Customer Feedback Analysis for Wex Photo Video

Aman Seth
100442755

August 22, 2024

Contents

1 Introduction **4**
1.1 Background and Motivation . . . 4
1.2 Aim and Objectives . . . 5
1.3 Difficulties and Risks . . . 6
1.3.1 Ethical, Legal and Social Issues . . . 6
1.3.2 Web Scraping Risks . . . 6
1.3.3 Language Nuances and Ambiguity . . . 6
1.3.4 Linguistic Variability . . . 6
1.4 Work Plan . . . 7

2 Related Work **7**
2.1 Text Sentiment Analysis using ML . . . 7
2.2 Text Sentiment Analysis using BERT . . . 8
2.3 Llama Models in Product Review Evaluation . . . 9
2.4 Summary . . . 10

3 Preparation & Design **11**
3.1 Analysis of the method . . . 11
3.1.1 Data Extraction . . . 11
3.1.2 Supervised ML Model . . . 11
3.1.3 BERT Transformer . . . 12
3.1.4 Issue Type Classification . . . 12
3.1.5 Reason behind issue text-generation . . . 12
3.1.6 Deploy Best Models on Web Application . . . 12
3.2 Tools and Resources . . . 13
3.3 Design of Methodology . . . 13
3.3.1 MoSCoW . . . 13
3.3.2 Stakeholders . . . 14
3.3.3 High-level Design Flowcharts . . . 15

4 Implementation **16**
4.1 Customer Reviews Extraction . . . 16
4.2 Data Pre-processing . . . 16
4.2.1 Feature Engineering . . . 16
4.2.2 Handling Missing Values & Data Balancing . . . 17
4.2.3 Emoji & Contractions Handling . . . 18
4.2.4 Special Characters & Other Symbols Handling . . . 19

4.2.5 STOP words & Lemmatization . . . 19
4.2.6 Duplicate Reviews Handling . . . 20
4.2.7 Remove Meaningless Reviews with Length <=2 & <=3 . . . 20
4.2.8 Remove More Meaningless Reviews . . . 20
4.2.9 Summary . . . 21
4.3 Machine Learning Models . . . 21
4.3.1 ML Low Level Design Flowchart . . . 21
4.3.2 ML Use-case Diagram . . . 22
4.3.3 ML Sequence Diagram . . . 23
4.4 BERT Models . . . 25
4.4.1 BERT Low Level Design Flowchart . . . 25
4.4.2 BERT Use-case Diagram . . . 26
4.4.3 BERT Sequence Diagram . . . 27
4.5 Web Application Deployment . . . 29

5 Testing 30
5.1 Train, Test and Validation Testing . . . 30
5.1.1 ML Models . . . 30
5.1.2 BERT Models . . . 34
5.2 Testing on Held-out External Test Set . . . 37
5.2.1 Deploying Best ML Models . . . 37
5.2.2 Deploying Best BERT Models . . . 37
5.3 A/B Testing(Split Testing) . . . 38
5.4 Regression & Integration Testing . . . 41
5.5 Web Application Usability Testing . . . 44

6 Evaluation and Discussion 46
6.1 Sentiment Predictions via ML . . . 46
6.1.1 Approach 1 . . . 46
6.1.2 Approach 2 . . . 47
6.1.3 Approach 3 . . . 49
6.1.4 Testing Best Approach on External Test Dataset . . . 51
6.1.5 Summary . . . 51
6.2 Sentiment Predictions via BERT . . . 51
6.2.1 Testing Best Approach on External Test Dataset . . . 52
6.2.2 Summary . . . 53
6.3 ML vs BERT . . . 53
6.3.1 Why distilBERT performed better than ML ? . . . 54
6.4 Issue Type Classification via spaCy PhraseMatcher . . . 55
6.5 Possible reason behind issues generation via Llama3 . . . 57
6.6 User Experience (UX) . . . 61

7 Areas of Improvement & Future Enhancements 61
7.1 Part A: Sentiment Prediction . . . 61
7.2 Part B: Issues Classification . . . 61
7.3 Part C: Reason Behind Issues Generation . . . 61
7.4 User Experience (UX) . . . 62

8 Conclusion 62

9 Appendix **64**
9.1 Cross Validations 64
9.1.1 Cross Validation Approach 1 64
9.1.2 Cross Validation Approach 2 66
9.1.3 Cross Validation Approach 3 69
9.1.4 Cross Validation Approach 4 71
9.2 User Experience (UX) 75
9.3 Project Management on Trello 77
9.4 Github Code Tracking & Maintenance 78

Abstract

The process of locating and categorizing viewpoints or feelings represented in source text is known as **sentiment analysis**. A significant amount of sentiment-rich data is being produced for *Wex Photo Video*, a UK-based e-commerce company, via different review websites and social media platforms such as Trustpilot, Power Reviews, Help Scout, Facebook and many more. Understanding customer opinions can greatly benefit from sentiment analysis of this user-generated data. However, *Wex Photo Video* reviews' sentiment analysis is more challenging than generic sentiment analysis due to the prevalence of misspellings and British slang phrases. This research aims to utilize a machine learning approach to assess reviews regarding various electronic devices, such as cameras, lenses, computers, and mobile phones. Specifically, Machine Learning (ML) and Deep Learning (DL) driven models are deployed to categorize reviews as positive, neutral, or negative and to extract user opinions regarding products. Our methodology involves pre-processing steps to handle misspellings, gibberish words, British slangs, other symbols and special characters etc. ensuring that the data is suitable for analysis. We have experimented with various ML models, alongside DL transformer models. Performance is evaluated using metrics such as confusion matrix, accuracy, precision, recall, and F1-score etc. By conducting sentiment analysis within this specific domain, we aim to determine the impact of domain information on sentiment classification. This study not only contributes to improving customer understanding for *Wex Photo Video* but also advances the field of sentiment analysis by addressing the unique challenges posed by domain-specific language.

1 Introduction

Sentiment analysis refers to the process of using Natural Language Processing (NLP) to ascertain whether a text conveys information that is subjective and what kind of information it has, i.e., whether the text's attitude is positive, negative, or neutral. Sentiment analysis combined with machine learning may be helpful in forecasting customer opinions of both recently released and current products as well as product reviews. Because opinions conveyed through reviews are frequently vague, it might be challenging to determine the correct opinion using simple computational models. Therefore, in order to effectively analyze the reviews response, computational methods based on natural language processing are needed. The majority of studies just look at word frequency in order to identify related terms that have a greater impact on the outcome. It's possible that this word frequency-based methodology misses terms that actually affect the result. It is necessary to comprehend significant connected events in relation to the main event inside the context in order to forecast the outcome of any event. [1].
In this paper, we try to analyse and predict sentiments behind the reviews for *Wex Photo Video* with assistance of different conventional **supervised learning classification models** and **Bidirectional Encoder Representations from Transformers (BERT)** which is an open source language model ML framework for NLP [2] .

1.1 Background and Motivation

Every day, *Wex Photo Video* receives more than 100 reviews from users of review websites and other social media platforms. During sporadic seasons like those around Christmas, New Year's, or Easter, the number rises noticeably. The Call Center Management team at Wex finds it incredibly difficult to go through all of the reviews from various platforms and determine the customer's sentiment behind them. The process takes a lot of time and is entirely manual.
Wex does not currently have an internal AI platform in place that can retrieve the needed reviews and carry out sentiment analysis using natural language processing (NLP), allowing them to get quick insights into customer comments. The absence of these abilities is the motivation behind the project.
Wex would be able to respond quickly to consumers who had written neutral or unfavorable feedback about their items, if they had such capabilities in place. They could use this information to inform larger business decisions. They may have increased customer engagement and retention rates as well.

1.2 Aim and Objectives

Aim: The aim of this research is to employ a machine learning / deep learning approach to evaluate customer reviews of various electronic devices sold by *Wex Photo Video* predicting sentiments behind them classifying them into 3 categories: *Positive, Negative, Neutral*. Furthermore, the aim is to classify the potential issues in the review to different categories: *Electronics, Delivery, Customer Service, Time, Packaging, General* through NLP techniques and predict the reason behind the classified issues using transformer model *Llama3*. Additionally, we seek to develop an on-demand web-based service that allows the *Wex* team to analyze these reviews in real-time and take appropriate actions based on the insights gained.

