Manipulated Review Analysis

Avantika Agarwal
Department of Machine Learning
DA-IICT, Gandhinagar, India
202411024@daiict.ac.in

Neel Bokil
Department of Machine Learning
DA-IICT, Gandhinagar, India
202411027@daiict.ac.in

Preet Shah
Department of Machine Learning
DA-IICT, Gandhinagar, India
202411053@daiict.ac.in

***Abstract*—The prevalence of deceptive online reviews has significantly influenced consumer trust and decision-making. This project investigates the "Deceptive Opinion Spam Corpus," a dataset comprising truthful and deceptive reviews of Chicago hotels, with the objective of detecting deceptive reviews using machine learning techniques.**

***Index Terms*—Deceptive reviews, Natural Language Processing, Machine Learning, Deceptive Opinion Spam Corpus, Sentiment Analysis**

I. Introduction

The rise of deceptive online reviews has become a growing concern, affecting consumer trust and purchasing decisions. This project focuses on the "Deceptive Opinion Spam Corpus," a dataset that includes truthful and deceptive reviews of Chicago hotels. The goal is to explore and analyze the dataset using machine learning techniques to detect whether a review is deceptive or truthful. The study includes data preprocessing, exploratory data analysis (EDA), and the implementation of machine learning classifiers to achieve high accuracy in review classification.

II. Problem Statement

The prevalence of deceptive online reviews undermines consumer trust and decision-making. This project explores the "Deceptive Opinion Spam Corpus," a dataset comprising truthful and deceptive reviews of Chicago hotels. The primary objective is to classify reviews as deceptive or truthful using machine learning techniques.

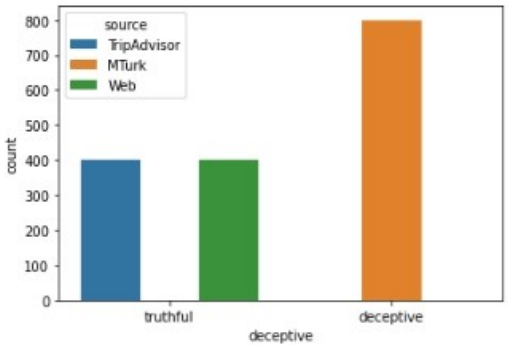

Fig. 1. Diverse Data Sources

Identify applicable funding agency here. If none, delete this.

III. Modules and Workflow

The project workflow is divided into four major modules: Dataset Loading and Exploration, Data Preprocessing, Exploratory Data Analysis (EDA), and Machine Learning Implementation.

A. Dataset Loading and Exploration

The dataset includes:

- **Attributes**: Review text, label (deceptive/truthful), and metadata such as polarity and review source.
- **Distribution**: 400 truthful and deceptive positive reviews (sourced from TripAdvisor and Mechanical Turk, respectively) and 400 truthful and deceptive negative reviews from various platforms.
- **Visualization**:

```
df.isnull().sum()

deceptive    0
hotel        0
polarity     0
source       0
text         0
dtype: int64
```

Fig. 2. Checking missing values

A thorough exploration revealed no significant missing data, ensuring a clean start for preprocessing.

B. Data Preprocessing

The Natural Language Toolkit (NLTK) was used to preprocess the dataset, including tokenization, stopword removal, and Part-of-Speech (POS) tagging.

1) Text Cleaning: The text cleaning process included:

- **Removing Punctuation**: Non-alphanumeric characters were removed using regular expressions.
- **Lowercasing**: Text was converted to lowercase to maintain uniformity.
- **Tokenization**: Text was split into individual words using `word_tokenize`.
- **Stopword Removal**: Common words (e.g., "is," "the") were excluded using NLTK's stopword corpus.
- **POS Tagging**: Words were tagged as nouns (NN), verbs (VB), or adjectives (JJ), which are most relevant for sentiment analysis.

2) Application to Dataset: The processed text was stored back into the dataset, ensuring uniform preprocessing for all reviews.

IV. Exploratory Data Analysis (EDA)

EDA focused on understanding the dataset through visualization and term analysis:

- **Word Clouds**: Generated for deceptive and truthful reviews to identify prominent terms.
- **Observations**: Deceptive reviews often used exaggerated terms (e.g., "amazing," "best"), while truthful reviews described genuine experiences with diverse vocabulary.

V. Machine Learning Implementation

A. Training Classifiers

A variety of machine learning models were employed to classify reviews as truthful or deceptive. The dataset was split into training and testing subsets, with models trained on the preprocessed data. The classifiers implemented include Logistic Regression, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Tree, and Naive Bayes.

