|  |
| --- |
| **Amazon Alexa Review**  **Technical Report**  amazon-header-s-min |

|  |
| --- |
| Distributed & Scalable Data Engineering |

|  |  |
| --- | --- |
|  |  |

Submitted by PADMAJA PHADKE

Submitted for

DSCI-6007-01

DR. ARDIANA SULA

Contents

[Introduction ........................................................ 3](#_Introduction)

[Executive Summary …………………................ 3](#_Executive_Summary)

[Background Theory ………………................... 4](#_BACKGROUND_THEORY)

[Exploratory Data Analysis .................................. 8](#_Exploratory_Data_Analysis)

[Methodology ........................................................ 4](#_Methodology)

[Results Section ....................................................... 5](#_Toc96341556)

[Conclusion ……………………………………... 6](#_Toc96341558)

[References ……………………………………… 7](#_Toc96341559)

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| Introduction GitHub link: <https://github.com/phadkep/Data_Distributed>  In the rapidly evolving landscape of e-commerce, customer reviews play a pivotal role in shaping product perceptions and influencing purchasing decisions. Understanding the sentiments and patterns within these reviews is essential for businesses aiming to enhance customer satisfaction and refine their products. The dataset under investigation, sourced from Amazon Alexa reviews, offers a rich tapestry of customer sentiments and preferences. This technical report embarks on a comprehensive analysis of the dataset, leveraging various techniques and methodologies to unveil insight that can inform strategic decisions.  Textual data, a valuable source of customer opinions, undergoes a rigorous pre-processing phase. Stop words are identified and removed, punctuation is addressed, and the dataset is refined to facilitate meaningful text analysis. We delve into the linguistic landscape by employing techniques such as word frequency analysis and word cloud visualization, unravelling the most prevalent terms within the reviews.  Beyond mere linguistic exploration, the report navigates through the realm of sentiment analysis. Machine learning models, including Naive Bayes, Random Forest, and Decision Tree classifiers, are harnessed to predict sentiments based on the review texts. Hyperparameter tuning enhances the models' performance, adding a layer of sophistication to the sentiment prediction process. | | |
|  |  |  |

## Executive Summary

This executive summary encapsulates the key findings and implications derived from the comprehensive analysis of the Amazon Alexa reviews dataset. The investigation spans exploratory data analysis (EDA), text analysis, and sentiment prediction, with a focus on unravelling customer sentiments and patterns within the review.

The dataset, comprising Amazon Alexa product reviews, is robust and well-structured, containing essential columns such as rating, date, product variation, feedback, and the actual review text. With no missing values, the dataset forms a solid foundation for subsequent analyses.

## 

## BACKGROUND THEORY

The goal of the project is to conduct a thorough analysis of Amazon Alexa customer reviews, extracting valuable insights from textual data. The primary focus is on sentiment analysis, aiming to predict whether a review expresses positive or negative feedback. Three machine learning models are employed for sentiment prediction: Naive Bayes, Random Forest, and Decision Tree classifiers. The analysis encompasses exploratory data exploration, including rating distribution and temporal trends. Text pre-processing techniques, such as stop words removal and punctuation handling, are utilized to refine the review texts. Additionally, the Tfi-dfVectorizer is employed for feature extraction, translating the textual data into numerical vectors. Hyperparameter tuning using Research fine-tunes the models, optimizing their predictive performance. The holistic approach integrates statistical analysis and machine learning methodologies to distil actionable business insights from customer reviews.

## Exploratory Data Analysis

**Analysis of data:**

We are getting information of every column.

A screenshot of a computer

Description automatically generated

We are describing the rating and feedback column. We can find the count, mean, std, min,25%, 50%, 75%, and max.

A screenshot of a computer

Description automatically generated

**Cleaning of data:**

Here, we are doing some data cleaning. In original dataset, we have stop words and we are removing it. This is the snap after removing all stop words from “verified\_reviews” column.

A screenshot of a computer

Description automatically generated

**Visualization of data**

Distribution of Ratings:

A graph with blue squares

Description automatically generated

Distribution of Ratings by Variation

A graph with blue and white lines

Description automatically generated with medium confidence

Distribution of Feedback

A graph with a blue and red square

Description automatically generated

Feedback vs. Rating

A screenshot of a survey

Description automatically generatedMonthly Review Count

A graph with a line going up

Description automatically generated

Top 10 Most Frequent Words

A graph of a number of blue rectangular objects

Description automatically generated with medium confidence

Top 5 Most Loved Products Based on Average Rating

A graph showing a bar chart

Description automatically generated with medium confidence

Most frequent words in positive reviews

A close up of words

Description automatically generated

Most frequent words in negative reviews

## A close-up of words Description automatically generated

## Methodology

**Data collection:**

The dataset used in the previous question includes a CSV file Which contains the reviews by the various customers and in the csv file we Can see the ratings given by them for the ALEXA.

