X1, X2) = X1 . X2$$

B. Polynomial Kernel

The polynomial kernel captures interactions between features and is useful for non-linear classification.

$K(X1, X2) = (a + X1^T X2)^b$, where c is a constant and d is the degree of the polynomial.

C. Sigmoid Kernel

The sigmoid kernel behaves similarly to activation functions used in neural networks.

$k(x, y) = tanh(\alpha x^T y + c)$, where α controls the slope and c is a constant.

D. RBF (Gaussian) Kernel

The RBF kernel is widely used for non-linear problems and measures similarity between data points. It maps data into a higher-dimensional space, allowing better separation of complex patterns.

$$K(X1, X2) = exp(-\frac{||X1 - X2||^2}{2\sigma^2})$$

2.9 System Working

The workings of the proposed system is demonstrated using Fig. 2 below. First, a customer will make a review using the web platform interface for the e-commerce site. The text entered will be formatted as a JSON object and then transmitted to the prediction API. At this point, the necessary processing will be done, and the machine learning model is applied for the classification of the review based on the features learned. A JSON response will be received, which will have the labeled information along with the associated confidence value. This will then be processed at the web back end, where decisions will be made based on it. The labeled review will now be saved in the database for future references. Therefore, each review submitted by customers will be examined and classified.

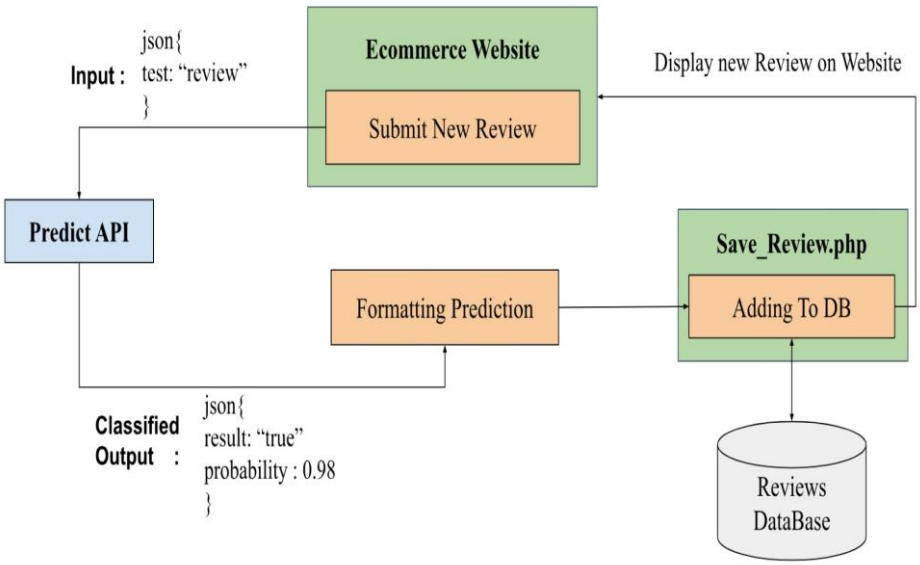

Figure 2. Work Flow of Proposed System in Real Time

The whole process will take place within a second due to the real-time nature of the system, thus providing an easy integration of the classification and the web platform.

3. Results and Discussion

3.1 Comparison of Algorithms

Table 2. Comparison of 7 Machine Learning Models

No.	Model	Accuracy	Precision	Recall	F1 Score
1	SVC	0.88	0.9170	0.8634	0.8894
2	K-NN	0.73	0.79	0.63	0.70
3	DT	0.63	0.58	0.91	0.71
4	LR	0.81	0.79	0.84	0.82
5	MNB	0.79	0.74	0.89	0.81
6	GNB	0.74	0.77	0.68	0.72
7	SGD	0.83	0.80	0.87	0.83

Comparison between seven implemented algorithms in the machine learning process is outlined in Table 2 above. It is also important to state that from among the models under study, the best model in terms of its accuracy score equal to 0.88 was obtained from the Support Vector Classifier (SVC). This model demonstrated balanced scores for both precision and recall resulting in a high F1-score in comparison with other models. The mentioned performance level can be attributed to the fact that SVM models can be effectively applied for high-dimensional text processing.

On the contrary, The Stochastic Gradient Descent (SGD) and Logistic Regression models also showed competitive performance, achieving accuracies of 0.83 and 0.81 respectively. In contrast, the accuracy score for other models varied between 0.63 and 0.79 thus making it clear that models such as the K-Nearest Neighbor, Multinomial Naive Bayes, Gaussian Naive Bayes, and Decision Tree are showing comparatively less effectiveness for the given textual dataset.

Summarizing the results received, it is necessary to emphasize that the SVC algorithm showed the highest performance.

