Evaluating the Impact of Data Preprocessing Techniques on Sentiment Analysis of TripAdvisor Reviews

36650570
Lancaster University
School of Computing and Communications
SCC-413: Applied Data Mining

Abstract

This research report explores the impact of various data preprocessing techniques on sentiment analysis of hotel reviews from TripAdvisor. It in specific examines the roles of stemming and lemmatization in preparing text data for machine learning algorithms that predict user sentiments and hotel ratings. The study evaluates different machine learning models to determine the best approach for classifying ratings effectively. It also investigates the influence of n gram strategies on model performance. The findings suggest that simple preprocessing methods, when combined with logistic regression, can yield great results in classification.

1 Introduction

1.1 Problem Setting

Hotels play a pivotal role in the tourism sector. As travelers increasingly rely on online reviews to make informed decisions, understanding the sentiments expressed in these reviews becomes crucial. The primary challenge lies in accurately interpreting the large amount of textual data, which requires effective data preprocessing methods to convert raw text into a structured format more amenable to machine learning algorithms.

1.2 Motivation

In the digital age the hospitality industry relies on online reviews and platforms like TripAdvisor offer valuable insights into customer satisfaction. Analyzing these reviews provides not just the hotel managers but also the customers with important information about the quality of services.

1.3 Research Question

This study aims to address the research question "What is the impact of data preprocessing on the sentiment analysis of TripAdvisor reviews?"

1.4 Contributions

This research makes two main contributions:

- It evaluates the influence of various data preprocessing techniques and n gram strategies on the performance of a sentiment analysis model.
- It compares multiple machine learning models to determine the most effective one for classifying the hotel ratings from TripAdvisor reviews.

2 Related Work

Research in sentiment analysis has explored a multitude of preprocessing techniques, revealing a complex landscape where the impact of these methods varies. Sharma et al., 2017 observed that the removal of stopwords, URLs, and punctuations, along with lowercasing, significantly improved the accuracy of classification in diverse Twitter datasets. This finding was corroborated by Ghag and Shah, 2015, who noted the effectiveness of stopword removal in traditional sentiment classifiers but not in others.((Işik and Dağ, 2020)

Krouska et al., 2016 found that the choice of feature representation, particularly 1-to-3 grams, was important in classification accuracy. Mat Zin et al., 2017 found that preprocessing strategies such as stopwords, numbers and punctuations proved that preprocessing affected the performance of the classification in a positive manner.

Whereas, Schofield et al. challenged the benefits of common preprocessing steps like stemming and stopword removal, advocating for a tailored approach depending on the application.

These studies highlight that there is robust research on different types preprocessing techniques and disagreements on their impacts as well. We will be focussing on impact of stemming, lemmatization and n-gram stratergies as well.

3 Data

3.1 Data Description

The dataset employed in this study comprises 20,000 reviews from TripAdvisor, including both the review texts and their corresponding ratings. Each entry in the dataset consists of a textual review along with a numerical rating, which provides a direct measure of customer sentiment towards their experience. This dataset offers an overview of customer opinions and sentiments, making it ideal for sentiment analysis.

3.2 Data Sourcing

The data was sourced from the work of Alam et al., 2016 in their study titled *Joint multi-grain topic sentiment: modelling semantic aspects for online reviews*. The dataset is publicly available under a CC-BY-4.0 license on Zenodo [1], ensuring it can be freely used for academic and research purposes. With over 11,000 downloads and citations in three different scholarly papers by Trivedi, Shrawan Kumar et al., Puh, Karlo & Bagić Babac, Marina, and Sarkar, Manash et al., this dataset has proven to be a valuable resource for research in sentiment analysis and natural language processing.

3.3 Justification of Suitability

This dataset is suitable for this research due to its focus on the hospitality industry, which is a sector where customer feedback plays a great role in influencing potential customers' decisions. The 20,000 individual reviews, allows for robust analysis and will help us see the impact of various data preprocessing and machine learning techniques. Moreover, the combination of textual reviews with numerical ratings facilitates us making a classification model using sentiment analysis.

3.4 Limitations

The dataset has some limitations that need our consideration. The reviews are all from TripAdvisor, which does not fully represent all demographics of hotel customers. This limitation could potentially introduce a bias towards the opinions and sentiments of TripAdvisor's user base, which may differ from those of other travel and hotel review platforms. Additionally, the dataset's fixed size and scope may not capture all the nuances of hotel service quality across different locations and time periods. These limitations should be kept in mind when interpreting the results of the analysis.