Objectives:

1. Scrape review websites for customer comments
   (a) Create per website scrapers to extract product reviews posted by *Wex Photo Video* customers.
   (b) Data extraction done by calling appropriate per review website API endpoints provided by the Wex engineering team.
   (c) Required reviews data consolidated in CSV files per review website.
2. Deploy Supervised learning models on extracted data
   (a) Collect the per review website data containing product reviews and feedback by customers.
   (b) Clean and balance the data to avoid biased results.
   (c) Pre-process the data for optimal utilisation when using conventional supervised learning machine learning (ML) models.
   (d) Develop a model which will efficiently provide sentiment labels (Positive, Neutral, Negative).
   (e) Measure and refine the model until a satisfactory quality has been accomplished.
3. Deploy BERT language model on extracted data
   (a) Preprocess the data for optimal utilisation for BERT [3].
   (b) Loading pre-trained BERT model
   (c) Develop a model which will efficiently provide sentiment labels (Positive, Neutral, Negative).
   (d) Measure quality and accuracy of the model.
4. Issue types behind reviews classification using NLP
   (a) Develop NLP techniques to identify the type or category of issue/s highlighted in the customer reviews.
   (b) Identified possible issue types are: Electronics, Delivery, Time, Customer Service, Packaging, General
5. Deploy llama3 transformer model to predict reason behind reviews containing issues
   (a) Integrate llama3 model to predict reasons behind the identified issues for the classified reviews.
   (b) The reason should be accurate and concise notifying the user if there was actually an issue within the review or not.
6. Deploy both Supervised Learning and BERT models on web application
   (a) Deploy models using open-source python packages such as Flask or Streamlit for better presentation and on-demand usability.

1.3 Difficulties and Risks

1.3.1 Ethical, Legal and Social Issues

Privacy concerns: A common requirement of NLP and sentiment analysis is the retrieval and examination of vast volumes of textual data, some of which may contain private information. This gives rise to worries regarding the security and privacy of the people whose data is being examined. It is crucial to guarantee that procedures for gathering and analyzing data are open, adhere to privacy laws, and protect user data [4].

Social Bias: NLP models and sentiment analysis algorithms might reflect the biases existing in the data they are trained on. This can lead to skewed findings that promote discrimination or unfairness against specific demographic groups. Addressing bias requires careful consideration of data collection, algorithm, and evaluation procedures to ensure fair outcomes. Ensuring that data collecting and analysis procedures are transparent, adhere to privacy standards, and protect user information is crucial [5].

Transparency and interpretability: Understanding how NLP models, particularly deep learning models, make predictions or classify sentiment can be difficult because these models are frequently referred to as "black boxes." In order to foster responsibility and confidence in NLP systems, transparency and interpretability are crucial. Researchers and practitioners need to create strategies to explain the judgments made by these models and make them interpretable to users [4].

Ethical data collection and labeling: The data sets must be ready for training in order to obtain a precise and accurate sentimental analysis model. There is huge possibility that the ethical issues might arise in the collected data. Hence it is extremely important that there is guaranteed consent to access and use this data in order to prevent any kind of biases. [6].

1.3.2 Web Scraping Risks

Before starting an automated scraping, web scraping is a problem that should be carefully considered from an ethical standpoint. Owing to ethical considerations, care must be taken to prevent flooding a website with too many requests in a short period of time. If these issues are not addressed, the website's hosting server may become overloaded and experience a general slowdown. This might be regarded as a denial-of-service (DoS) assault in worst situation. Anti-scraping techniques are often used on websites where a user must complete CAPTCHA in order to access data. If the user fails the CAPTCHA, their access to the website may be momentarily or occasionally permanently blocked. Additionally, personal information found on websites should never be scraped by a scraping program. Although the websites and data should reduce any personal information, care will be taken to ensure that no personal information is collected.

1.3.3 Language Nuances and Ambiguity

Detecting sarcasms in the customer reviews can be fairly challenging for NLP models as they might find it difficult to interpret the sentiment behind it. Such reviews may get classified into wrong category.

1.3.4 Linguistic Variability

There could be a possibility the reviews might contain British slangs which again the NLP models might find hard to interpret the actual sentiment behind them.

Risk	Priority	Impact
Ethical, Social & Legal	Low	Medium Severity
Web Scraping	Medium	High Severity
Language Nuances and Ambiguity	Medium	High Severity
Linguistic Variability	Medium	High Severity

Table 1: Risks Segregation

1.4 Work Plan

ID	Task	Start Date	End Date	Duration	Progress %
1	Defining Project Scope	Jun 03, 2024	Jun 04, 2024	2 days	100
2	Requirement Gathering	Jun 05, 2024	Jun 14, 2024	8 days	100
3	Data Extraction	Jun 17, 2024	Jun 21, 2024	5 days	100
4	ML - Feature Engg. & Preprocessing	Jun 24, 2024	Jun 28, 2024	5 days	100
5	Training ML Models	Jul 01, 2024	Jul 05, 2024	5 days	100
6	ML - Testing & Evaluation	Jul 08, 2024	Jul 12, 2024	5 days	100
7	Progress meet with Wex	Jun 19, 2024	Jun 19, 2024	0.5 days	100
8	Sentiment Classification per product	Jul 15, 2024	Jul 19, 2024	5 days	100
9	BERT - Data Preprocessing	Jul 22, 2024	Jul 26, 2024	5 days	100
10	BERT Model Implementation & Training	Jul 29, 2024	Aug 02, 2024	5 days	100
11	BERT - Testing & Evaluation	Aug 05, 2024	Aug 09, 2024	5 days	100
12	ML & BERT Comparison Analysis	Aug 12, 2024	Aug 16, 2024	5 days	100
13	Thesis Report	Jun 03, 2024	Aug 22, 2024	61 days	100
14	Project Management on JIRA	Jun 03, 2024	Aug 22, 2024	61 days	100
15	Progress meet with Wex	Jul 17, 2024	Jul 17, 2024	0.5 days	100
16	Progress meet with Wex	Aug 14, 2024	Aug 14, 2024	0.5 days	100
17	Web implementation using Flask / Streamlit	Aug 07, 2024	Aug 14, 2024	6 days	100
18	Dissertation Submission - Milestone	Aug 22, 2024	Aug 24, 2024	0.5 days	100

Table 3: Tasks Segregation

2 Related Work

2.1 Text Sentiment Analysis using ML

Sentiment Analysis is a study of how emotions, responses and attitude are expressed with reference to any object or any subject. For service providers, it might therefore be crucial because it enables them to evaluate their new features and products quickly. Using the OpinionFinder lexicon scoring technique, Brendon O'Connor [7] conducted a sentiment analysis mapping Twitter sentiments with public opinion mood for a time-series study that yields positive and negative orientation of the terms with respective scores.

A paper related to ML techniques in Sentiment Analysis by Bhavita BK and Dr. Niranjan Chiplunkar [8] provides an overview of recent studies on the categorization and analysis of sentiments. They conclude that supervised learning techniques, such as support vector machines and naive Bayesian networks, which are regarded as conventional techniques, the accuracy's of the Support Vector Machine is superior than that of several other classifiers. Regarding accuracy, they found that SVM is the optimal option for large

feature sets, whereas Naive Bayes performs well with small feature sets. Because lexical-based techniques necessitate manual labor on the document, they are optimally aggressive. Maximum Entropy exhibits superior performance, yet, it is hampered by excessive fitting. Numerous studies have used various opinion mining methodologies, however automated analysis is still required to solve all of the problems associated with sentiment analysis at once.
An article by Waqar Muhammad, Maria Mushtaq and Muhammad Yaseen Khan [9], which focuses on the sentiment analysis of customer reviews on online shopping portals, using ML and lexicon approaches. They conclude that ML approaches showcased better results with 100 % data coverage. However, they need labelled training data to build the model which can be a resource consuming and expensive process. But they show thay even if the labelled data is not available, then supervised learning techniques such as Support Vector Machines (SVM) which learnt from other sources can be efficiently used to classify product reviews for the platforms such as Facebook datasets. This was concluded after creating 5 different ML models and 3 different lexicon techniques and finally comparing their results.

2.2 Text Sentiment Analysis using BERT

BERT, which was trained on a sizable English Wikipedia corpus, is the first pre-trained bidirectional and fully unsupervised language representation technique in history. It is a Google AI-developed Open-Source Language Representation Model. BERT is better than a single-direction method since it can read texts (or a set of words) in either way prior to training [2]. The essential element of BERT is the transformer's attention mechanism. When a phrase in a sentence is often connected to its surroundings, the attention mechanism aids in deriving its semantic meaning. A word's semantic representation is strengthened by the context it is in. Concurrently, additional terms in the context usually have more than one function in enhancing semantic representation. By analyzing contextual data, an attention mechanism can improve the target sentence's semantic representation. BERT uses two different strategies: next sentence prediction (NSP) and disguised language modeling (MLM), in contrast to earlier word embedding systems [10]. An article by Nusrat Jahan Prottasha, Abdullah As Sami and Md Kowsher compares the ML and Deep Learning algorithms for text classification as per their topics. The demonstrate that LLM transformer model such as *BERT* with precise fine-tuning can play a vital role in sentimental analysis. They use a specific BERT model called *Bangla-BERT* which is a pre-trained language model Bengali language using mask language modeling [11] along with the combination of LSTM, gave them the most significant results with 94.15% accuracy. Another article by B. Selvakumar and B. Lakshmanan in which they have proposed work for classifying users sentiments on IMDB movie reviews. Their work showcased how BERT based sentiment classification outperformed other machine learning (ML) and deep learning (DL) models and BERT was able to classify much more precisely [12] as displayed in figure 1 below.