3.2 Comparison of SVM Kernels

The results regarding different SVM kernels are summarized in Table 3 below. The RBF kernel yielded the best accuracy score at 0.88 along with impressive precision, recall, and highest F1 score, thus indicating better performance than other kernels tested. The reason behind the efficiency of the RBF kernel is explained by its ability to deal with non-linear relations that may occur in text data.

The results from the Polynomial and Linear kernel were similar because they both had the same level of accuracy of 0.82, making them both efficient classifiers for dealing with simple tasks. Conversely, the Sigmoid kernel was inefficient because it yielded the lowest accuracy at 0.79.

Based on the results, it can be said that selection of the right kernel could greatly influence the performance of an SVM classifier.

Table 3. Comparison of 4 SVM Kernels

No.	Kernel Implemented	Accuracy	Precision	Recall	F1 Score
1.	Gaussian Kernel (rbf)	0.88	0.9170	0.8634	0.8894
2.	Polynomial Kernel	0.82	0.79	0.86	0.82
3.	Linear Kernel	0.82	0.797	0.86	0.82
4.	Sigmoid Kernel	0.79	0.77	0.83	0.80

The outcomes produced as a result of applying this research were cross-checked by comparing the results to existing literature related to identification of fake reviews. As per Wang et al., [1] existing literature regarding this subject matter has yielded accurate results in the range of 0.80 to 0.85. However, the SVM model using the RBF kernel yielded a higher accuracy rate of 0.88, indicating its competency within such a context. However, any form of comparison between this research and previous literature requires careful analysis owing to the different data sets used for experimentation.

From the above experiment, it is obvious that SVM has proven to work well in dealing with text classification challenges. Apart from the excellent performance in terms of text classification, it also makes provisions for practicality by real-time predictions through API-based deployment architecture .

3.3 Comparison with a Transformer-Based Baseline

DistilBERT is another transformer-based model that has been trained using the same dataset for a benchmark comparison [13]. The reason behind the choice of DistilBERT was due to its performance efficiency. As can be seen from Table 4 below, the DistilBERT model obtained 0.8788 accuracy and 0.8769 F1-score. The performance level of DistilBERT is quite similar to the TF-IDF/SVM model; however, transformer-based models need more computing power than their counterparts.

Table 4. Baseline Distribution Using DistilBERT

No.	Model	Accuracy	F1-score
1.	DistilBERT	0.8788	0.8769

3.4 Cross Validation and Robustness

To check for robustness and minimize sensitivity to a particular train-test split, five-fold cross validation was carried out on the whole data set to maintain consistency of the classes within each fold. As can be seen from Table 5, the SVM classifier with the RBF kernel showed the following results: 0.8889 ± 0.0043 for the mean accuracy metric and 0.8932 ± 0.0043 for the mean F1-score metric. The small standard deviations show stability and consistency of the obtained results.

Table 5. Cross-Validation performance of SVC(RBF)

No.	Metric	Mean	Std. Dev
1.	Accuracy	0.8889	0.0043
2.	F1-score	0.8932	0.0043

3.5 Confusion Matrix and Precision Recall Curve

The precision-recall curve and confusion matrix together reflect a model that is confident most of the time but still leaves room for small errors—something quite normal with real-world data[22] . Fig. 3 represents the confusion matrix of the SVC model. The matrix clearly shows that the model performs well while classifying between real and fake hotel reviews and it correctly identifies a large proportion of both classes. Even with some misclassifications, the confusion matrix shows that the model performs well and is capable of accurate classification.

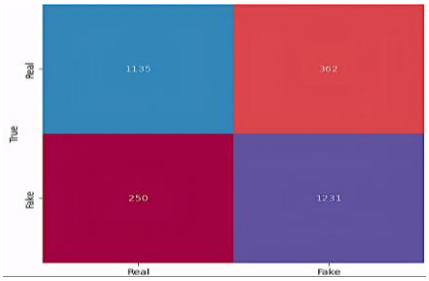

Figure 3. Confusion Matrix

Along with the confusion matrix, Fig. 4 shows a Precision–Recall Curve. It illustrates that the SVC model is effective. The curve shows larger values of precision for a wide range of recall values, and it declines only when recall values reach maximum. This curve thus reflects the model's ability to find significant occurrences while keeping false positives relatively low. The shape of the curve validates the reliability of SVC model, especially for implementations where high precision is required, such as classifying suspicious or potentially harmful user-generated content.