4 Methodology

The methodology is structured in four sections preprocessing comparison (stemming versus lemmatization), visualization through word clouds, model comparison for predictive accuracy, and comparison of different n-gram strategies. But first we converted the numerical ratings into three categories(positive, negative and neutral). Then converted all text to lowercase to maintain uniformity, removed all numbers using regular expressions, removed all punctuation marks and using NLTK's stopwords list filtered out the common English stopwords. This was done in order to clean the text before comparing and contrasting the coming preprocessing steps.

4.1 Stemming versus Lemmatization

Stemming reduces words to their root form (Lovins), while lemmatization converts words into their dictionary form (Balakrishnan and Lloyd-Yemoh, 2014). Our objective was to identify which method more effectively supports the accuracy of subsequent text classification models.

We used the text review data and each entry was preprocessed by using both techniques separately. TF-IDF vectorizer is then used to transform the processed text into a format suitable for model input. The preprocessed data was then fed into a logistic regression model to benchmark their impact on model accuracy.

Method	Accuracy
Baseline	0.856794
Stemming	0.857770
Lemmatization	0.856794

Table 1: Accuracy comparison between stemming and lemmatization

Results as shown in Table 1 tell us that stemming yielded a marginally higher accuracy compared to lemmatization which actually did not improve on the baseline accuracy at all. This outcome led to us selecting stemming as our primary preprocessing method for further experiments.

4.2 Word Cloud Generation

Following the selection of stemming as our preprocessing method, we generated word clouds to

[1] https://doi.org/10.5281/zenodo.1219899

visualize the most frequent terms in our dataset. This step was to ensure that the stemming process did not distort key terms' recognizability. Thus as can be seen in Figure 2 word clouds provided a visual confirmation that stemmed terms maintained their relevance and context within the corpus.

4.3 Model Comparison

We then compared different machine learning models to determine the best performer for our classification task. We tested four models: Decision Tree, Logistic Regression, Support Vector Classifier (SVC), and Random Forest. Each model was trained on the stemmed data after using TF-IDF vectorizer, and their performance was evaluated based on accuracy.

Model	**Accuracy**
Decision Tree	0.738961
Logistic Regression	0.856794
SVC	0.854355
Random Forest	0.786777

Table 2: Accuracy of different machine learning models

The results showed that Logistic Regression performed the best, closely followed by SVC. Decision Tree and Random Forest models lagged behind. The superior performance of Logistic Regression led us to adopt it for the last phase of our methodology.

4.4 N-gram Strategy

The set of co-occurring words present within the text is known as N-gram. In order to develop features for the supervised machine learning models, N-gram is utilized as mentioned in Kaur et al., 2018. Thus in the last phase, we experimented with different n-gram strategies to explore their impact on the accuracy of the Logistic Regression model. We tested six configurations: unigrams, bigrams, unigrams+bigrams+trigrams like in Krouska et al., 2016 and unigrams, bigrams, unigrams+bigrams+trigrams with all features and their performance was evaluated based on accuracy and training time. The difference between n-grams with all features and the ones without it is that those n-grams have a set maximum features (1000). This means the TF-IDF vectorizer will consider only the top 1000 features, limiting features should reduce the training time.

5 Results

The experiments conducted as part of this study yielded both quantitative data which we summarized in tables, and qualitative visualizations in the form of graphs.

5.1 Stemming vs Lemmatization

Table 1 compares the accuracies of using Stemming vs lemmatization with the baseline. Stemming was the slightly more accurate model with lemmatization not showing any measurable improvement over the baseline.

5.2 Model Accuracies

Figure 3 presents a bar chart comparing the accuracies of different machine learning models. Logistic Regression emerged as the most accurate model.

5.3 N-gram Strategies

Figure 1, Figure 4 and the Table 3 shows the impact of different n-gram strategies on the accuracy and training times of the Logistic Regression model when stemming is applied. The use of unigrams with all features offered the highest accuracy but uses four times the amount of time for a marginal improvement in accuracy as compared to unigrams+bigrams+trigrams strategy.