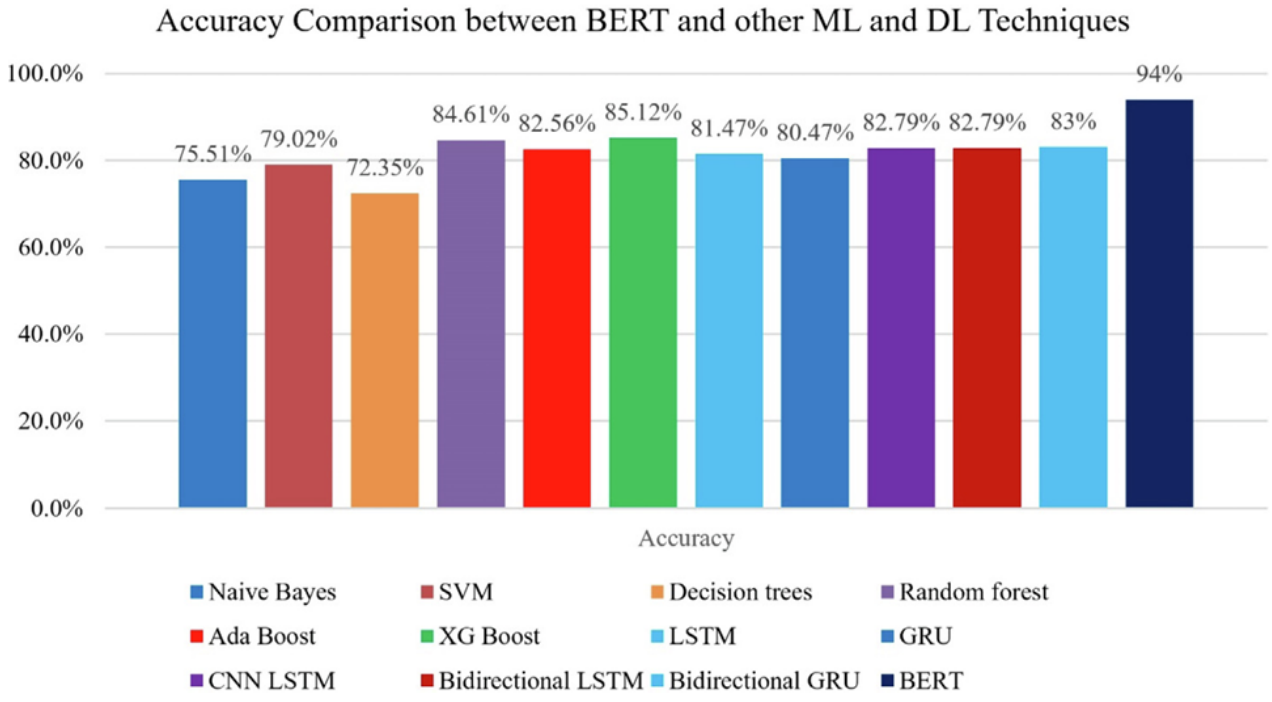

Figure 1: Comparison of accuracy of BERT with other ML and DL techniques

As displayed in the figure above, the accuracy is 94% for BERT model which is much higher comparatively to all the other models used.
Another technical blog by Yash Inaniya [13], in which he demonstrates performing sentimental analysis using BERT on *Amazon* products customer reviews. He blog concludes that BERT model performed fairly well comparatively to Recurrent Neural Network (RNN). However, the only constraint he explained is related to quite high computational power and a fairly huge amount to train the data. But he also says that the computational time is larger due to the complexity of the data. If the data is not complex then simpler models can be used and they might perform much faster and might produce fairly efficient results.

2.3 Llama Models in Product Review Evaluation

As per article [14], understanding and improving consumer happiness through the use of innovative technology, especially Large Language Models (LLMs) such as *llama-2-70b-chat, gpt-3.5-turbo-1106* etc, is turning e-commerce operations into a whole new ball game. According to a recent study, LLMs have proven to be highly effective at interpreting customer sentiments and purchase decisions, which is critical for e-commerce enterprises to remain successful over time in the face of intense market rivalry. The study looked at using LLMs to measure reviewers' happiness after their buying experiences, and the findings were very encouraging. Surprisingly, LLMs in both their baseline and optimized variants had predicted accuracy rates close to 65% when evaluating review ratings. This high degree of accuracy demonstrates how well the models comprehend the complex context and language of client feedback. The alignment between the lexical elements detected by human assessors and those identified as relevant by LLMs was a very interesting conclusion. This suggests that LLMs have an advanced capacity to understand the significance of individual words in the larger context of reviews, simulating human understanding. For e-commerce businesses, these LLM skills are priceless since they offer deeper insights into the opinions and preferences of their customers. By using these information, organizations may make strategic decisions

that will eventually improve the experience and happiness of their customers. Using the power of LLMs, e-commerce entrepreneurs can gain a deeper understanding of the elements that influence customer satisfaction and dissatisfaction. This understanding helps them handle issues more skillfully and accurately customize their services to match the needs of their customers. The study's findings highlight the revolutionary potential of LLMs in e-commerce, as they offer a strong instrument for strategic decision-making and the improvement of customer happiness in addition to increasing review rating prediction accuracy. This innovative technology gives e-commerce companies a competitive advantage by creating a setting where consumer insights result in better customer service and greater customer loyalty.

2.4 Summary

The related work on text sentiment analysis in e-commerce highlights the significant advancements and applications of various machine learning (ML) and language model techniques. Sentiment analysis is crucial for service providers as it helps evaluate customer feedback on products and services. Traditional ML techniques like support vector machines (SVM) and naive Bayes have been widely used for sentiment classification. SVM is found to be optimal for large feature sets, while naive Bayes performs well with smaller sets. Despite their effectiveness, these techniques often require extensive manual effort and labeled data, which can be resource-intensive. Recent studies have shown that the integration of advanced models like BERT (Bidirectional Encoder Representations from Transformers) offers substantial improvements in sentiment analysis. BERT, developed by Google AI, is a pre-trained language model capable of understanding context bidirectionally. It leverages the transformer's attention mechanism to derive semantic meaning from context, enhancing the accuracy of sentiment classification. Studies have demonstrated that fine-tuned BERT models, such as Bangla-BERT for the Bengali language, achieve high accuracy rates, outperforming traditional ML and deep learning (DL) models. For instance, BERT-based sentiment classification on IMDB movie reviews showed superior performance compared to other models. Moreover, the emergence of Large Language Models (LLMs) like llama-2-70b-chat and GPT-3.5-turbo-1106 has further revolutionized sentiment analysis in e-commerce. These models have proven highly effective in interpreting customer sentiments and purchase decisions, achieving predictive accuracy rates close to 65% for review ratings. A key finding is the alignment between the lexical elements identified by LLMs and those recognized by human evaluators, indicating LLMs' advanced ability to understand the significance of words in context. This capability provides e-commerce businesses with deeper insights into customer opinions, enabling strategic decision-making to enhance customer satisfaction and loyalty. Overall, the integration of BERT and LLMs into sentiment analysis offers a promising approach for e-commerce enterprises. These models not only improve the accuracy of sentiment classification but also provide valuable insights for strategic decision-making, ultimately leading to better customer service and increased satisfaction. This technological advancement gives e-commerce businesses a competitive edge by leveraging customer feedback to drive improvements and foster loyalty.

3 Preparation & Design

3.1 Analysis of the method

Listed below are some main python libraries that are used to achieve the objectives listed in 1.2.

3.1.1 Data Extraction

- Requests: This module used to extract data via HTTP/HTTPS requests by using the private API Token / Key along with Wex Business ID.
- Selenium or BeautifulSoup: These libraries were originally supposed to be used to scrape data from the review websites in case the API token / key with Business ID is not provided by Wex due to *GDPA* constraints. However, fortunately the API key was provided by the Wex engineering team and these libraries were never used but remained as a backup in case the key expires.

***Trust Pilot* extracted data via private API token using Requests module along with other libraries such as json, csv, os etc.**
Title: Excellent company.
Content: Excellent company to deal with,ordered my item and was shocked how quickly it came considering postage was free.Wouldnt hesitate to use again.
Date of Experience: 2024-04-28T00:00:00Z
Review Date: 2024-04-29T19:27:23Z
Rating: 5
Reviewer Name: Christian Hindle
Location:
No. of reviews given: 7

***Power Reviews* extracted data via private API key using Requests module along with other libraries such as json, csv, os etc.**
Created Date: Wed May 1 22:01:43 2024
Page ID: 614293
Review Rating: 4
Review Headline: Slow to accept payment.
Comment: Slow to accept payment. Had to process twice. Then having selected named day delivery ended up with standard delivery. Not what I normally expect from Wex.
Review Location: GB
Reviewer Nickname: aar6478

3.1.2 Supervised ML Model

- Spacy & NLTK: These libraries are used for text lemmatization, stop words removal etc.
- Contractions: This module is used to remove contractions, emojis, punctuations, currencies, numbers in the review.
- CountVectorizer: This library is used during the pre-processing phase to convert text to numerical presentation.
- Matplotlib, plotly & Seaborn: These libraries are used to plot bargraphs, histograms, plots etc for better visualizations.

