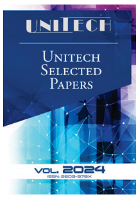

INTEGRATING DATA SCIENCE AND PREDICTIVE MODELING FOR DETECTING INCONSISTENT HOTEL REVIEWS

Milena Nikolić[1,2]*, Miloš Stojanović[2], Marina Marjanović[3]

[1,2]Department of Information and Communication Technologies
[2]Academy of Technical-Educational Vocational Studies, Niš, Serbia
[3]Faculty of Technical Sciences, Singidunum University, Belgrade, Serbia
** Corresponding author: milena.nikolic@akademijanis.edu.rs*

Abstract

With the rising dependence on online reviews in the hotel industry, it is essential to identify and remove inconsistent or misleading feedback to ensure the credibility of review platforms. This paper presents a predictive modeling approach aimed at detecting inconsistent hotel reviews through a combination of sentiment analysis, correlation assessment, and advanced feature engineering techniques. Our methodology involves extracting sentiment scores from review texts, titles, and tags using the VADER sentiment analysis tool, which is particularly suited for evaluating informal, user-generated content. By analyzing the correlations between sentiment scores and the numerical ratings provided by reviewers, we identify potential mismatches that indicate inconsistencies. To enhance detection accuracy, we implement sophisticated criteria based on sentiment mismatches and correlation thresholds. For the classification of reviews, we employ the XGBoost algorithm, known for strong performances in handling structured data. Using RandomizedSearchCV, we fine-tune the model to achieve higher levels of precision. This technique successfully filters out inconsistent reviews and provides insights for enhancing the reliability of online feedback systems. Our results emphasize the value of data science and predictive modeling in ensuring the integrity of review data, ultimately enabling consumers to make more well-informed decisions.

Keywords: Hotel Industry, Online Reviews, Sentiment Analysis, Feature Engineering, Correlation Analysis

1. Introduction

In recent years, online reviews have become crucial for the hotel industry, as guests increasingly rely on user feedback to make booking decisions. However, the rise in reviews presents the challenge of distinguishing authentic feedback from misleading or inconsistent ones, which can compromise platform credibility and lessen consumer trust. Ensuring reliable, truthful feedback has become essential, prompting the need for advanced methods to detect and filter inconsistencies in hotel reviews [1].

Existing solutions primarily use basic sentiment analysis, rule-based systems, and manual moderation. While tools like VADER assess the overall tone of reviews, they often fail to evaluate the alignment between sentiment and numerical ratings. Machine learning models, like random forests and support vector machines, have been used as well, but may face challenges with informal language and sentiment mismatches, especially when the textual sentiment does not align with the given rating.

This research introduces an innovative approach that integrates sentiment analysis with a correlation-based framework to accurately detect inconsistencies. By utilizing VADER to extract sentiment scores from review texts, titles, and tags, and analyzing their alignment with ratings, we can effectively pinpoint sentiment mismatches that suggest inconsistencies. For classification purposes, we utilize XGBoost, a robust gradient-boosting model, and then we optimize it by using RandomizedSearchCV to enhance accuracy over large datasets.

Our approach has limitations, including reliance on correlation thresholds, which may overlook subtle sentiment variations not fully captured in ratings. Although effective, additional refinement may be necessary to adapt this method to different hospitality contexts. Despite this weakness, the

doi: www.doi.org/10.70456/.........................

presented approach introduces a scalable framework aimed at improving the reliability of online booking reviews by filtering out inconsistencies. By enhancing the credibility of review platforms, we empower consumers to make more informed decisions and demonstrate how data science techniques can strengthen the trustworthiness of user-generated content in the hotel industry.

1.1. Related work

The problem of detecting and filtering inconsistent online reviews has gained significant attention over time, particularly as businesses increasingly rely on user-generated feedback. Early approaches were primarily focused on manual or rule-based methods, which often lacked the scalability needed for large datasets and proved as time-consuming. More recent efforts have leveraged machine learning techniques to enhance detection capabilities, allowing for quicker and more efficient analysis.