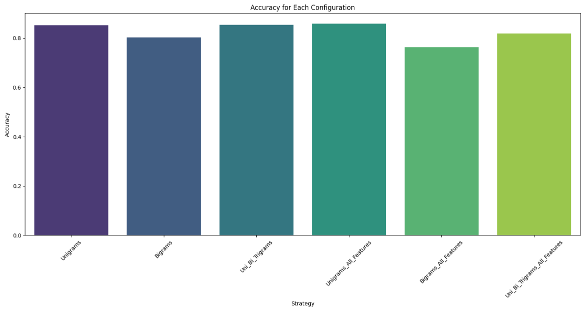

Figure 1: Accuracy for each N-gram Strategies

Model	**Accuracy**	**Testing Time (s)**
Unigrams	0.851427	0.289575
Bigrams	0.802391	0.134584
Uni_Bi_Trigrams	0.853867	0.256621
Unigrams_All_Features	0.857770	1.070697
Bigrams_All_Features	0.762381	5.914545
Uni_Bi_Trigrams_All_Features	0.818736	28.579087

Table 3: Accuracy and Testing Time of Different N-Gram Strategies

6 Findings

6.1 Comparison of Preprocessing Techniques

Our experiments revealed that stemming resulted in a slightly higher accuracy (85.777%) in text classification tasks compared to lemmatization (85.679%). This marginal difference suggests that for the purposes of our study, the aggressive truncating approach of stemming was sufficient for reducing words to their base forms, without the nuanced understanding of language that lemmatization provides. These results align with other findings in the field, such as those by Işik and Dağ, 2020, in whose paper it can be seen that when a linear regression model is preprocessed with stemming it performed better than when using lemmatization. It also validates Camacho-Collados and Pilehvar, 2018 who suggested that simpler tokenization methods are more impactful than complex preprocessing techniques like lemmatization.

6.2 Model Performance

The Logistic Regression model outperformed the Decision Tree, SVC(support vector classification), and Random Forest in classification accuracy. Our Logistic Regression model achieved an accuracy of 85.568%, validating that simpler linear models may sometimes be more effective for NLP tasks, as suggested by the results in IPI, 2022.

6.3 N-gram Analysis

The exploration of n-gram strategies revealed that our findings are consistent with those of Işik and Dağ, 2020, who found that the unigrams+bigrams+trigrams have the highest accuracy but unlike them we found that bigrams are not as accurate for us as unigrams. Especially unigrams when using all features as they produced the highest accuracy (85.777%) but the training time needed for it is four times as much as compared to unigrams+bigrams+trigrams (85.38%) for a mariginal improvement. As shown in Fig 4 the time required also increases exponentially for bigrams and unigrams+bigrams+trigrams with all features.

7 Conclusions and Future Work

7.1 Conclusions

Our research concluded that stemming, when used as a preprocessing step for sentiment analysis for our data, can slightly outperform lemmatization without compromising the interpretability of our text.

Furthermore, the Logistic Regression model provided the most accurate predictions among the models tested, reinforcing its suitability as a great baseline for text classification tasks. This finding suggests that complex models do not always translate to better performance in textual data classification though this may be related to the nature of the data.

The unigram strategy, especially when combined with a complete feature set, emerged as the most effective approach for our model but we have to keep in mind that it is not as efficient, indeed it is four times slower as compared to unigrams+bigrams+trigrams startergy for a very small improvement and thus may not be useful for a more complex data which would take even more time.

These conclusions underline the importance of selecting appropriate preprocessing techniques and models based on the nature of the text data as we know other researchers have had different outcomes based on their own data.

7.2 Implications

The findings from our study have a couple of implications for instance, they support the continued use of simpler models in certain contexts, which can benefit practitioners looking for efficient and transparent solutions. Moreover, our work suggests that while advanced models are appealing, for some data they may not always lead to practical improvements in model performance and can sometimes even detract from it.

7.3 Limitations

This study has limitations that must be acknowledged. The data used is limited in scope as mentioned before and the selection of models and other preprocessing methods was also constrained by project timelines and the length of the report. Furthermore, the performance metrics were limited to accuracy, which may not capture the nuances of model performance for the data.

7.4 Future Work

Future research could explore the application of more advanced NLP techniques like deep learning models, which might be able to capture subtleties in the data that simpler models cannot. It should also explore the impact of preprocessing techniques on varied text data and assess the scalability of these methods for broader real-world application and use more performance metrics.

References

2022. Sentiment analysis compare linear regression and decision tree regression algorithm to determine film rating accuracy. *Sean Institute*, Vol. 10 No. 02 (2022): Juni, Data Mining, Image Processing, and artificial intelligence.

Md. Hijbul Alam, Woo-Jong Ryu, and SangKeun Lee. 2016. Joint multi-grain topic sentiment: modeling semantic aspects for online reviews. *Information Sciences*, 339:206–223.