- Pandas & Numpy: These modules are used for data storage inside dataframes along with dataframe manipulation tasks.
- Train_test_split: This python library is used to split the data into training, validation and test sets.
- SVC, LogisticRegression, KNeighborsClassifier, xgb, RandomForestClassifier: These are some supervised and emsemble model python libraries to predict sentiment classes on the train and test data.
- Confusion_matrix, accuracy_score, f1_score, recall_score: These libraries are used to display how well the model has performed after training.

3.1.3 BERT Transformer

- Transformers: This is library maintained by Hugging Face community, for state-of-the-art Machine Learning for Pytorch and TensorFlow [15].
- BertModel, BertTokenizer, DistilBertTokenizer: These are vital BERT transformers library's in-built classes.
- Torch or Tensorflow: These libraries are used for building DL neural network algorithms.
- AutoTokenizer: This is one of the transformers library's in-built classes that converts text into array of numbers.
- DataCollatorWithPadding: This is one of the transformers library's in-built classes that helps in preparing batches of data for training transformer models.

3.1.4 Issue Type Classification

- spacy PhraseMatcher: This library is used match pre-defined / hard-coded phrases related to the possible issues found in the customer reviews.

3.1.5 Reason behind issue text-generation

- ollama: This library is used to extract llama3 8B parameter pre-trained model to generate accurate and appropriate text notifying the issue if the issues detected by the system are actually genuine or not by analysing the.
- Multi-threading: concurrent.futures library is used to run Ollama in parallel using virtual core of the local machine to speed up the process of generating reason behind issue and saving it into a read-able csv.

3.1.6 Deploy Best Models on Web Application

- Dash or streamlit: On-demand web application is created using dash library that comes within *Python's Flask* framework that can be shared with *Wex Photo Video* as an end product.
- Web application using streamlit python library was originally developed as well but it did not work as expected on Windows enviroment but only worked well on Linux OS. Hence, it was deprecated later.

3.2 Tools and Resources

The primary programming languages used for this project is *Python*. Extracted data from review websites is stored in local csv files for manual access purposes.

Programming Languages: Python.

Webapp Development: Dash (Flask).

Packages: **For NLP**: NLTK, Spacy, PhraseMatcher. **For Graphical Visualizations**: Matplotlib, plotly and seaborn. **For ML Models**: SVC, LogisticRegression, KNeighborsClassifier, xgb, RandomForestClassifier etc. **For Evaluation Metrics**: Confusion_matrix, accuracy_score, f1_score, recall_score. **For DL LLM**: Transformers, torch, BertModel, BertTokenizer, from transformers import DistilBertModel, DistilBertTokenizer, DistilBertForSequenceClassification, DistilBertConfig, Tensorflow, Ollama etc. **For Webapp**: dash, flask.

3.3 Design of Methodology

3.3.1 MoSCoW

A crisp summary of objectives also explained in the M(Must haves)oS(Should haves)C(Could haves)oW(Won't haves) in the diagram below:-