The CSV file has five columns:

“rating”: Given by the customer.

“date”: The day when the review was given.

“variation”: Here it shows the information about the different colours and different versions of ALEXA.

“Verified reviews”: Here the reviews are given by the customers and then. It is stored.

“Feedback”: Here the feedback column has the information that how Much feedback given by the customers.

Algorithms:

We have employed three machine learning algorithms for our task: Naive Bayes Classifier, Random Forest Classifier, and Decision Tree.

1. Naive Bayes Classifier
   * Naive Bayes is a probabilistic machine learning algorithm based on Bayes' theorem, utilizing probability theory for classification tasks.

* Beyond text classification, Naive Bayes is applied in various fields, including email filtering, medical diagnosis, and document categorization. It adapts well to diverse domains and is particularly effective with large datasets.
* It assumes feature independence, meaning the presence of a particular feature is considered independent of the presence of other features.
* Widely used in text classification tasks like spam detection and sentiment analysis due to its efficiency with high-dimensional data.
* Known for its simplicity and computational efficiency, requiring a small amount of training data for effective model building.

A screenshot of a computer

Description automatically generated

1. Random Forest Classifier:
   * Random Forest is an ensemble learning algorithm that builds a multitude of decision trees during training and merges them for more accurate and stable predictions.
   * Provides a measure of feature importance, helping in identifying the most influential features in the dataset.
   * It relies on the foundation of decision trees, where each tree independently contributes to the final prediction.
   * During the construction of each tree, a random subset of features is considered for each split, enhancing diversity and reducing overfitting.
   * Random Forest employs a technique called bagging (Bootstrap Aggregating), creating multiple subsets of the training data to train individual trees.
   * Known for its high accuracy and robustness against overfitting, making it effective for complex datasets and minimizing the risk of individual tree biases.
   * Applied in various domains, including classification and regression tasks, due to its adaptability and ability to handle diverse data structures.

A screenshot of a computer screen

Description automatically generated

1. Decision Tree
   * A Decision Tree is a tree-like structure where each node represents a decision based on a particular feature.
   * The tree structure involves hierarchical decision-making, with the root node making the initial decision and subsequent nodes refining those decisions.
   * At each node, the dataset is split based on a chosen feature and a corresponding criterion, such as Gini impurity or information gain.
   * Decision Trees are used for both classification and regression tasks, making predictions by traversing the tree from the root to a leaf node.
   * Known for its interpretability, Decision Trees provide a clear and visual representation of decision-making processes, making them accessible for non-experts.
   * Decision Trees are prone to overfitting, capturing noise in the training data. Techniques like pruning are often applied to mitigate this issue.
   * Decision Trees form the basis for ensemble methods like Random Forests, combining multiple trees to improve accuracy and robustness.
   * Decision Trees provide a measure of feature importance, indicating the contribution of each feature to the decision-making process.

A screenshot of a computer screen

Description automatically generated

## 

## Results Section

The Amazon Alexa review dataset served as the training ground for three distinct algorithms—Naive Bayes, Decision Tree, and Random Forest—aiming to discern product quality based on customer reviews. The Naive Bayes algorithm demonstrated a test set accuracy of 0.9095, the Decision Tree algorithm achieved 0.9333, and the Random Forest algorithm excelled with an accuracy of 0.9413. These findings suggest a marginal advantage for the Random Forest in effectively identifying product quality within this dataset.

It's crucial to consider various factors when assessing classifier performance, such as precision, recall, F1 score, and the trade-offs between false positives and negatives. Furthermore, exploring alternative techniques, like adjusting hyperparameters, employing diverse feature engineering methods, or experimenting with alternative classification algorithms, could enhance classifier accuracy. These results serve as a preliminary exploration into the effectiveness of different techniques for product quality detection in the Amazon Alexa review dataset, emphasizing the importance of meticulous evaluation and comparison of diverse machine learning approaches.

The code is hosted on GitHub:

<https://github.com/phadkep/Data_Distributed>

## 

## Conclusion

In summary, an Amazon Alexa review classifier is a machine learning model employing natural language processing techniques to assess the quality of Amazon Alexa products. Building the classifier involves utilizing various algorithms like decision trees, Naive Bayes, and random forest, necessitating a meticulously labeled dataset of Amazon Alexa reviews.

The classifier's accuracy is contingent on factors such as the dataset's quality and size, algorithm selection, and hyperparameter choices. While achieving 100% accuracy is challenging for any classifier in product evaluation, the development of Amazon Alexa review classifiers serves as a valuable tool in indicating the actual quality of Amazon Alexa products to users.

## References

[https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/#:~:text=A%20decision%20tree%20algorithm%20is,each%20node%20of%20the%20tree.](https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/" \l ":~:text=A%20decision%20tree%20algorithm%20is,each%20node%20of%20the%20tree.)

<https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>

<https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/>

<https://www.kaggle.com/datasets/sid321axn/amazon-alexa-reviews>