Several studies have explored popular machine learning models for irregularity detection in reviews, highlighting a growing interest in automating this process [2]. For instance, techniques such as clustering and outlier detection have been employed to identify unusual patterns in review data, revealing insights that might not be noticeable through the manual examination. Moreover, natural language processing (NLP) methods, including sentiment analysis and topic modeling, have been frequently utilized to extract meaningful features from review texts, further enriching the analysis [3]. These methods have shown success in improving the accuracy of anomaly detection, but a lot of challenges persist in managing the large volumes of data and the varied nature of review content, highlighting the need for continued research in this field.

1.2. Overview of Existing Methods

Existing approaches and techniques for identifying inconsistencies in online reviews generally fall into two main categories: rule-based approaches and machine learning models [4].

Early methods rely on predefined rules and heuristics to identify inconsistencies [5]. These methods usually use keyword matching, sentiment mismatch detection, and rule-based anomaly scoring. While these strategies can be effective in specific contexts, they often struggle with flexibility and growth potential, making them less suitable for large datasets or diverse review content.

Recent advancements involve the use of machine learning algorithms to automate the overall detection process. Supervised learning techniques, where models are trained on labeled datasets to spot inconsistencies, have significantly improved accuracy [6]. Common algorithms include Decision Trees, Support Vector Machines, and ensemble methods such as Random Forests and Gradient Boosting Machines, which enhance predictive performance [7].

In summary, although rule-based methods offer a foundational approach, machine learning approaches offer a more scalable and adaptable solution for detecting inconsistencies in online reviews. Our paper leverages advancements by incorporating sentiment analysis, correlation assessment, and advanced feature engineering to achieve high accuracy in inconsistency detection.

1.3. Dataset Description

To conduct the desired analysis, we use a dataset from Kaggle, which is a part of the extensive Booking.com Reviews Dataset. This collection includes 26,675 hotel reviews from Booking.com website, offering detailed insights into customer experiences across various hotels and locations [9]. The dataset includes a wide range of details, such as review titles, review texts, review dates, reviewer (user) names, hotel names, URLs, numeric ratings, nationalities of reviewers, image counts, assigned tags, and other relevant metadata. We mainly chose this dataset because it effectively represents the key review components necessary for effective detecting inconsistencies and ensuring data reliability.

1.4. Feature Selection

The decision to select specific columns from the dataset was driven by the challenge of detecting inconsistencies within the large number of high ratings that we observed through initial analysis. We focused on features like review title, review text, tags, and numerical ratings, given the complexity of separating genuine and potentially inconsistent high ratings. By examining these crucial elements, our objective is to assess how well the content of the reviews aligns with given ratings, aiming to identify any discrepancies.

2. Methodology

The given section details the methodology employed to identify inconsistencies in hotel reviews. The proposed idea incorporates several essential steps: data preprocessing to clean and prepare the dataset, sentiment analysis to assess the emotional tone of review content, correlation analysis for detecting inconsistencies between key review elements, defining reliability metrics, then feature engineering, model training for classifying inconsistencies, and finally, model evaluation to assess and interpret performance results.

2.1. Data Preprocessing

In this phase, we focus on preparing the dataset for analysis by selecting relevant features and handling any missing values. We specifically choose textual features, such as review titles and texts, along with the rating given in each review. Fig. 1. illustrates the distribution of review ratings in the dataset, providing insights into the overall sentiment conveyed by reviewers. To streamline the analysis, irrelevant columns are dropped, and rows missing both review title and review text are excluded to ensure that only complete data is considered.

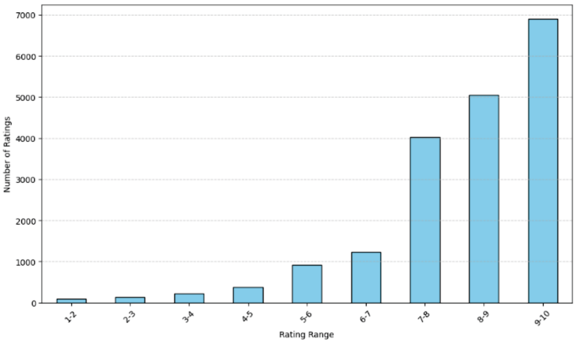

Fig. 1. Distribution of Review Ratings in the Dataset

Remaining missing values in the review text, review title, and tags columns are handled by substituting specific text placeholders to maintain dataset completeness. For reviews lacking text, we fill them with the placeholder "There are no comments available for this review", ensuring these entries are not left blank. Missing titles are replaced with "No title available for this review", while any absent tags are assigned the placeholder "There are no tags available for this review". By filling these gaps, we maintain consistency in the dataset, ensuring it remains usable and ready for analysis despite incomplete data points. To facilitate tag analysis, we extract unique tags from each review. As the tags in each review are separated by the ~ symbol, we first split the tag strings and then aggregate them to identify all unique tags in the dataset. This process enables a more detailed analysis of the sentiment associated with various tags.

