**Name: Awais Amin**

**Project Report**

**Introduction:**

The following report focuses on sentiment analysis conducted on customer reviews (English) made on products available on Daraz Pakistan, an e-commerce platform. The primary objective is to classify these reviews into positive, negative, and neutral sentiments, providing valuable insights into customer perceptions and feedback. This report aims to present a comprehensive overview of the entire process, including data collection, labeling, feature selection and extraction, model development, testing, and results. Additionally, it addresses the problem statement, provides a literature review summarizing previous work in this field, discusses the limitations encountered during the analysis, suggests future improvements, and concludes with key findings and recommendations.

**Problem Statement:**

The problem at hand is to analyze customer reviews (English) on products available in Daraz Pakistan and classify them into positive, negative, or neutral sentiments. The aim is to detect the sentiment expressed in the reviews and provide valuable insights into customer perceptions and feedback. The focus is on identifying sentiments related to product experiences, such as quality, delivery, customer service, and pricing, among others.

Tagging in this report was done based on specific criteria established by the group in class. The reviews were analyzed, and the sentiment was determined based on the following factors:

**Positive sentiment:** reviews expressing satisfaction, appreciation, or positive experiences related to the product.

**For example:**

"I love this product! It works perfectly and exceeded my expectations."

“Very good quality wiper rubbers. Hope it will last longer. The seller is also very cooperative. Highly recommend 👍.”

**Negative sentiment:** reviews expressing dissatisfaction, disappointment, or negative experiences related to the product.

**For example:**

"I'm extremely disappointed with the poor quality of the product."

“It was not measuring temperature accurately I compared it with other thermometers but very bad result.”

**Neutral sentiment**: Reviews that do not strongly lean towards either positive or negative sentiment. These reviews may contain neutral observations, simple statements of fact, or opinions that do not express a clear positive or negative sentiment.

**For example:**

"The product arrived on time, but it was just as described."

"It's an OK product, nothing special."

By considering the overall sentiment expressed in the reviews and using these established guidelines, the group tagged the reviews as positive, negative, or neutral accordingly.

**Literature Review:**

**Summary:**

**"Sentiment Analysis of Roman Urdu on E-Commerce Reviews Using Machine Learning" by Bilal Chandio, Asadullah Shaikh, Maheen Bakhtyar, Mesfer Alrizq, Junaid Baber, Adel Sulaiman, Adel Rajab, and Waheed Noor [1].**

The research paper focuses on sentiment analysis of Roman Urdu language in the context of e-commerce reviews. The authors highlight the challenges posed by Roman Urdu text, such as irregularity, variant spellings, and the mixing of English and Urdu words.

The authors propose a human-coded dictionary-based stemming method to normalize Roman Urdu text and improve feature selection. They create a publicly available Roman Urdu dataset and develop an online system and API as a baseline model for future research. Various machine learning and deep learning approaches are evaluated for sentiment analysis, including SVM, KNN, Naive Bayes, Adaboost, Random Forest, Decision Tree, Logistic Regression, RCNN, LSTM, CNN, RNN, and GRU.

The results show that SVM with stemming outperforms other approaches with an accuracy of 68%. The positive sentiment class performs the best, while the neutral and negative classes show slightly lower performance. The study contributes by providing a comprehensive analysis of Roman Urdu sentiment and proposes an effective approach for sentiment classification in this language.

**"Sentiment Analysis and Product Review Classification in E-commerce Platform" by Mahmud Hasan Munna, Md Rifatul Islam Rifat, and A. S. M. Badrudduza [2].**

The paper addresses the challenges faced by e-commerce service providers in helping customers select reliable products based on reviews. The authors acknowledge the growing popularity of online shopping and the overwhelming nature of customer reviews for manual analysis.

The authors propose the use of Natural Language Processing (NLP) and two Deep Neural Network (DNN) models for sentiment analysis and product review classification. They develop specific DNN architectures and implement an autonomous system to save time and provide efficient services. The models are evaluated using accuracy, precision, recall, and F1-score.

