Documentation Report

On

**Text Classification**

Submitted by

Name: **Yash Bedagkar**

Mentored by

Sudheer Kumar Yedunuri

**Infosys Springboard Internship Program**

**Year 2024**

**Index**

|  |  |  |
| --- | --- | --- |
|  | Title | Page No |
|  | 1. Abstract 2. Introduction 3. Sentiment Analysis for Text Data Understanding | 3  3  4 |
|  | 1. Dataset Description 2. Data Preprocessing 3. Vectorization and Data Splitting 4. Handling Class Imbalance using SMOTE 5. Modeling Approach 6. Models used 7. Performing Hyperparameter Tuning 8. Choosing the best model and its evaluation | 5  5  6  6  7  8  9  9 |
|  | 1. Project Achievements 2. Conclusion | 11  12 |
|  |  |  |

**1 . Abstract**

This project aims to develop a model for classifying emotions in social media comments to understand customer sentiment regarding product launches. The problem addressed is the overwhelming volume of unstructured text data, which makes it challenging for businesses to gauge customer reactions promptly. By preprocessing the text data, vectorizing it using TF-IDF, and applying machine learning models like Logistic Regression, Naive Bayes Classifier, and XGBoost, we achieved high accuracy in emotion classification. The best-performing model, XGBoost, demonstrated significant improvements in detecting customer sentiment, enabling businesses to make data-driven decisions for product development, marketing, and brand management.

**2 . Introduction**

Background and Context of the Project

In today's digital age, social media platforms have become a vital channel for customers to express their opinions and emotions about products and services. Analyzing this unstructured text data can provide valuable insights into customer sentiment, helping businesses to understand public reactions and improve their offerings accordingly. However, the sheer volume and complexity of social media data pose significant challenges for manual analysis, necessitating automated and efficient methods for sentiment classification.Problem Statement

The primary challenge is to accurately classify the emotions expressed in social media comments related to product launches. Traditional methods are inadequate for processing large-scale text data, leading to delayed responses and missed opportunities to address customer concerns promptly. This project aims to develop a robust text classification model that can automatically identify and classify emotions in customer comments, providing real-time insights into customer sentiment.

Business Use Case

In today's competitive market, understanding customer sentiment about new product launches is crucial for businesses. By analyzing social media reactions, companies can gain real-time insights into how their products are perceived by the public. This information is invaluable for making informed decisions on product improvements, marketing strategies, and customer engagement initiatives.For example, consider the recent launch of the Apple Vision Pro, a highly anticipated augmented reality headset. Following the launch, social media platforms were flooded with user comments and reactions. By applying sentiment analysis to these comments, Apple can quickly gauge the overall sentiment and identify specific emotions associated with the product. This analysis can help Apple understand the aspects of the Vision Pro that customers love, such as its innovative features and design, as well as areas that may require improvement, like pricing or usability.

**3 . Sentiment Analysis for Text Data Understanding**

Objective: To familiarize with text data processing and sentiment extraction.Approach:Lexicon-based Sentiment Analysis: This approach uses predefined lists of words (lexicons) that are labeled with their corresponding sentiment (e.g., positive or negative). The analysis determines the sentiment of a text by looking up the sentiment of each word in the lexicon.Outcome: This approach provides a basic understanding of handling text data for sentiment analysis and sets the foundation for more advanced techniques.

**4 . Dataset Description**:

This labeled data provides a rich source for training and evaluating models to accurately classify the emotional tone of customer comments. The dataset 'Emotions\_training.csv' consists of 16,000+ text records, each labeled with one of six emotions:

* sadness
* joy
* love
* anger
* fear
* surprise

**5 . Data Preprocessing :**

Data preprocessing is the process of transforming raw data into a clean and usable format for analysis and modeling. It involves a series of steps to remove noise and inconsistencies, handle missing values, and convert data into a suitable format for machine learning algorithms. In text classification tasks, this typically includes converting text to lowercase, removing links, eliminating special characters, and more. Data preprocessing is crucial because real-world data is often messy, incomplete, and inconsistent. Preprocessing ensures that the data is clean, accurate, and ready for analysis, which helps improve the performance of machine learning models. Data preprocessing techniques used here are:

1.Lower Case: Convert all text to lower case to maintain uniformity and avoid case-sensitive mismatches.2.Remove Links: Eliminate URLs from the text as they do not contribute to sentiment.3.Remove Next Lines (\n): Replace newline characters with spaces to ensure proper formatting.4.Remove Words Containing Numbers: Exclude words that contain numbers, as they are unlikely to contribute to sentiment.5.Remove Extra Spaces: Strip out unnecessary spaces to clean the text.6.Remove Special Characters: Remove punctuation and special symbols that do not add meaning.7.Removal of Stop Words: Eliminate common words that do not contribute to sentiment (e.g., "and", "the").8.Stemming: Reduce words to their root form by removing suffixes (e.g., "changing" to "chang").9.Lemmatization: Reduce words to their base or dictionary form, considering context (e.g., "changing" to "change").

**6 . Vectorization and Data Splitting**

Vectorization is the process of converting text data into numerical representations using TF-IDF that can be used by machine learning algorithms.

TF-IDF converts text data into numerical representations that can be used by machine learning algorithms.TF (Term Frequency): Measures how often a word appears in a document.IDF (Inverse Document Frequency): Measures the importance of a word by evaluating how common or rare it is across all documents.

Data Splitting

The dataset is split into three subsets:Training Data (70%): Used to train the model.Testing Data (20%): Used to evaluate the model's performance during training.Validation Data (10%): Used for the final evaluation of the model to ensure its generalizability.

**7 . Handling Class Imbalance using SMOTE**

Handling Class Imbalance Techniques:

1. Resampling the Dataset
2. Algorithmic Techniques
3. Ensemble Techniques
4. Anomaly Detection
5. Data Augmentation

SMOTE is an oversampling technique used to address class imbalance by creating synthetic samples of the minority class. It generates new instances by interpolating between existing ones, effectively increasing the representation of the minority class without duplicating data.

How SMOTE Works:

1. Select Minority Class Instances:
2. Find Nearest Neighbors:
3. Generate Synthetic Samples:

It generates synthetic samples in the training set to balance the class distribution. SMOTE is applied only to the training data to avoid information leakage.

**8 . Modeling Approach**

* Train the model on the training data.
* Evaluate its performance on training and testing data.
* Perform hyperparameter tuning.
* Re-train with the best parameters and evaluate.
* Choose the best model based on performance metrics.
* Evaluate the best model on validation data.

**9 . Models used**

1 . Logistic Regression

* A simple and interpretable model that estimates probabilities using a logistic function.
* Effective for both binary and multiclass classification, providing clear insights into the influence of each feature.
* Logistic Regression is a statistical model used for binary and multiclass classification tasks.
* It predicts the probability of an instance belonging to a particular class using a logistic curve.
* The model is interpretable, providing coefficients that indicate the relationship between each feature and the target variable.
* Logistic regression outputs probabilities, which can be thresholded to obtain class predictions.
* It uses the sigmoid function to model the probability

2 . Naïve Bayes Classifier

* Naive Bayes is a simple and efficient classification algorithm.It's based on Bayes' theorem with the "naive" assumption of independence between features.
* Suitable for both binary and multiclass classification tasks.
* Works well with large datasets and is computationally efficient
* Robust to irrelevant features, as it assumes independence.
* Requires relatively small amounts of training data to estimate the parameters necessary for classification.
* Handles both continuous and discrete data types.
* Often used as a baseline model for text classification tasks.

3 . XGBoost

* XGBoost stands for eXtreme Gradient Boosting, a powerful and popular machine learning algorithm.
* It belongs to the ensemble learning methods, specifically boosting algorithms.
* Capable of handling both regression and classification tasks.
* Uses a gradient boosting framework, which sequentially builds multiple weak learners to create a strong learner.
* Employs decision trees as weak learners, typically shallow trees called "stumps" or "shallow trees."
* Implements a regularized objective function to control model complexity and prevent overfitting.
* Utilizes both a gradient descent optimization algorithm and a second-order approximation to the objective function for more efficient and accurate learning.
* Incorporates advanced features like column subsampling, row subsampling, and tree pruning to further enhance performance and generalization.
* Known for its high predictive accuracy and robustness against overfitting.