2.2. Sentiment Analysis

For sentiment analysis, we used VADER tool (Valence Aware Dictionary and sEntiment Reasoner), which is mostly designed to evaluate sentiment in shorter, informal texts written in English language. VADER is highly effective for analyzing online reviews and social media posts, as it includes a specialized lexicon and is tailored to handle the distinctive language characteristics often present in this type of content.

VADER operates with a pre-established dictionary that assigns sentiment scores to words, reflecting their positive, negative, or eventually neutral impact. Specifically designed for English, this dictionary captures subtle linguistic details, like slang, abbreviations, emoticons, commonly used in informal communication. By calculating the sentiment scores of words within a review and aggregating these scores, VADER provides an overall sentiment assessment for the text.

The main advantage of VADER is the ability to account for context, including sentiment intensity and the influence of eventual negations or intensifiers. This capability leads to a more accurate sentiment evaluation, making VADER particularly effective for brief, informal texts where other tools may struggle.

2.2.1. Sentiment Scores for Text Components

We calculated sentiment scores for both review title and review text fields using VADER's *polarity_scores* method. This method produces a compound score that reflects the overall sentiment of a text by combining the sentiment values of individual words while also considering their intensity and context. The score ranges from -1 (most negative) to +1 (most positive), with values close to 0 indicating neutrality. Once computed, these scores were stored in *title_sentiment* and *text_sentiment* data variables, respectively.

2.2.2. Sentiment Scores for Tags

For analyzing the sentiment of all review tags, we defined the *get_tag_sentiment* function, which uses VADER to calculate sentiment scores of each tag in a review separately, categorizing them as 'positive', 'negative', or 'neutral' based on the compound score. Tags with a score of 0.1 or more are considered 'positive', tags with a score of -0.1 or less are considered 'negative' and scores between -0.1 and 0.1 are labeled as 'neutral'.

Given the structure and the contents of the dataset, the thresholds of -0.2 and 0.2 are chosen to distinguish between moderate sentiment and neutrality. These thresholds successfully identify distinct sentiments while considering possible variations in general textual sentiment.

This classification helps us understand the emotional connotations of the tags. A dictionary, *tag_sentiments*, was created to store and look up the sentiment of each unique tag across the entire dataset. To integrate the overall tag sentiment for each review into the complete data analysis, the *get_overall_tag_sentiment* computes the average sentiment score for the tags associated with each review. These values are stored in *tags_sentiment*, incorporating this aspect into the review analysis.

2.3. Correlation Analysis

To analyze relationships between different features within our dataset, we use the Pearson correlation coefficient. This statistical measure assesses the linear relationship between two variables, providing a value between -1 and 1. The Pearson correlation coefficient ***r*** between two variables ***X*** and ***Y*** can be expressed using the following formula:

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}}$$

where:

- X_i and Y_i are the individual sample points.
- $\bar{X}$ and $\bar{Y}$ are means of X and Y, respectively.
- The numerator is the covariance of X and Y .
- The denominator is the product of the standard deviations of X and Y.

A coefficient of 1 signifies a perfect positive linear relationship, -1 indicates a perfect negative linear relationship, and 0 suggests no linear relationship. In our context, we use this coefficient to assess the correlation between sentiment scores of provided review titles, texts, ratings, and tags.

When a review component is missing, we calculate the correlation based on the remaining available components. For instance, if the review title is missing, we assess the correlation between the review text sentiment, tags sentiment, and rating. This methodology ensures that we use all available data to establish meaningful correlations and detect inconsistencies.

2.4. Metrics for Review Reliability

After computing correlations, we establish thresholds to pinpoint inconsistencies based on the standard sentiment variations and identified patterns. We set a sentiment difference threshold of 0.8 to detect substantial discrepancies between review components, a correlation threshold of 0.5 to identify weak or non-existent relationships that may indicate anomalies, and thresholds of -0.3 and 0.3 (respectively) to flag sentiment scores that diverge from expected norms.