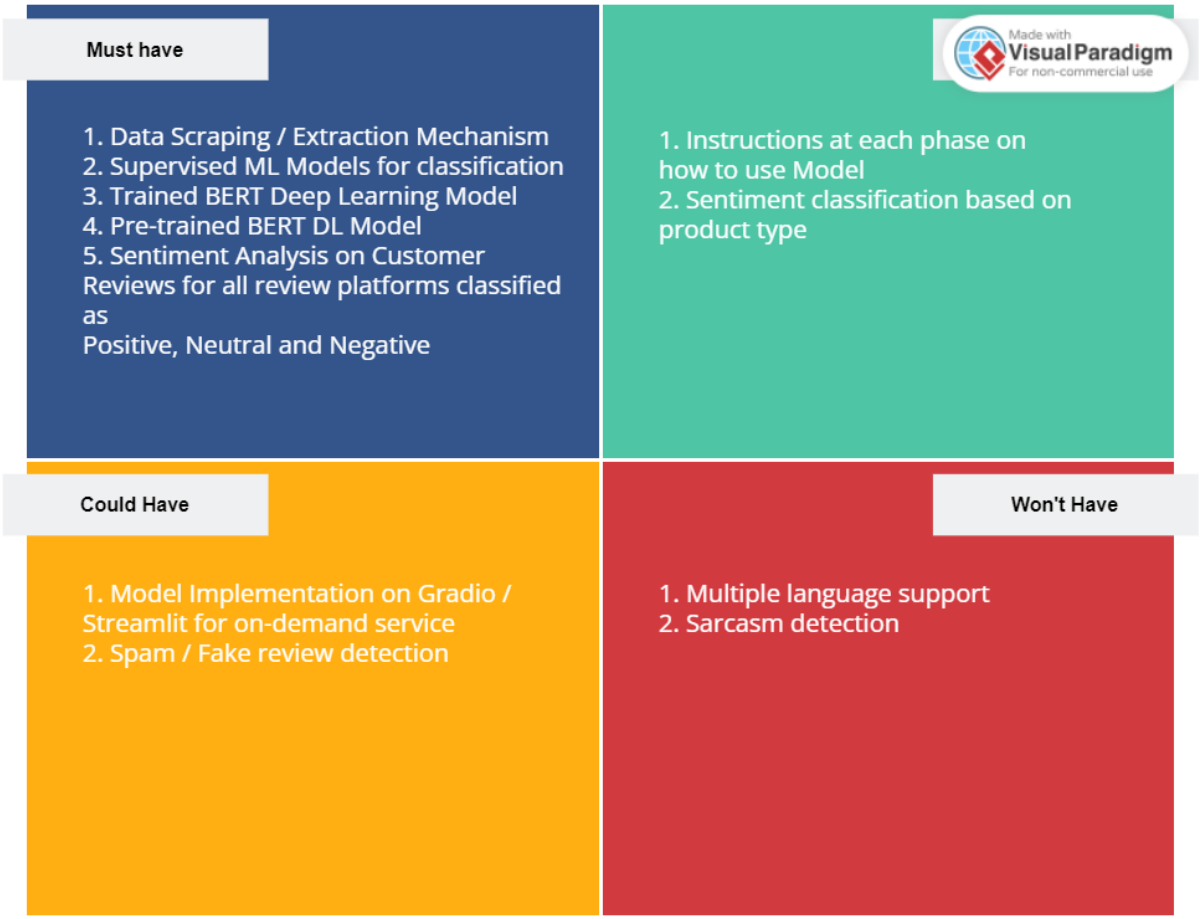

Figure 2: MoSCoW - Sentiment Analysis for Wex

3.3.2 Stakeholders

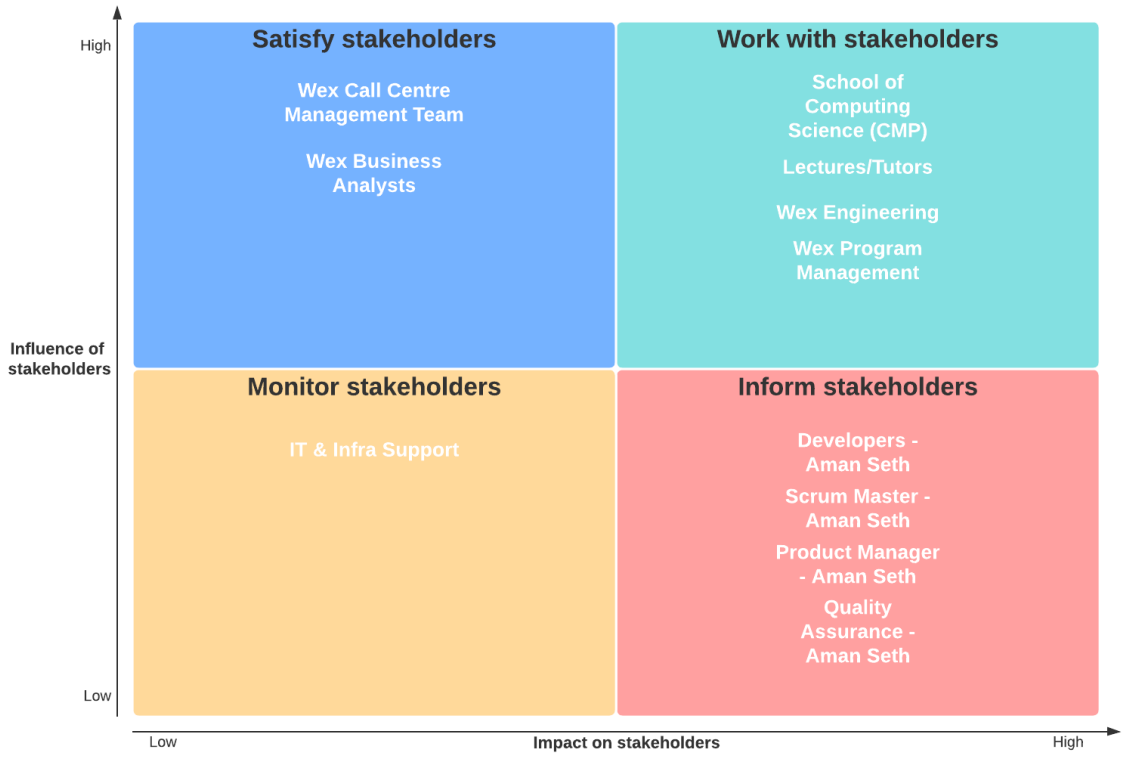

Figure 3: Relevant Project Stakeholders

Satisfy stakeholders: Within the framework of Wex Photo Video, the Wex Business Analyst and the Wex Call Center Management Team are the primary stakeholders to be satisfied. To guarantee their contentment, it is imperative to offer lucid, practicable deductions from the sentiment research to augment the efficacy of client service and guide corporate choices, therefore elevating overall operational efficiency and strategic planning.
Work with stakeholders: In order to guarantee technological viability, efficient project management, and academic rigor for this project, close collaboration with stakeholders including the Wex Engineering Team, Wex Program Management Team, and UEA CMP Department is required. In order to achieve successful project outcomes, this collaboration helps to align project goals, expedite execution, and harness expert knowledge.
Monitor Stakeholders: Keeping a careful eye on the actions and output of stakeholders, including the UEA IT and Infra assistance Team, is necessary to guarantee dependable infrastructure and technical assistance. Proactive monitoring ensures that the project's technological needs are satisfied, that technical concerns are immediately addressed, and that project operations remain smooth.
Inform Stakeholders: Providing regular updates on project progress, modifications, and significant milestones is part of informing stakeholders, including developers, the scrum master, product owners, and quality assurance engineers. Throughout the project lifecycle, effective communication helps maintain high standards of quality, promotes agile development, and guarantees alignment among all parties involved.

3.3.3 High-level Design Flowcharts

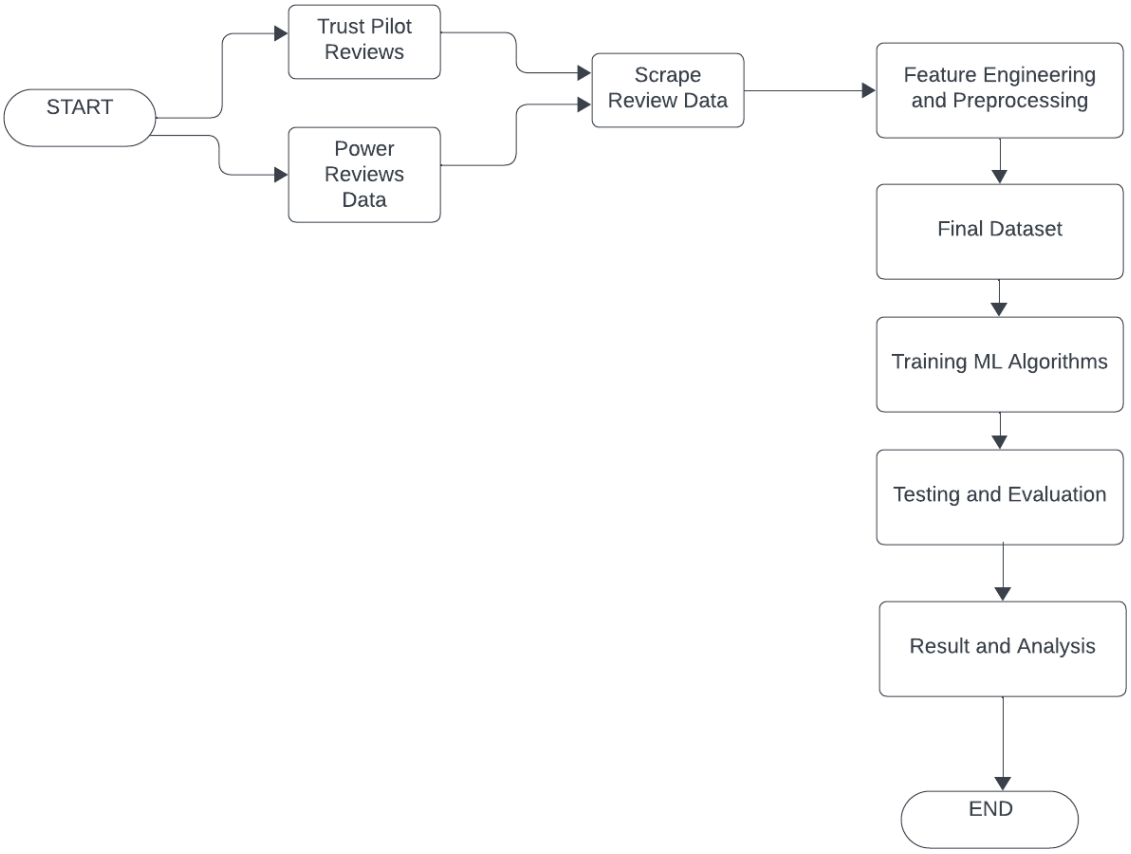

Figure 4: Supervised Learning using ML

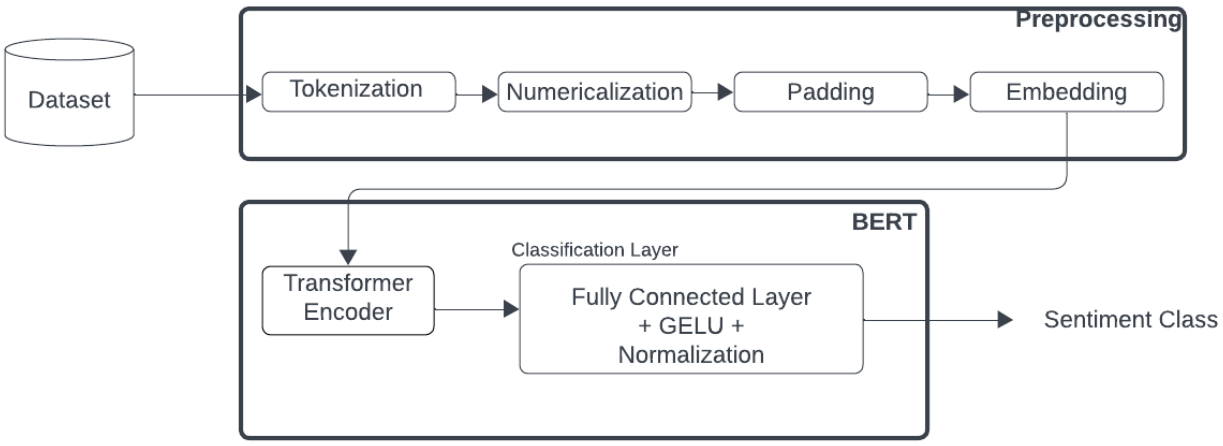

Figure 5: BERT Model Flowchart [12]

4 Implementation

4.1 Customer Reviews Extraction

As described in section 3.1.1, below are the vital fields which were extracted and played a crucial role in train the Machine Learning and Transformer models:-

- Content
- Rating

The other fields are either dropped or not considered during models training. The content column is the actual review from the review platforms *Trustpilot* and *Power Reviews*. The rating field lies within the range 1 -5. Reviews with ratings 1-2 are considered to be poor reviews or the reviews where the customers have expressed their dissatisfaction and disappointment with Wex Photo Video services or products. The reviews with rating 3 are considered to be neutral reviews where the customers neither are too happy and contented nor completely disappointed. The ratings between 4-5 are positive reviews where customers are extremely happy and satisfied with Wex's services and their products.
Below are sample customer reviews from both platforms:-

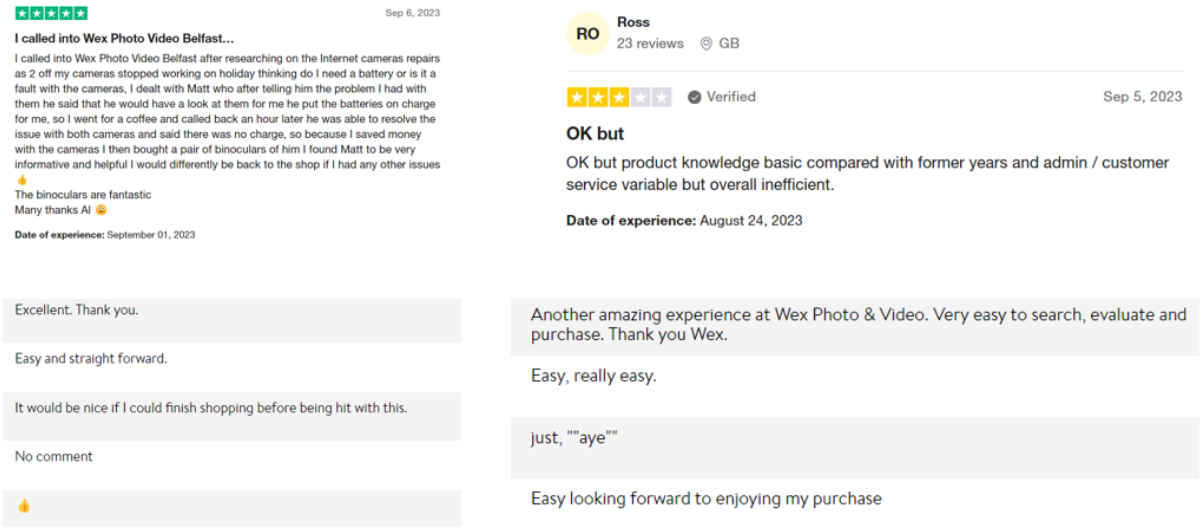

Figure 6: Sample Customer Reviews

4.2 Data Pre-processing

This section describes the different pre-processing techniques which were applied in order to clean and sanitize the data suitable for model training to achieve better model accuracies, scores and final results.