The sentiment analysis model achieves high accuracies in training, validation, and testing. Similarly, the product review classification model demonstrates satisfactory accuracies across different stages. The study provides valuable insights for improving e-commerce services, enhancing customer experiences, and optimizing product selection in online shopping.

**"Amazon Product review Sentiment Analysis using BERT" by Yash Inaniya [3].**

The paper introduces the application of Natural Language Processing (NLP) and the BERT model for sentiment analysis in Amazon product reviews. The authors highlight the increasing relevance of NLP and sentiment analysis in customer feedback analysis.

The paper provides a step-by-step demonstration of implementing the sentiment analysis model using BERT. It includes details on installing necessary libraries, preprocessing the Amazon product review dataset, tokenizing the text, and building the model using TFDistilBertForSequenceClassification.

The trained model achieves a promising accuracy of 94.73% on the training set and 92% on the validation set in just two epochs. The paper highlights the effectiveness of BERT models in sentiment analysis for Amazon product reviews and suggests their superiority over traditional models like RNNs, considering their ability to capture contextual information effectively.

**Refrences:**

**[1]:** B. Chandio, A. Shaikh, M. Bakhtyar, M. Alrizq, J. Baber, A. Sulaiman, A. Rajab, and W. Noor, "Sentiment Analysis of Roman Urdu on E-Commerce Reviews Using Machine Learning," in Proceedings of the IEEE International Conference on Machine Learning and Applications.

**[2]:** M. H. Munna, M. R. I. Rifat, and A. S. M. Badrudduza, "Sentiment Analysis and Product Review Classification in E-commerce Platform," in Proceedings of the 2020 23rd International Conference on Computer and Information Technology (ICCIT), 19-21 December 2020.

**[3]:** Y. Inaniya, "Amazon Product Review Sentiment Analysis using BERT," [Online]. Available: [https://www.analyticsvidhya.com/blog/2021/06/amazon-product-review-sentiment-analysis-using-bert/].

**Selection:**

The class collectively decided to gather data for a research project. The objective was to collect reviews from an e-commerce platform, specifically Daraz. Each student was assigned the task of collecting 500 reviews individually. This ensured a comprehensive dataset for analysis.

**Data Design:**

To avoid duplication and maintain organization, the categories of products available on Daraz were allocated to different students in the class. Each student was responsible for collecting reviews for their assigned product category. This division ensured that all categories were covered, and data collection was efficient.

In the data design, there were no limitations imposed on the design of the data, such as word limits for the reviews. Students had the freedom to collect reviews of varying lengths, capturing a wide range of customer opinions and experiences. However, one specific requirement was that all reviews had to be in English, ensuring consistency and ease of analysis.

**Collection:**

The students were given the freedom to choose their preferred method of data collection. They had two options: using a web scraper or manual collection. Some students, including myself, opted to create their own web scraper to automate the data collection process. The web scraper code was shared among the students for those who chose this method.

In my case, I also developed a website specifically designed to aid in the labeling process. The website allowed me to label each review as either neutral, positive, or negative. This labeling was crucial for the subsequent analysis of the collected data. After labeling the reviews, I was able to export the labeled data into a CSV file for further analysis.

**Data Labeling:**

In our data labeling process, each student labeled their own collected reviews. To ensure consistency, we had thorough discussions and reached a consensus on how the reviews should be labeled within the class. The following guidelines and processes were established:

**Consensus on Labeling Criteria:**

As a group, we discussed in class with the help of our instructor and accepted the criteria for labeling reviews as positive, negative, or neutral. This involved considering factors such as sentiment expressed, overall tone, specific keywords, and the overall impression conveyed by the review. By collectively agreeing on the criteria, we aimed to ensure consistent labeling across the dataset.

**Individual Labeling:**

Each student labeled the reviews they collected, as they had the contextual knowledge required to accurately interpret the reviews. By labeling their own reviews, we minimized potential bias and maintained a closer connection between the labeled data and its collector.