**10 . Performing Hyperparameter Tuning**

Hyperparameter Tuning: The process of optimizing the hyperparameters that govern the training of a machine learning model to achieve the best possible performance.Methods:- Grid Search: Evaluates all possible combinations of hyperparameters.- Random Search: Samples random combinations of hyperparameters.Process:- Define the search space for hyperparameters.- Select the search method (grid search or random search).- Evaluate the performance of different combinations.- Identify the best parameters that yield optimal performance.

**11 . Choosing the best model and its evaluation**

Choosing the best Model

* Based on the evaluation metrics, XGBoost is identified as the best model for this problem.
* XGBoost consistently performs well across all metrics, particularly excelling in F1 Score and Recall.
* The high F1 Score indicates a good balance between precision and recall, meaning the model is proficient in identifying positive instances while minimizing false positives and false negatives.
* The high Recall score signifies that the model effectively captures most of the actual positive instances, making it reliable for scenarios where missing positive instances (false negatives) would be costly.
* Overall, XGBoost's robust performance across these metrics demonstrates its suitability for handling the complexities of the given text classification task.

Evaluate on Training and Testing Data with Best Parameters FoundEvaluation Metrics:

* Accuracy:

Definition: The proportion of correctly classified instances out of the total instances.

Explanation: Accuracy measures how often the model correctly predicts the class labels. However, it can be misleading for imbalanced datasets, as it does not consider the distribution of different classes.

* Precision:

Definition: The proportion of true positive predictions out of all positive predictions made by the model.

Explanation: Precision indicates how many of the predicted positive instances are actually positive. It is crucial when the cost of false positives is high.

* Recall:

Definition: The proportion of true positive predictions out of all actual positive instances.

Explanation: Recall, also known as sensitivity or true positive rate, measures how well the model can identify all actual positive instances. It is important when the cost of false negatives is high.

* F1 Score:

Definition: The harmonic mean of precision and recall.

Explanation: The F1 Score provides a balance between precision and recall, especially useful for imbalanced datasets. It ensures that both false positives and false negatives are considered.

The best choosen model (XGBoost) when evaluated on validation data gives :  
Accuracy = 83%  
Precision = 79%  
Recall = 85%  
F1 score = 81%

**12 . Project Achievements**

* Understanding User Sentiment:

Quickly gauges customer reactions to product launches by analyzing social media comments. Classifying comments into emotions helps identify overall sentiment and key areas of feedback, providing immediate insights into customer perceptions.

* Product Improvement:

Emotional insights guide product development by highlighting features that evoke positive reactions and identifying pain points. This helps prioritize enhancements and make data-driven decisions to improve the product based on user feedback.

* Targeted Marketing:Emotion-driven insights enable personalized marketing campaigns. Understanding the emotional triggers of the audience allows marketers to craft messages that resonate more deeply, driving engagement and conversions. Positive emotions can be amplified, and concerns can be addressed directly.
* Brand Management:Continuous sentiment monitoring helps maintain a positive brand image. By proactively managing negative sentiments and responding swiftly to customer concerns, the business can build trust and loyalty, showing responsiveness and attentiveness to customer needs and issues.

**13.** **Conclusion:**

This project demonstrates the power of text emotion classification in extracting valuable insights from social media comments regarding product launches. By employing advanced data preprocessing techniques, vectorization, and handling class imbalance with SMOTE, we prepared the dataset for effective model training. The comparative analysis of three models—Logistic Regression, Naive Bayes Classifier, and XGBoost—revealed that XGBoost consistently performed best across key evaluation metrics.

XGBoost's robust performance, especially in F1 Score and Recall, ensured a balanced approach to precision and recall, making it ideal for real-world applications. The model's ability to quickly and accurately classify emotions aids businesses in understanding customer sentiment, guiding product improvements, and crafting targeted marketing strategies.

Furthermore, continuous sentiment monitoring enables proactive brand management, fostering customer trust and loyalty. By aligning project achievements with business objectives, the emotion classification model not only enhances decision-making but also drives overall business growth through improved customer satisfaction and engagement. This project underscores the significant impact of leveraging machine learning for sentiment analysis in enhancing a business's responsiveness and adaptability in a competitive market.