Total extracted review platforms customer reviews: 23784 (Since Jan 1 2015)

4.2.1 Feature Engineering

The reviews distributed amongst rating 1-5 were classified into a new column named *Sentiments*. Below is the classification of sentiments as per ratings:-

- Sentiment Label 0: Rating 1-2 (Negative)
- Sentiment Label 1: Rating 3 (Neutral)
- Sentiment Label 2: Rating 4-5 (Positive)

4.2.2 Handling Missing Values & Data Balancing

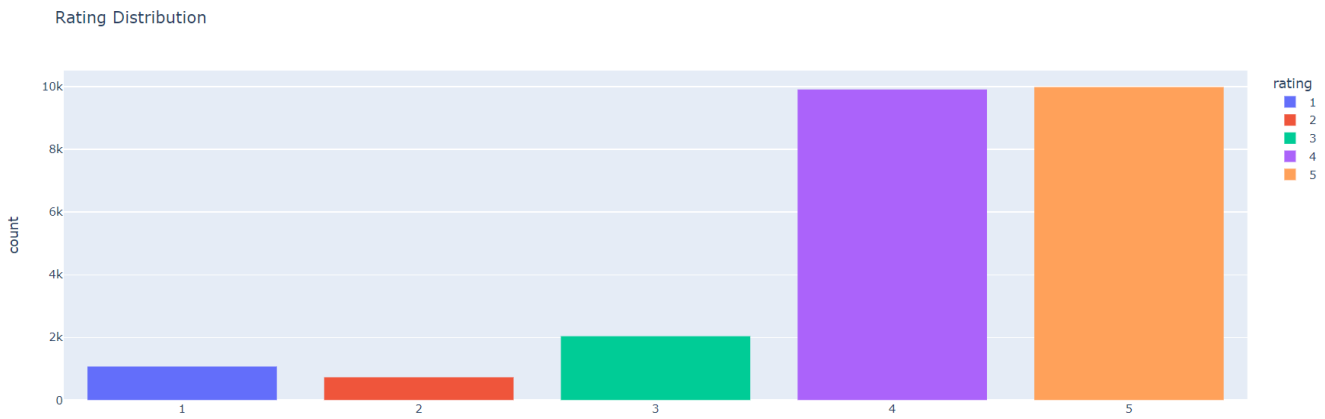

Figure 7: Customer Reviews per ratings

Approach 1 : TF-IDF (Term Frequency - Inverse Document Frequency)
As observed in figure 6 above, it is vividly visible that reviews with ratings 4 and 5 are clearly dominating the other ratings reviews by a huge margin of about 90+%. *Undersampling* these reviews by dropping them would not have been an appropriate approach since majority of the data to train would have been lost leading to un-expected results later. Hence, the first approach tried was to *randomly oversample* the minority classes i.e. randomly impute the reviews between rating 1-3 to level out the reviews. Below is the sentiment distribution after applying TF-IDF oversampling:-

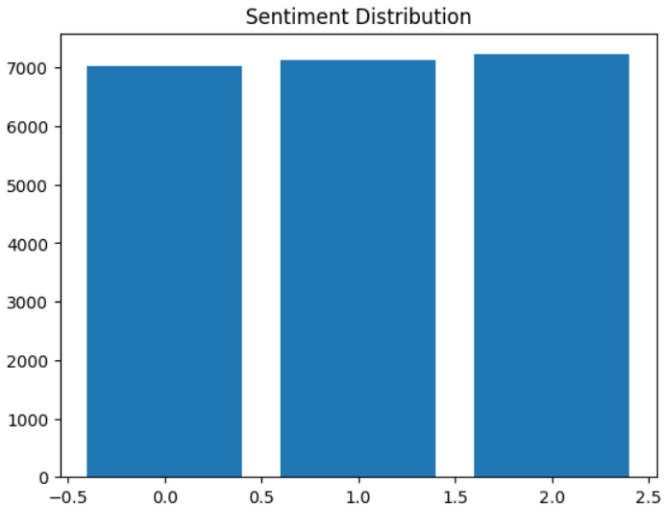

Figure 8: Balanced Dataset via TF-IDF

The 3 sentiment labels (positive, neutral, negative) are equally distributed to about ˜7000 reviews each.

Approach 2 (Final): Oversampling using external data
The oversampling performed via TF-IDF approach resulted in poor model scores, confusion matrix and overall result due to highly **redundant** data. Hence, to mitigate this problem, it was decided to integrate the real and actual neutral and negative (between ratings 1 - 3) customer reviews from external sources. Two sources were found on *Kaggle* [16] and [17] from where, e-commerce customer reviews were extracted and integrated amongst sentiment label 2 (rating 4-5), wex customer reviews to balance out the data. Below is the balanced dataset based on sentiments distribution:-

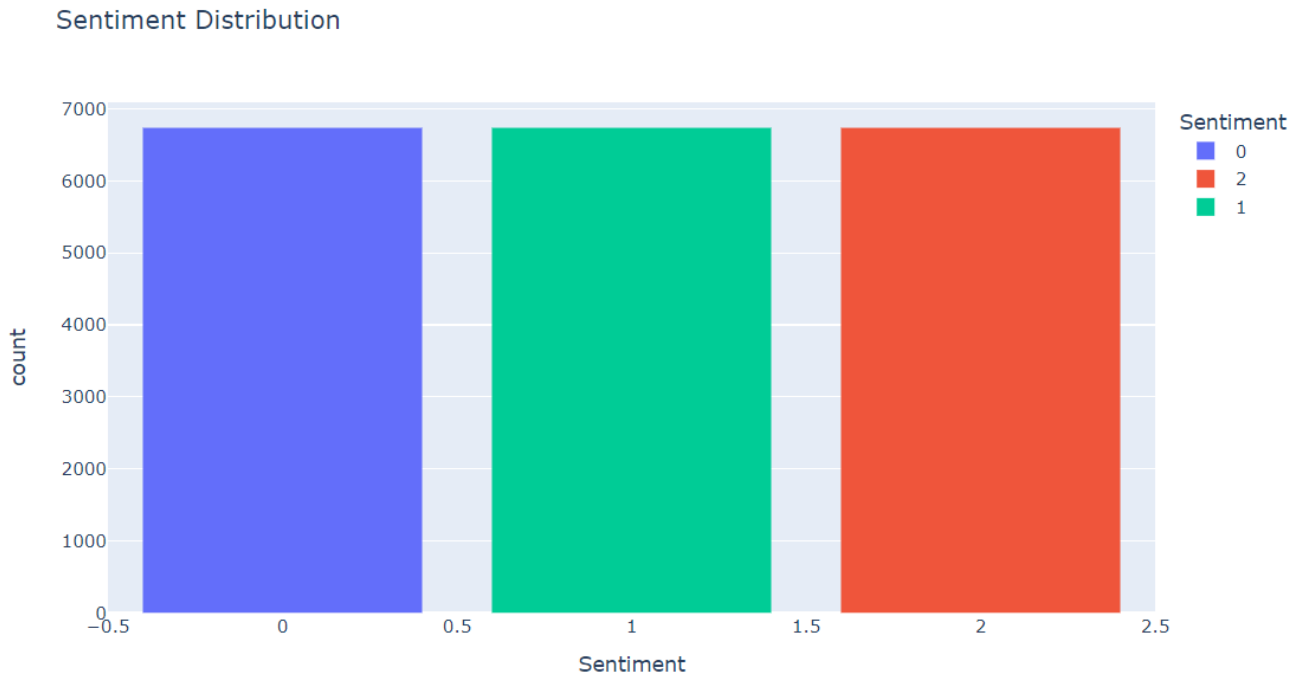

Figure 9: Sentiment Distribution

The model accuracies and other results achieved on this balanced dataset were far more better and productive which is explained in detail and discussed in later sections 5 and 6.

4.2.3 Emoji & Contractions Handling

Approach 1 : Emoji Removal
It was observed that many customer reviews on review platforms did not contain text / words but customers sentiments were expressed only in the form on different emojis or the reviews sentenced contained single or multiple emoticons. The first adapted approach was to remove and drop such comments completely. However, this approach resulted in un-expected and poor end results by the models discussed in later section under evaluation and discussion 6.