If they meet these criteria, reviews are classified as inconsistent (value of 1), otherwise, they are considered legitimate (value of 0). The results of classification process are stored in the *is_inconsistent* column of the dataset.

2.5. Feature Engineering

For feature engineering tasks, we begin by transforming the provided textual features into numerical formats using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. This technique quantifies word significance in a document relative to a larger collection known as a corpus. It computes each word's frequency in a specific document (Term Frequency) and adjusts this value based on the word's rarity in the entire corpus (Inverse Document Frequency).

We employ TF-IDF vectorizers to convert the review title, review text, and tags columns into numerical feature matrices. These matrices reflect the significance of words within their respective text fields, allowing us to represent textual data in a format that is compatible with machine learning models.

Furthermore, we incorporate the numeric rating feature alongside the TF-IDF matrices. This is accomplished by horizontally stacking the TF-IDF matrices for the title, text, and tags, followed by appending the rating feature. The resulting combined feature matrix integrates both textual and numeric data, offering a comprehensive input for our proposed detection model.

2.6. Model Training

To train and evaluate the model, we first examined the performance of the Random Forest classifier. However, the initial results did not meet our expectations, leading us to seek alternative approaches. We then transitioned to the XGBoost, which showed improved performance [10].

The primary goal was to detect anomalies within the dataset, focusing on the is_inconsistent column as the target variable.

XGBoost (Extreme Gradient Boosting) is especially suitable for this task because of its advanced features. In contrast to Random Forest, XGBoost incorporates regularization methods to prevent overfitting and enhance overall model generalization. Additionally, it manages larger datasets and high-dimensional feature spaces successfully, providing improved scalability. The inherent capability to address missing values also boosts its effectiveness in practical applications.

To optimize the XGBoost model, we used RandomizedSearchCV for efficient optimization of hyperparameters. This method was selected over GridSearchCV due to its greater efficiency and ability to explore the parameter space more thoroughly. While GridSearchCV exhaustively examines all possible parameter combinations, RandomizedSearchCV samples a subset, which reduces computational resources and time. This methodology allows for a broader exploration of the hyperparameter space by sampling from specified distributions, which may reveal more effective parameter configurations. By opting for RandomizedSearchCV, we aimed to establish a balance between computational efficiency and the depth of parameter exploration, ultimately leading to a more optimized model. This strategy not only enhanced the performance of a model but also ensured more practical utilization of resources, enabling us to focus on refining our results.

2.7. Model Evaluation

After optimizing the hyperparameters, we evaluated the model’s performance using standard metrics such as accuracy, precision, recall, and F1-score. The model achieved an accuracy of 88%, showcasing its capability to effectively differentiate between and inconsistent reviews, which indicates a strong level of performance in identifying anomalies within the dataset. The attained level of accuracy indicates that the model is reliable in its predictions, minimizing both false positives and false negatives.

The illustration below (Fig. 2.) displays a feature importance graph that emphasizes the relative significance of the top five features used in the model's decision-making. Upon analysis, it is evident that the leading terms mainly derive from highly positive reviews, indicating that these features play a crucial role in shaping the model's evaluations. This insight not only reinforces the robustness of a model but also provides valuable guidance for understanding which aspects of the review contents contribute most significantly to its predictive capabilities.

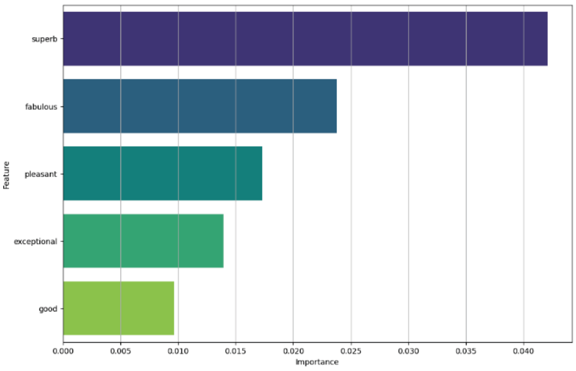

Fig. 2. Feature Importance Graph

3. Results

The accuracy of 88% indicates that the model correctly classified 88% of the reviews in the test set. This result underscores the model's capability to effectively differentiate between legitimate and inconsistent reviews. Furthermore, the classification report offers deeper insights into the performance metrics. Specifically, for class 0 (legitimate reviews), the precision is 0.88, with a recall of 0.83. In contrast, for class 1 (inconsistent reviews), both precision and recall are 0.88 and 0.92, respectively. These metrics reveal that the model excels at identifying inconsistent reviews, which is crucial for anomaly detection.