1) Logistic Regression: Logistic Regression is a linear model used for binary classification. It predicts probabilities using the logistic function and classifies based on a threshold.

- **Advantages**: Simple, efficient, and interpretable.
- **Limitations**: Assumes linear relationships between features and the log-odds of the outcome.

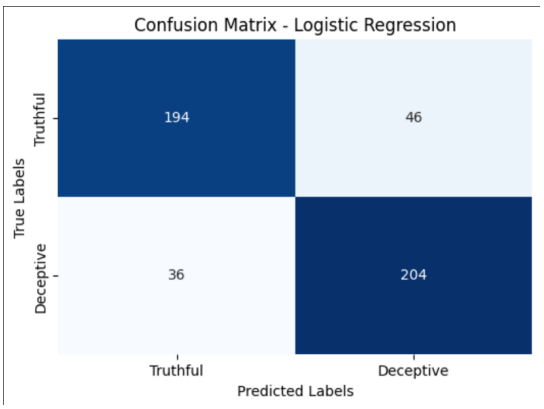

Fig. 3. Logistic Regression: Confusion Matrix

2) Support Vector Machine (SVM): SVM constructs a hyperplane that separates classes with the maximum margin. It is effective in high-dimensional spaces and uses kernel functions for non-linear data.

- **Advantages**: Robust to overfitting, especially with a clear margin of separation.
- **Limitations**: Computationally intensive for large datasets.

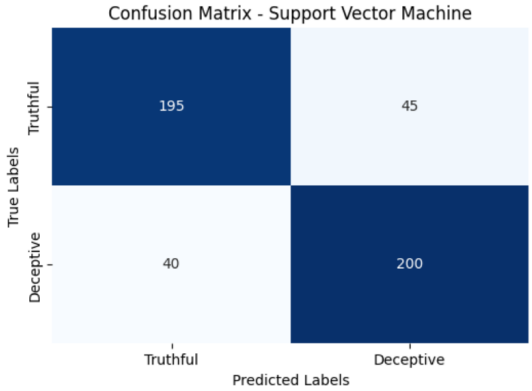

Fig. 4. SVM: Confusion Matrix

3) K-Nearest Neighbors (KNN): KNN is a non-parametric algorithm that classifies samples based on the majority label among their k nearest neighbors.

- **Advantages**: Simple to implement and requires no assumptions about data distribution.
- **Limitations**: Sensitive to the choice of k and computationally expensive for large datasets.

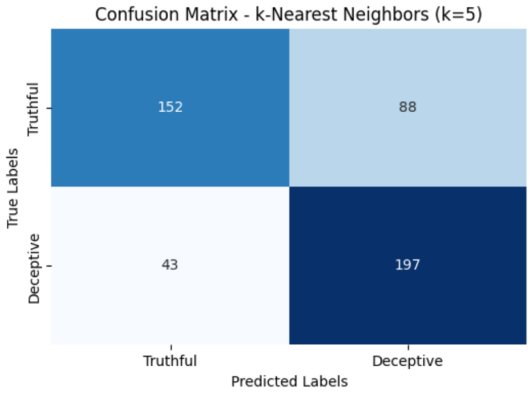

Fig. 5. KNN: Confusion Matrix

4) Decision Tree: Decision Tree creates a tree-like model of decisions based on feature splits. It is interpretable and works well with categorical and numerical data.

- **Advantages**: Easy to visualize, handles both numerical and categorical data.
- **Limitations**: Prone to overfitting if not properly pruned.

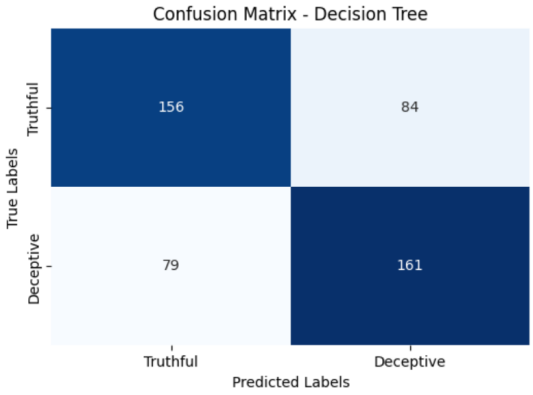

Fig. 6. Decision Tree: Confusion Matrix

5) Naive Bayes: Naive Bayes is a probabilistic model based on Bayes' theorem. It assumes feature independence, which simplifies computation.