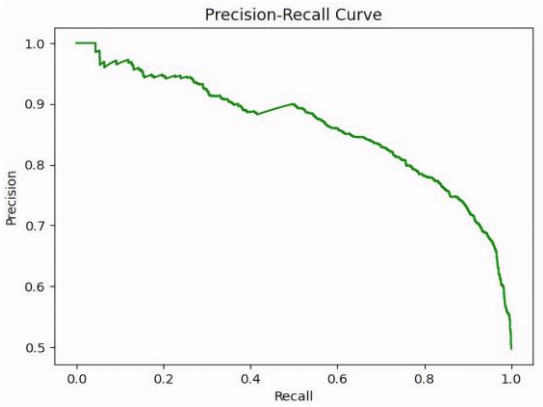

Figure 4. Precision-Recall Curve

Key Contributing Terms Identified by Linear SVM

Deceptive-indicative terms: *not, mile, michigan, dirty, ave, poor, rude, horrible, stayed, and avenue.*
Genuine-indicative terms: *luxury, excellent, clerk, staying, millennium, modern, soon, questions, vacation, and bit.* Overall, Deceptive reviews primarily contain complaint-oriented language, whereas genuine reviews emphasize contextual and accommodation-related terms.

A. Runtime Performance:
In order to measure the suitability of the proposed approach for real-time applications, the inference latency was measured. The experiment was performed using a CPU-based setup, reporting the average inference latency of 11.90 ms with a standard deviation of 4.20 ms, over 100 runs. The proposed approach is found to be computationally efficient, enabling the implementation of the proposed review classification system in a near-real-time environment.
Apart from offline inference measurements, the model was incorporated into a real-time prediction pipeline, wherein the measured response times were found to match those mentioned for the inference latency.

B. Error Analysis:
An error analysis was carried out to identify the misclassified reviews. On the test data, the model generated 232 false positives, i.e., actual reviews classified as deceptive, and 123 false negatives, which are deceptive reviews classified as actual. False positives were found to be actual reviews that strongly expressed dissatisfaction, whereas false negatives were found to be deceptive reviews that were written in an impersonal factual way, simulating actual reviews.

4. Conclusion

In this study, the possibility of constructing a machine learning classifier system with real-time prediction capabilities able to detect false hotel reviews was explored. The best result in terms of accuracy was observed for the Support Vector Classifier with RBF kernel, reaching the score of 0.88, while achieving consistently high scores during cross-validation.

One of the contributions made in this study is that it encompasses the following aspects: multiple models' comparison, exploration of SVM kernels and real-time deployment via the use of API-based approach. Differing from many research papers that mostly concentrate on comparing models on a static dataset, this paper also demonstrates practical applicability by providing a possibility to predict outcomes in real time. Moreover, the

performance level of the proposed model reaches that of DistilBERT transformer model while using less computational resources.

Nevertheless, there are also some limitations in the study presented. Namely, the dataset is related only to hotel reviews, thus affecting generalization abilities in other types of reviews. Even though the baseline is provided using the DistilBERT transformer model, the focus of the paper was on classical ML methods due to their lower computational complexity and ease of deployment.
Potential future works may consist of extending the proposed framework by means of implementing more sophisticated transformer-based models such as BERT, and adding more context-based factors, like behaviors of reviewers, temporal characteristics, and meta information to combat highly intelligent spams. Moreover, the proposed framework can be applied to other fields, such as product reviews, restaurant reviews, and services review systems.

5. Acknowledgement

It is with immense pleasure that we thank Prof. Kamal Shah and Prof. Sunny Sall for their inspiration and provision of necessary resources for this study. We also take this opportunity to acknowledge the support and cooperation of our peers and faculty members throughout this work.

6. Funding

No funding was received for this study.

7. Data Availability

The dataset used in this study is available upon request.

8. Declarations

Competing Interests: The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

References

[1] Wang G, Shang G, Pu P, Li X, Peng H. Fake review identification methods based on multidimensional feature engineering. Mobile Information Systems. 2022;2022(1):5229277. Available from: https://doi.org/10.1155/2022/5229277

[2] Kadam M, Marewad S, Nemade C, Mote P. Fake review detection using machine learning and deep learning. International Journal of Scientific Research in Science and Technology. 2024;11:489-497. Available from: https://doi.org/10.32628/IJSRST24115119

[3] Hájek P, Hikkerova L, Sahut JM. Fake review detection in e-commerce platforms using aspect-based sentiment analysis. Journal of Business Research. 2023;167:114143. Available from: https://doi.org/10.1016/j.jbusres.2023.114143

[4] Liu M, Poesio M. Data augmentation for fake reviews detection in multiple languages. Proceedings of Recent Advances in Natural Language Processing. 2023;673-680. Available from: https://doi.org/10.26615/978-954-452-092-2_073

[5] Sultana N, Palaniappan S. Deceptive opinion detection using machine learning techniques. International Journal of Information Engineering and Electronic Business. 2020;12(1):1-7. Available from: https://doi.org/10.5815/ijieeb.2020.01.01