**Consistent Labeling Format:**

In our discussions, we also addressed the labeling format to ensure uniformity and consistency. We agreed upon a standardized format, such as using specific labels like "positive," "negative," or "neutral." This format ensured clarity and facilitated easier analysis of the labeled data.

**Preprocessing:**

An analysis of the label values in the data revealed inconsistencies in spelling, such as incorrect capitalization. To address this, all labels were transformed to lowercase, and rows with misspelled labels were removed from the dataset.

**Feature Selection and Extraction:**

In the process of feature selection and extraction for this project, multiple iterations were carried out to refine the approach. Here is a breakdown of each iteration:

**Iteration 1:**

The initial plan was to use a technique similar to the one employed in the spam and ham assignment. The process involved replacing integers and decimals with the word ‘number’, removing alphanumeric tokens, converting letters to lowercase, performing tokenization, removing stop words and punctuation, and applying stemming. However, this approach did not yield satisfactory results for this project.

**Result**: The initial iteration did not perform well for the project's requirements.

**Adjustment for the Next Iteration:**

To improve the results, the following adjustments were made:

* Stop Word Removal: The removal of stop words was stopped in order to retain more contextual information.
* Vectorization removal: The application of a vectorizer was removed as it did not seem to provide a suitable representation of the textual data.

**Iteration 2:**

The adjusted plan included the removal of stopwords and the vectorization step. Additionally, it also involved the removal of emojis while keeping the text in lowercase form.

**Result**: The adjustments made in this iteration improved the performance of the feature extraction and selection process.

**Adjustment for the Next Iteration:**

Based on the improved results, the following adjustments were implemented:

* Lemmatization: Lemmatization was introduced as an alternative to stemming. This helped in obtaining better output.

**Iteration 3:**

In the final iteration, I incorporated the bag-of-words technique and parts of speech analysis. This improved the accuracy of the feature selection and extraction process. The bag-of-words approach captured word frequencies while considering parts of speech added linguistic context.

**Result**:

These adjustments resulted in higher classification accuracy.

By iteratively adjusting the feature selection and extraction techniques, the approach was refined and resulted in improved performance for the project. Each iteration aimed to address specific limitations observed in the previous iteration, leading to a more effective and reliable feature extraction and selection process.

**Model Development:**

In the process of model development, the following steps were followed:

**Initial Classifier Evaluation:**

Initially, a set of classifiers were evaluated using default hyperparameters. The classifiers included Logistic Regression, Support Vector Machines (SVM), Naive Bayes, Decision Tree, K-Nearest Neighbors (KNN), Random Forest, AdaBoost, Bagging, Extra Trees, Gradient Boosting, and XGBoost. However, none of these classifiers yielded satisfactory accuracy or F1 scores, with the highest achieved score being around 72%.

**Hyperparameter Tuning:**

To improve the performance of the better-performing classifiers, hyperparameter tuning was performed. The classifiers that showed potential were Extra Trees Classifier (ETC), XGBoost (XGB), Random Forest Classifier (RFC), and Support Vector Classifier (SVC). The following hyperparameters were tuned for these classifiers:

RFC: n\_estimators=500, max\_depth=None

SVC: kernel='sigmoid', gamma=1.0

ETC: n\_estimators=500, max\_depth=None, min\_samples\_split=5, random\_state=40

XGB: n\_estimators=1000, learning\_rate=0.01, max\_depth=10, subsample=0.9, colsample\_bytree=0.6

However, even after hyperparameter tuning, the results did not show significant improvement.

**Neural Network Exploration:**

To further enhance the performance, neural networks were explored. Different variations of neural network architectures were tested, including varying the number of layers, epochs, batch sizes, and layer types. Attention mechanisms were also incorporated to assess their impact on F1 score and accuracy. However, these modifications did not yield noticeable improvements.

