Sentiment Analysis Project – Project Report

1. Introduction

This project focuses on building a sentiment analysis model using the Amazon Alexa dataset. Sentiment analysis refers to the task of computationally identifying and categorizing opinions expressed in text, typically to determine whether the writer's attitude is positive, negative, or neutral. The dataset contains customer reviews for Amazon Alexa products such as Echo, Echo Dot, and Echo Show. The main objective of this project is to preprocess text reviews, extract meaningful features using TF-IDF, and train machine learning models to classify the sentiment of each review.

2. Dataset Overview

The dataset used for this project is the 'amazon_alexa.tsv' file. It includes the following key columns:

• 'verified_reviews' – Text review written by customers.
• 'rating' – Numerical rating provided by the customer.
• 'feedback' – Target label for sentiment (1 = positive, 0 = negative).

The dataset contains thousands of reviews which provide a good base for training supervised learning models. Text data is unstructured, so preprocessing and feature engineering play a crucial role.

3. Data Preprocessing Steps

Text data often contains noise that must be cleaned before feeding into a model. The following preprocessing steps were implemented:

1. **Lowercasing:** All text was converted to lowercase to ensure uniformity.
2. **Removing special characters:** Regular expressions were used to remove punctuation, numbers, and symbols.
3. **Tokenization:** Splitting text into individual words.
4. **Stopwords removal:** Eliminating common words that do not contribute much to sentiment (e.g., 'the', 'is', 'and').
5. **Vectorization using TF-IDF:** Converts text into numerical form by giving importance to unique words.

4. Feature Engineering – TF-IDF Vectorization

TF-IDF (Term Frequency–Inverse Document Frequency) is used as the primary feature extraction technique.

• **Term Frequency (TF):** Measures how often a term appears in a document.
• **Inverse Document Frequency (IDF):** Reduces the weight of commonly used words.

TF-IDF helps highlight important words that contribute to sentiment. This results in a high-dimensional sparse matrix which is well-suited for linear models.

5. Train-Test Split

The dataset was divided into training and testing subsets using an 80-20 split. The training set is used to train the model, while the testing set is used to evaluate generalization performance.

6. Machine Learning Models Used

Two machine learning algorithms were implemented:

1. **Logistic Regression:**
   - Best suited for high-dimensional sparse data.
   - Performs well on text classification tasks.
   - Uses sigmoid function for binary classification.

2. **Random Forest Classifier:**
   - Ensemble method of decision trees.
   - Handles nonlinearity but can struggle with sparse TF-IDF data.
   - Good for understanding feature importance.

7. Model Training & Evaluation

Both models were trained using TF-IDF features. Accuracy, precision, recall, F1-score, and confusion matrix were used for evaluation. Typically, Logistic Regression performs better in text classification tasks due to:

- Ability to handle sparse matrices efficiently.
- Lower chance of overfitting.
- Faster training time.

Random Forest may not perform as well because text data leads to thousands of features, making tree-based models less efficient.

8. Insights from the Results

The following insights were observed from the model performance:

- Customer reviews are mostly positive, causing class imbalance.
- Logistic Regression achieves higher accuracy due to the linear nature of text data.
- Words such as 'love', 'excellent', 'amazing' strongly indicate positive sentiment.
- Negative sentiment is often indicated by keywords like 'worst', 'bad', 'poor', 'disappointed'.
- TF-IDF vectorization provides an effective numerical representation of reviews.

9. Challenges Faced

Some challenges identified in the project include:

• **Class imbalance:** Majority reviews are positive, which can bias the model.
• **High-dimensional vectors:** TF-IDF creates thousands of features, which may slow training.
• **Context loss:** Bag-of-words methods like TF-IDF do not capture word order or deeper meaning.
• **Overfitting risk:** Especially with ensemble methods like Random Forest.

10. Possible Improvements

Future enhancements that can improve the model include:

• Using advanced language models like Word2Vec, GloVe, or BERT.
• Implementing deep learning models such as LSTM or Transformer networks.
• Applying SMOTE to address class imbalance.
• Hyperparameter tuning using GridSearchCV.
• Adding visualization of word frequencies and model performance metrics.

11. Conclusion

This sentiment analysis project demonstrates how machine learning can effectively classify customer emotions based on textual reviews. TF-IDF combined with Logistic Regression provides a strong baseline for sentiment classification. The project shows the importance of preprocessing, feature extraction, and selecting appropriate models for text data. The insights gained can help businesses understand customer satisfaction, enhance products, and make data-driven decisions.