Moreover, the F1-scores, which balance precision and recall, are 0.85 for class 0 and 0.90 for class 1, highlighting a balanced performance across both classes. The substantial sample size, comprising 2,241 instances for class 0 and 3,037 for class 1, further confirms the model's reliability in detecting anomalies, achieving a noteworthy equilibrium between precision and recall.

While these findings align with current literature on anomaly detection in online reviews, they also uncover distinct aspects of the model's performance that may foster further discussions and research into review authenticity. Limitations include the potential bias present in the dataset and the necessity for a varied selection of review sources to improve generalizability. In summary, this section not only presents the results but also examines the earlier discussed points, establishing a strong foundation for the conclusions drawn.

4. Conclusions

The conducted study presents an effective methodology for identifying inconsistent hotel reviews using techniques like sentiment analysis, correlation assessment, and predictive modeling. By employing the VADER sentiment analyzer, we evaluated sentiments in review titles, texts, and tags to reveal discrepancies with assigned ratings. The integration of various correlation thresholds enhanced our capacity to detect inconsistencies, leading to an XGBoost algorithm that achieved an accuracy of 88%. Additionally, hyperparameter optimization through RandomizedSearchCV was essential, providing a refined balance between computational efficiency and in-depth exploration of the parameter space.

These findings enhance the reliability of online reviews and underscore their potential for practical applications in the hotel industry. Future research could refine these methodologies with additional model training and deeper exploration of features for improving detection capabilities and accuracy. Notably, this analysis highlights the intersection of data science and machine learning through addressing challenges in online review authenticity. The unique approach and findings discussed here pave the way for future research, making a significant contribution to the field.

REFERENCES

[1] R. Hassan and M. R. Islam, "Impact of Sentiment Analysis in Fake Online Review Detection," in *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, Bangladesh, 2021, pp. 21-24. doi: 10.1109/ICICT4SD50815.2021.9396899.

[2] C. G. Harris, "Decomposing TripAdvisor: Detecting Potentially Fraudulent Hotel Reviews in the Era of Big Data," *2018 IEEE International Conference on Big Knowledge (ICBK)*, Singapore, 2018, pp. 243-251. doi: 10.1109/ICBK.2018.00040.

[3] N. Kumar, D. Venugopal, L. Qiu, and S. Kumar, "Detecting Anomalous Online Reviewers: An Unsupervised Approach Using Mixture Models," *Journal of Management Information Systems*, vol. 36, no. 4, pp. 1313-1346, 2019.

[4] C. G. Harris, "Comparing Human Computation, Machine, and Hybrid Methods for Detecting Hotel Review Spam," in *Digital Transformation for a Sustainable Society in the 21st Century*, I. O. Pappas, P. Mikalef, Y. K. Dwivedi, L. Jaccheri, J. Krogstie, and M. Mäntymäki, Eds. Cham: Springer International Publishing, 2019, pp. 75-86. doi: 10.1007/978-3-030-29374-1_7.

[5] J. Kumar, *Fake Review Detection Using Behavioral and Contextual Features*, Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan, February 2018.

[6] S. N. Alsubari, S. N. Deshmukh, A. A. Alqarni, N. Alsharif, T. H. H. Aldhyani, F. W. Alsaade, and O. I. Khalaf, "Data Analytics for the Identification of Fake Reviews Using Supervised Learning," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 3189-3204, 2022.

[7] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "Fake review detection: Classification and analysis of real and pseudo reviews," *UIC-CS-03-2013. Technical Report*, 2013.

[8] S. T. Lai and M. Raheem, "Sentiment Analysis of Online Customer Reviews for Hotel Industry: An Appraisal of Hybrid Approach," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 12, pp. 1355-1359, Dec. 2020.

[9] TheDevastator, *Booking.com Hotel Reviews.* [Dataset] https://www.kaggle.com/datasets/thedevastator/booking-com-hotel-reviews

[10] S. Raschka, J. Patterson, and C. Nolet, "Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence," *Information*, vol. 11, no. 4, art. no. 193, 2020.