[6] Gupta D, Bhargava A, Agarwal D, Alsharif MH, Uthansakul P, Uthansakul M, Aly AA. Deep learning-based truthful and deceptive hotel reviews. Sustainability. 2024;16(11):4514. Available from: https://doi.org/10.3390/su16114514

[7] Du, W., Li, J., Zhou, J., Lu, Q., & Sun, Y. (2025). Mitigating the proliferation of fake image-text reviews: A two-tier intra- and inter-modal fusion framework. International Journal of Electronic Commerce, 29(2): 304–332. Available from : https://doi.org/10.1080/10864415.2025.2471673

[8] Nawara, D., & Kashef, R. (2025). A dual-phase framework for detecting authentic and computer-generated customer reviews using large language models. Decision Analytics Journal, 15, 100581. Available from : https://doi.org/10.1016/j.dajour.2025.100581

[9] Goyal NK, Pal A, Keswani B, Goyal D, Gupta MK. A novel hybrid feature extraction technique and spam review detection using ensemble machine learning algorithm by web scraping. Indian Journal of Science and Technology. 2023;16(29):2261-2268. Available from: https://doi.org/10.17485/IJST/v16i29.1500

[10] Davoodi L, Mezei J, Heikkilä M. Aspect-based sentiment classification of user reviews to understand customer satisfaction of e-commerce platforms. Electronic Commerce Research. 2026;26:1417-1459. Available from: https://doi.org/10.1007/s10660-025-09948-4

[11] Banik, S., & Rajak, R. (2025). Fake review detection: Taxonomies, benchmarks, and intent modeling frameworks. Journal of Information Systems Engineering and Management, 10(42s), 1270–1289. Available from : https://doi.org/10.52783/jisem.v10i42s.9224

[12] Olivo C, Santin AO, Viegas EK. Towards a reliable spam detection: an ensemble classification with rejection option. Cluster Computing. 2025;28:24. Available from: https://doi.org/10.1007/s10586-024-04742-7

[13] Abhijeet R, Gayatri B, Ankita J, Kishor D, Jayshree M. Fake reviews detection using NLP model and neural network model. International Journal of Engineering Research and Technology. 2023;12(5):51-56. Available from: https://doi.org/10.17577/IJERTV12IS050063

[14] Asaad W, El-Sayed M, Hussein H. Fake review detection using machine learning. Revue d'Intelligence Artificielle. 2023;37:507. Available from: https://doi.org/10.18280/ria.370507

[15] Advanced E-Commerce Authenticity: A Novel Fusion Approach based on Deep Learning and Aspect Features for False Reviews Detection B. Jhansi, B Keerthana, P Mohan, B Bhavana, B. K. Dinesh. International Journal of Emerging Research in Science, Engineering, and Management Vol. 2, Issue 1, pp.147-152, January 2026. Available from : https://doi.org/10.58482/ijersem.v2i1.20

[16] Ignat O, Xu X, Mihalcea R. MAiDE-up: multilingual deception detection of GPT-generated hotel reviews. arXiv preprint. 2024. Available from: https://doi.org/10.48550/arXiv.2404.12938

[17] Gasparetto A, Marcuzzo M, Zangari A, Albarelli A. A survey on text classification algorithms: from text to predictions. Information. 2022;13(2):83. Available from: https://doi.org/10.3390/info13020083

[18] Asudani DS, Nagwani NK, Singh P. Impact of word embedding models on text analytics in deep learning environment: a review. Artificial Intelligence Review. 2023;56:10345-10425. Available from: https://doi.org/10.1007/s10462-023-10419-1

[19] Sánchez-Paniagua, M., Fidalgo, E., Alegre, E., & Jáñez-Martino, F. (2026). Fraud detection in e-commerce: A comparative analysis of features to enhance machine learning models. Electronic Commerce Research, 26, 2467–2502. Available from : https://doi.org/10.1007/s10660-025-10029-9

[20] Prathibha R, Sahana S, Yashashwitha R. Fake product review detection and removal system using NLP. International Journal of Advanced Research in Science, Communication and Technology. 2022;2(9):342-345. Available from: https://doi.org/10.48175/IJARSCT-5349

[21] Elamin N, Talab SA, Khalid A. Sentiment analysis with supervised learning techniques: a survey. Indian Journal of Science and Technology. 2020;13(3):249-268. Available from: https://doi.org/10.17485/ijst/2020/v13i03/148900

[22] Ashraf SA, Javed AF, Bellary S, Bala PK, Panigrahi PK. Leveraging stacking framework for fake review detection in the hospitality sector. Journal of Theoretical and Applied Electronic Commerce Research. 2024;19(2):1517-1558. Available from: https://doi.org/10.3390/jtaer19020075