**Transformation of preprocessed data to use in Neural networks:**

In my code I have transformed and preprocessed text data using the Tokenizer and pad\_sequences functions from the TensorFlow Keras library

Tokenizer is used to convert text into integer sequences by assigning each word an index based on a vocabulary built from the training data. The texts\_to\_sequences method applies this transformation to both the training and test data. pad\_sequences ensures all sequences have the same length by padding or truncating them to a specified length. These preprocessing steps prepare the text data for machine learning models that require fixed-length input sequences.

**Additional Layers:**

In the final attempt, additional layers were added to the neural network model. The model architecture consisted of an embedding layer, spatial dropout layer, convolutional layer, global max pooling layer, dense layer, and softmax output layer with epochs of 3 and batch size of 64. This configuration finally achieved a satisfactory accuracy of 88%.

By systematically trying different models with different hyperparameters, the model development process aimed to optimize the classification performance. Also, during the model development process, continuous analysis and improvement of the feature set were carried out. Different techniques were explored, including the introduction of part-of-speech (POS) tagging and bag-of-words representation. Lemmatization was also applied to further refine the feature set. These additions helped enhance the performance of the model, contributing to the final accuracy achieved.

**Testing & results:**

During the testing phase, the final trained model was evaluated on a separate test set to assess its performance. Both accuracy and F1 score were chosen as evaluation measures. The test set, which constituted 20% of the entire dataset, was used to provide an unbiased assessment of the model's capabilities.

After multiple iterations and adjustments in the model development process, significant improvements were observed in both accuracy and F1 score. The neural network architecture was enhanced by introducing specific layers mentioned earlier, such as embedding, spatial dropout, convolutional, global max pooling, and dense layers.

The model's architecture consisted of an embedding layer, which learned meaningful representations from the input data. The spatial dropout layer helped regularize the model by randomly disabling portions of the inputs during training, reducing overfitting. The convolutional layer extracted features through convolutional operations, capturing local patterns in the data. The global max pooling layer selected the most salient features from the convolutional layer's outputs. The dense layer, with its increased capacity, facilitated learning complex relationships within the data. Finally, the softmax output layer provided the probabilities of each class label.

Training the model with a suitable number of epochs (in this case, 3) and a batch size of 64 resulted in the best performance. The model was fine-tuned to optimize its parameters, leading to improved accuracy and F1 score.

**Classification Report:**

**precision recall f1-score support**

0 0.73 0.63 0.67 269

1 0.85 0.91 0.88 498

2 0.94 0.94 0.94 957

**accuracy** 0.88 1724

**macro avg** 0.84 0.83 0.83 1724

**weighted avg**  0.88 0.88 0.88 1724

In the classification report, the precision, recall, and F1-score are provided for each class label (0, 1, 2). The support column indicates the number of instances belonging to each class in the test set.

Overall, the model achieved an accuracy of 0.88, meaning it correctly classified 88% of the instances in the test set. The macro-average F1-score is 0.83, indicating the average performance across all classes. The weighted average F1-score is also 0.88, considering the class distribution in the dataset.

The model demonstrates good performance with high precision, recall, and F1-scores for classes 1 and 2. Class 2 has the highest performance, achieving a precision of 0.94, recall of 0.94, and F1-score of 0.94. Class 1 also shows strong performance with precision, recall, and F1-score of 0.85, 0.91, and 0.88 respectively.

However, class 0 has lower performance compared to the other classes, with a precision of 0.73, recall of 0.63, and F1-score of 0.67. This indicates that the model struggles relatively more with correctly identifying instances of class 0.

Overall, the model's performance is satisfactory, with high accuracy and F1-scores, demonstrating its ability to effectively classify instances across multiple classes.

On testing the model with an independent dataset, an F1 score of 88 was achieved, showcasing the model's ability to effectively classify instances. Furthermore, when evaluated on a separate dataset provided by the instructor during the viva examination, the model attained an F1 score of 87, indicating its generalization capabilities. However, it is worth noting that the overall result for class 0 was not as good, both in the independent dataset and the instructor-provided dataset, with lower precision, recall, and F1-score compared to the other classes. This discrepancy may be attributed to the fact that in the dataset, instances labeled as neutral or belonging to class 0 were not labeled by the students as "good," which might have impacted the model's ability to correctly classify them. Further investigation and improvements are necessary to address this issue.