Vimala Balakrishnan and Ethel Lloyd-Yemoh. 2014. Stemming and lemmatization: A comparison of retrieval performances. pages 174–179, Seoul, Korea.

Jose Camacho-Collados and Mohammad Taher Pilehvar. 2018. On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 40–46, Brussels, Belgium. Association for Computational Linguistics.

Kranti Vithal Ghag and Dr Ketan Shah. 2015. Optimising Sentiment Classification using Preprocessing Techniques. 8(2).

Muhittin Işik and Hasan Dağ. 2020. The impact of text preprocessing on the prediction of review ratings. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, 28(3):1405–1421.

Sumandeep Kaur, Geeta Sikka, and Lalit Kumar Awasthi. 2018. Sentiment Analysis Approach Based on N-gram and KNN Classifier. In *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pages 1–4.

Akrivi Krouska, Christos Troussas, and Maria Virvou. 2016. The effect of preprocessing techniques on Twitter sentiment analysis. In *2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–5.

Julie Beth Lovins. Development of a stemming algorithm.

Harnani Mat Zin, Masrah Murad, and Nurfadhlina Sharef. 2017. The effects of pre-processing strategies in sentiment analysis of online movie reviews. volume 1891, page 020089.

Alexandra Schofield, Måns Magnusson, Laure Thompson, and David Mimno. Understanding text preprocessing for latent dirichlet allocation.

Palash Sharma, Aishwarya Agrawal, Lalit Alai, and A Garg. 2017. Challenges and techniques in preprocessing for twitter data. *International Journal of Engineering Science and Computing*, 7(4):6611–6613.

8 Appendix

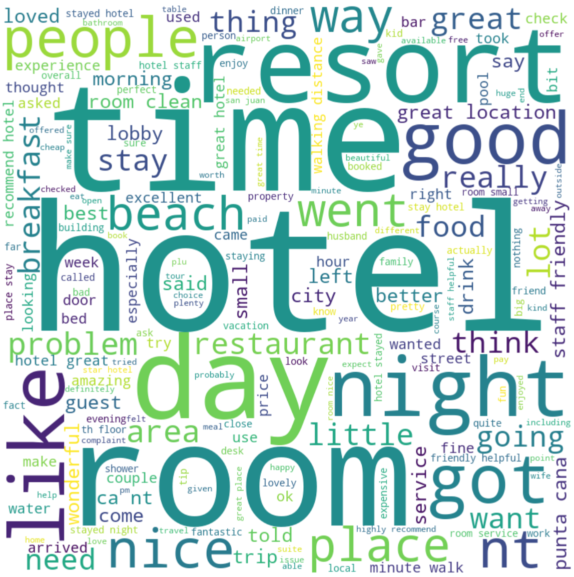

Figure 2: Word cloud

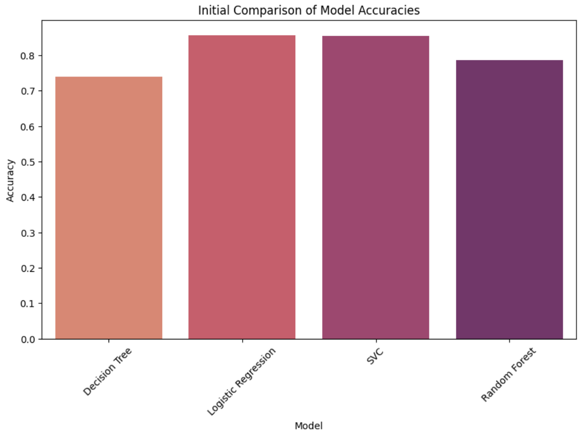

Figure 3: Initial Comparison of Model Accuracies

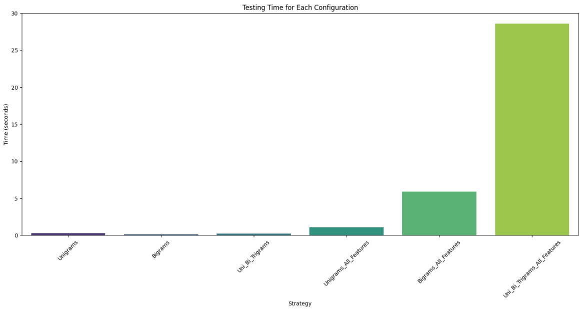

Figure 4: Training time for each N-gram Strategies