- **Advantages**: Fast, efficient for high-dimensional data, and works well with text data.
- **Limitations**: Assumption of independence may not hold in all cases.

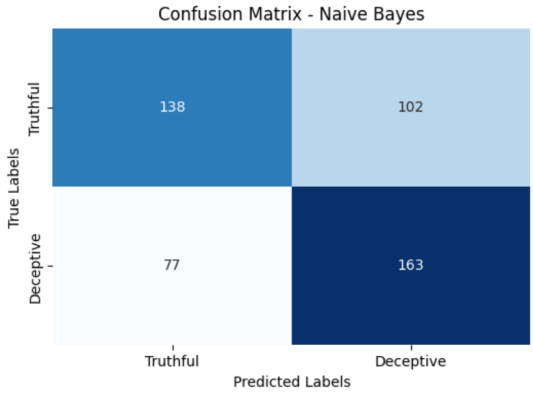

Fig. 7. Naive Bayes: Confusion Matrix

B. Word Cloud

To further explore the differences between deceptive and truthful reviews, word clouds were generated for both categories. The word cloud for truthful reviews predominantly featured words such as "clean," "comfortable," and "friendly," which reflect genuine experiences and a focus on the actual quality of the hotel stay. These words suggest more moderate and practical descriptions, often linked to specific experiences.

In contrast, the word cloud for deceptive reviews contained exaggerated terms like "amazing," "perfect," and "best," which are often used to embellish the review and create an overly positive impression. Such words are typically used in deceptive reviews to mislead consumers by presenting experiences in an unrealistically favorable light. The presence of these terms highlights a pattern of exaggeration and emotional appeal, which is a key characteristic of deceptive reviews.

These visualizations offer clear insights into the distinct language patterns associated with deceptive and truthful reviews. By analyzing the frequency and context of these terms, we can better understand how machine learning models can be trained to differentiate between the two categories effectively.

Fig. 8. Word Cloud for Truthful Reviews

Fig. 9. Word Cloud for Deceptive Reviews

C. Evaluation Metrics

The performance of all models was evaluated using:

- **Accuracy**: Proportion of correctly classified reviews.
- **Confusion Matrix**: Detailed insights into:
  - True Positives (TP): Correctly predicted deceptive reviews.
  - True Negatives (TN): Correctly predicted truthful reviews.
  - False Positives (FP): Truthful reviews misclassified as deceptive.
  - False Negatives (FN): Deceptive reviews misclassified as truthful.

VI. Conclusion

This project successfully demonstrated the potential of machine learning in detecting deceptive online reviews. By analyzing linguistic patterns and utilizing preprocessing techniques, the study contributes to enhancing consumer trust in review platforms.

Acknowledgment

The author thanks the contributors of the "Deceptive Opinion Spam Corpus" for providing the dataset and the resources used in this study.

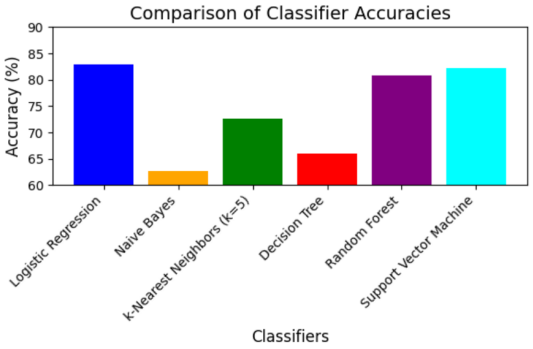

Fig. 10. Comparison of classifier accuracies

REFERENCES

[1] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," *Proc. 49th Annu. Meeting Assoc. Comput. Linguist.: Human Lang. Technol.*, Portland, OR, USA, Jun. 2011, pp. 309–319. DOI: 10.3115/2002472.2002512.

[2] M. Ott, C. Cardie, and J. T. Hancock, "Negative deceptive opinion spam," *Proc. 2013 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol.*, Atlanta, GA, USA, Jun. 2013, pp. 497–501. DOI: 10.5555/1620686.1620691.

[3] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retriev.*, vol. 2, no. 1–2, pp. 1–135, 2008. DOI: 10.1561/1500000011.

[4] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," *Proc. Eur. Conf. Mach. Learn. (ECML)*, Chemnitz, Germany, Apr. 1998, pp. 137–142. DOI: 10.1007/BFb0026683.

[5] Cornell NLP Group, "Deceptive Opinion Spam Corpus," accessed Dec. 2024. [Online]. Available: https://www.cs.cornell.edu/people/pabo/movie-review-data/