These results validate the effectiveness of the feature selection, extraction, and model development approaches employed throughout the process. The inclusion of additional layers and careful parameter tuning contributed to the overall enhancement in accuracy and F1 score, demonstrating the model's ability to make accurate predictions and generalize well to unseen data.

**Limitations:**

* **Labeling Inconsistencies**: The labeling process was affected by inconsistencies and errors. Some students did not take the labeling task seriously, resulting in misspelled labels and deviations from the agreed-upon requirements.
* **Limited Data Quality:** The quality of the collected data may vary due to the lack of strict adherence to labeling guidelines. This could impact the overall accuracy and reliability of the dataset.
* **Language and Sentiment Variability:** English reviews vary in language use, writing style, and sentiment expression, making it challenging for sentiment analysis models to interpret nuances and colloquial expressions.
* **Domain-specific Language**: E-commerce platforms have unique jargon and terminology, posing difficulties for sentiment analysis models trained on general English language data.
* **Contextual Understanding:** Sentiment analysis models may struggle to grasp the specific context and references in reviews, affecting their ability to accurately analyze sentiment related to product issues, delivery experiences, or customer service interactions.

**Suggestions for Improvements:**

* **Enhanced Labeling Process:** Implement a more rigorous and standardized approach to labeling, including clear guidelines and training for the individuals involved. Regular quality checks and feedback sessions can help ensure consistency and accuracy in labeling.
* **Data Quality Assurance:** Implement measures to ensure data quality, such as conducting regular data audits, validating labels against established guidelines, and removing instances with significant labeling inconsistencies or errors.
* **Language and Sentiment Modeling:** Explore advanced natural language processing techniques that can handle language and sentiment variability more effectively. This may involve utilizing pre-trained language models, incorporating sentiment lexicons or emotion detection algorithms, and considering context-aware sentiment analysis approaches.
* **Domain-specific Adaptation**: Fine-tune sentiment analysis models using domain-specific data or incorporate domain-specific language resources to improve the understanding of e-commerce-related terminology and jargon.
* **Contextual Analysis:** Develop strategies to incorporate contextual understanding into sentiment analysis models. This can involve leveraging knowledge graphs, entity recognition, or topic modeling techniques to capture the specific context and references mentioned in reviews.

**Conclusion:**

The project focused on conducting sentiment analysis on customer reviews from Daraz Pakistan, an e-commerce platform. The process involved data collection, labeling, feature selection and extraction, model development, testing, and analysis of results. The data collection process ensured a comprehensive dataset by assigning students to collect reviews for specific product categories. However, labeling inconsistencies and deviations from guidelines affected the data quality. Suggestions for an enhanced labeling process and data quality assurance were provided.

The feature selection and extraction process underwent multiple iterations to improve accuracy. Adjustments such as removing stop words, lemmatization, and parts-of-speech analysis led to better results. Challenges remained in handling language and sentiment variability, as well as domain-specific language. Suggestions included advanced language and sentiment modeling techniques and domain-specific adaptation.

Model development involved exploring various classifiers and hyperparameter tuning, followed by the exploration of neural networks. The inclusion of additional layers in the neural network architecture improved accuracy. The final model was evaluated on a separate test set, demonstrating good accuracy and F1 score.

However, limitations were identified, including the need for improved contextual understanding and challenges in interpreting nuances and colloquial expressions. Suggestions for improvements included leveraging advanced natural language processing techniques and incorporating contextual analysis.

In conclusion, while this project made progress in sentiment analysis for e-commerce reviews, addressing limitations in labeling, data quality, language variability, domain-specific language, and contextual understanding is crucial. By implementing the suggested improvements, future sentiment analysis models can achieve higher accuracy and effectively capture customer sentiments expressed in e-commerce reviews.