Approach 2 (Final) : Emoji Replacement
To get better results produced by the ML / DL models, it was decided not to remove the emojis but replace them with meaningful texts as per the type of the emoticon. Below are some examples of emojis and their respective replaced texts:-

Emoji	Replacement Text
:)	Happy
:(	Sad
:/	Confused
:X	Angry

Table 4: Emoji Replacement

Contractions Removal : The contractions were removed. Below is the example of some reviews before and after removing contractions:-

Before Contractions Removal	After Contractions Removal
I've purchased from wex before and always love their service	**I have** purchased from wex before and always love their service
Had a bad experience ordering Sony MDRX 1000 camera and **it's** the first time I **must've** had such an experience	Had a bad experience ordering Sony MDRX 1000 camera and **it is** the first time I **must have** had such an experience

Table 5: Before vs After Contractions Removal

4.2.4 Special Characters & Other Symbols Handling

There were many reviews which also had many other special characters, symbols, chinese symbols, flags, chinese or other language characters. They all were removed as well. It was kept in mind not to drop the whole word but only remove these characters. Some examples of special characters removed are:-

Special Character	Meaning
!,."	Punctuation Marks
\[ ]	Square Brackets
{ }	Curly Brackets
0-9, $	Currency Symbols and Numbers

Table 6: Special Characters Removal

4.2.5 STOP words & Lemmatization

Approach 1: STOP words removal and lemmatization
Initially, it was planned to drop the stop words and lemmatize the reviews as much possible so that the reviews could be sanitized enough before passing them to model training. However, the model did not produce expected results and was biased as explained under evaluation section. Surprisingly, the whole meaning of the sentence changed post applying lemmatization and STOP word removal techniques. Some examples are displayed below of pre and post lemmatized review:-

Before Lemmatization	After Lemmatization
Un-clear website. Not easy to use	clear website easy to use
not so easy to use online system	easy to use online system

Table 7: Before vs After Lemmatization

As seen above, the whole meaning / sentiment behind the customer reviews changed from negative to positive.

Approach 2 (Final) : Exclusion of lemmatization
Since the whole sentiment was being changed after review lemmatization, hence it was decided not lemmatize the reviews at all and not removing STOP words.

4.2.6 Duplicate Reviews Handling

To remove the redundant data, the reviews which were exactly the same were removed completely. It was taken under consideration the fact that even if the review is exactly the same but the rating given by the customer is different, then not to remove it. Only the cases where ratings and review content is exactly the same were removed. Below are few examples:-

Rating	**Review**
3	good
5	good
3	ok
3	ok

Table 8: Before Duplicate Removal

Rating	**Review**
3	good
5	good
3	ok

Table 9: After Duplicate Removal

4.2.7 Remove Meaningless Reviews with Length <=2 & <=3

The intention here to find the reviews with length less than and equal to 2 and 3 characters and remove the ones which do not make any sense and therefore might not contribute in the model training. Below are some examples of such reviews:-

Content	**Rating**
??	3
!!	3
:[	1
mmm	2

Table 10: Meaningless Reviews

The reviews which had emoji like reviews that is :[, those are replaced with their respective meaningful text. In this case :[was replaced by text *sad*.

4.2.8 Remove More Meaningless Reviews

After removing short meaningless reviews of length less or equivalent to 3, it was observed that there were many reviews left with greater lengths with no meaning. The reviews which had same repetitive characters were excluded as well. Some examples of such reviews are presented in table below;-

Content	Rating
aaaaaaaaaaaaaaaaaa	4
ooooo	2
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz	1
xxxxxxxyyyyyyyy	2

Table 11: More Meaningless Reviews

4.2.9 Summary

Total Reviews Before Pre-processing	Total Reviews After Pre-processing
23,784 (Since Jan 1 2015)	20,748 (Since Jan 1 2015)

Table 12: Total Reviews After Pre-processing

4.3 Machine Learning Models

4.3.1 ML Low Level Design Flowchart

Figure 10 demonstrates the low level design flowchart of the implemented Machine Learning models.

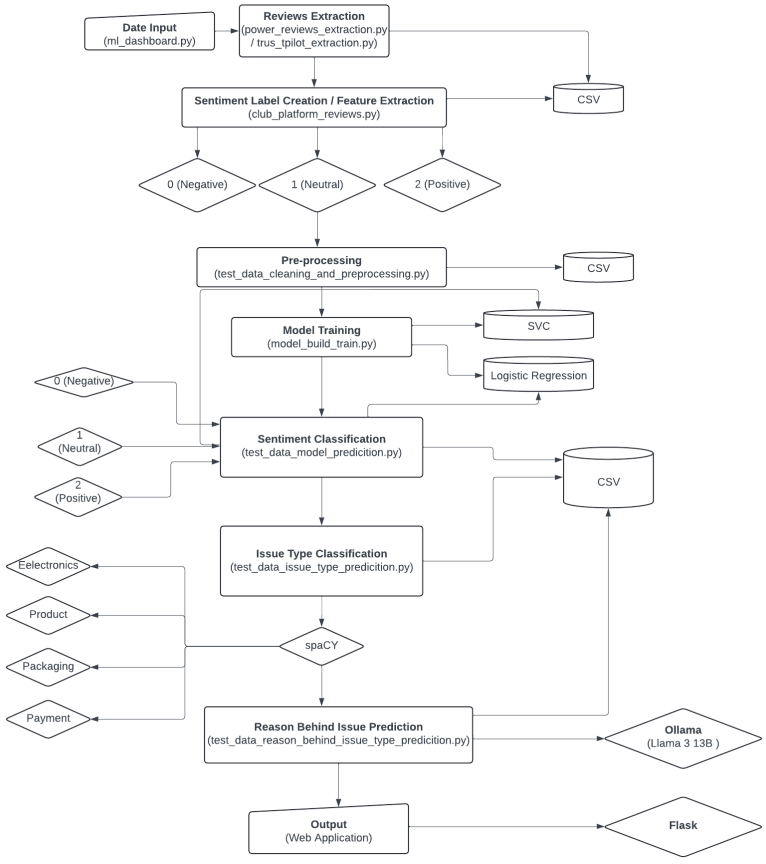

Figure 10: ML Low Level Flowchart

4.3.2 ML Use-case Diagram

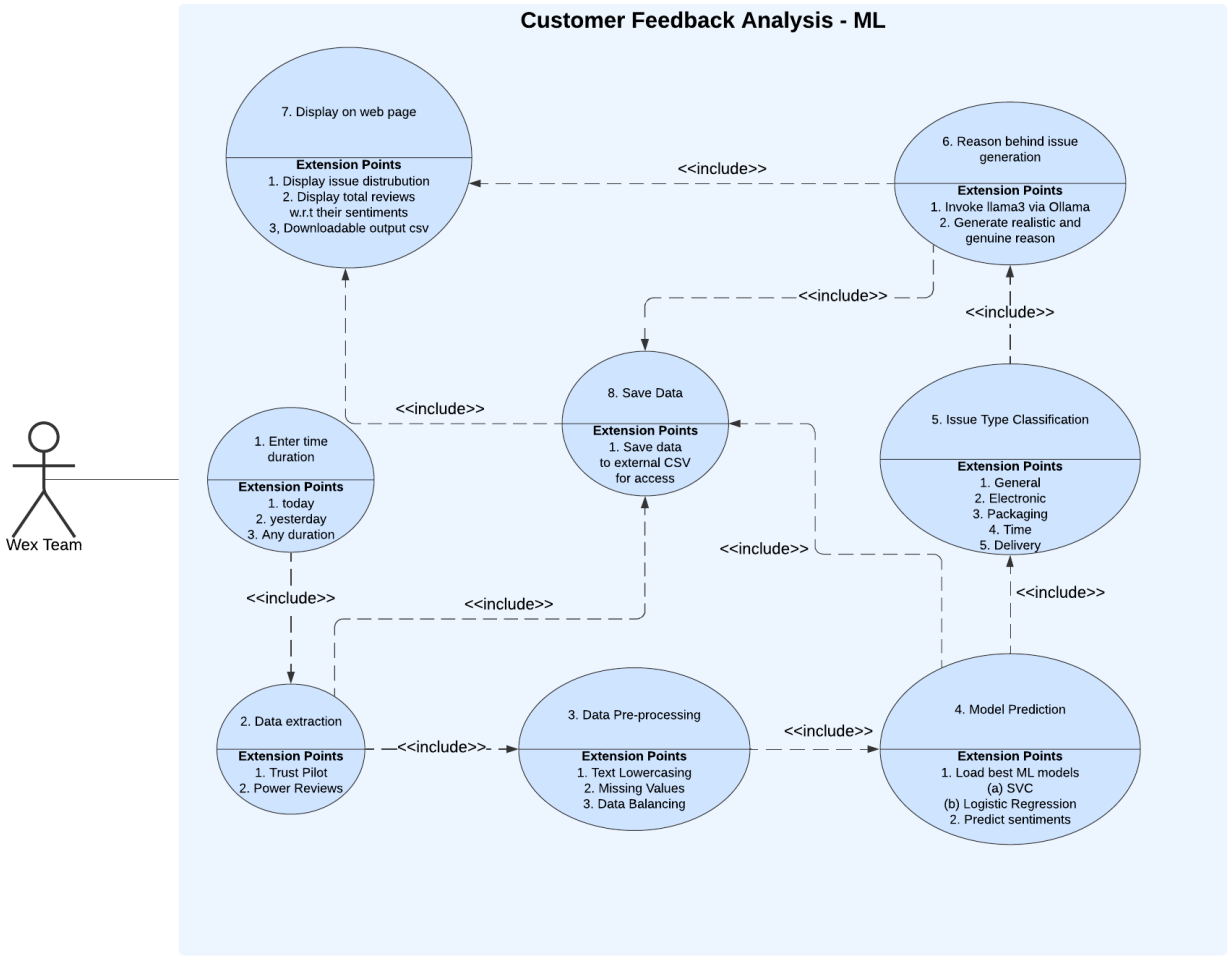

Figure 11: Use Case Diagram - ML Model Webapp

Use Case Diagram Summary:

- Use Case 1: User enters duration which can be from today to last 12 months
- Use Case 2: Reviews data from Trust Pilot and Power Review is extracted and dumped to a csv
- Use Case 3: Data is pre-processed and sanitized in order to be passed to the saved ML models
- Use Case 4: SVC and Logistic Regression (best) models are loaded and sentiment predictions are made as labels 0 (negative), 1 (neutral) or 2 (positive)
- Use Case 5: spaCy phrase matcher is invoked to classify the reviews as per detected issues
- Use Case 6: llama3 model by Ollama is invoked to generate reasons behind the customer reviews that contains issues
- Use Case 7: Display sentiment distribution, issue distribution and downloadable csv with reasons behind issues on a webpage for easy access

4.3.3 ML Sequence Diagram

The sequence diagrams are designed per use case explained in 11.

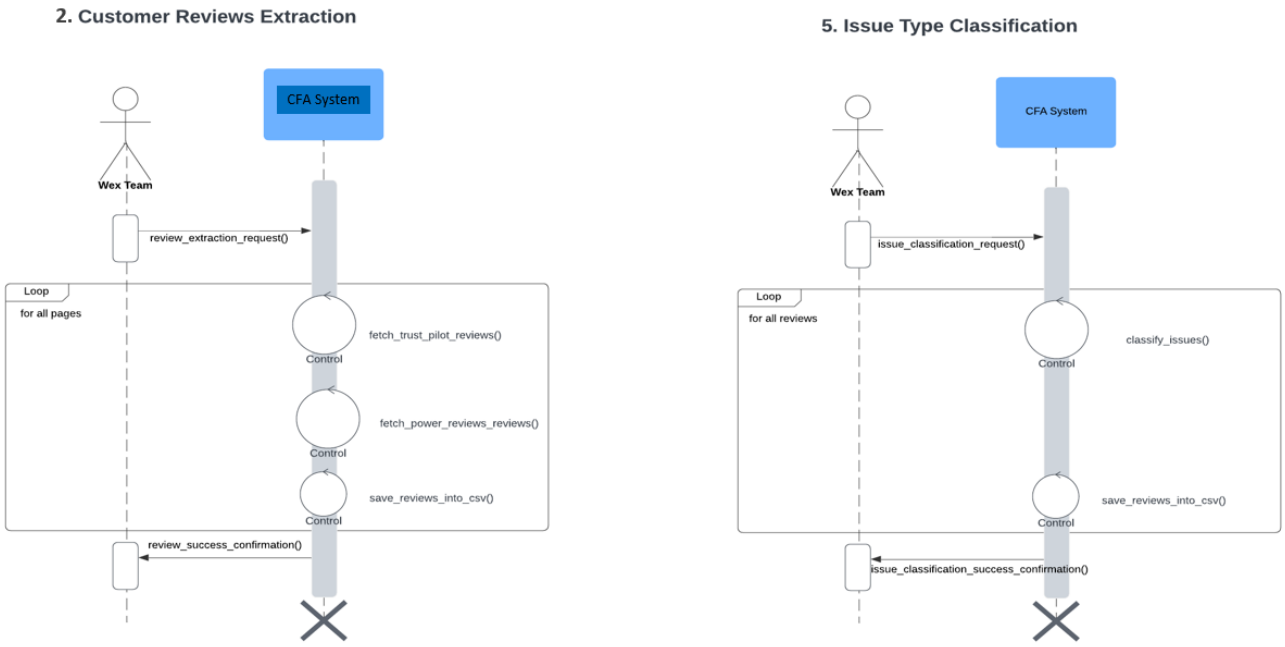

Figure 12: Sequence Diagrams - ML

Figure 12 Sequence Diagrams Summary:

- Case Number 2: Customer Reviews Extraction
  - Wex team user requests for reviews extraction for a given duration
  - System fetches reviews from Trust Pilot
  - System extracts reviews from Power Reviews
  - System stores the reviews into an external CSV for access
- Case Number 5: Issue Type Classification
  - Wex team user requests to classify reviews into issues
  - System invokes spaCy phrase matcher library to classify the predicted sentiments into cateogorical issues
  - The types of issues classified are: General, Electronics, Packaging, Delivery, Time and Customer Service
  - System stores the reviews with classified issues into an external CSV for access

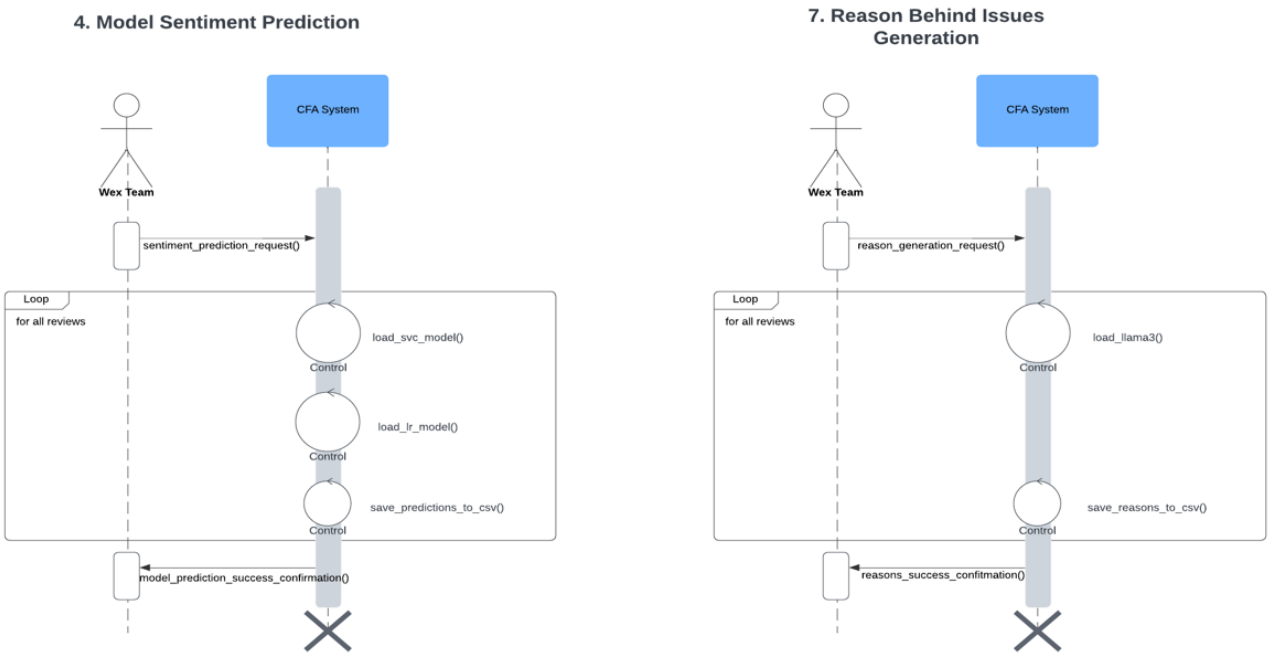

Figure 13: Sequence Diagrams - ML

Figure 13 Sequence Diagrams Summary:

- Case Number 4: Model Sentiment Prediction
  - Wex team user requests for model prediction once reviews are fetched successfully from their respective platforms
  - System loads the best models again: SVC and Logistic Regression
  - System sanitize the input data for it to be acceptable to the loaded models
  - System predicts sentiments in form of labels 0 (negative), 1 (neutral) and 2 (positive)
  - System saves the predicted data into it's respective CSV file for easy access
- Case Number 7: Reason Behind Issues Generation
  - Wex team user requests to generate reasons behind the classified issues for the respective reviews
  - System invokes llama3 8B parameter model through Ollama library
  - System generates the potential and genuine reasons behind the issues
  - System saves the data including predictions, issues and generated reasons for respective customer reviews into a CSV file for access

4.4 BERT Models

4.4.1 BERT Low Level Design Flowchart

Figure 14 demonstrates the low level design flowchart of the implemented BERT Deep Learning models.

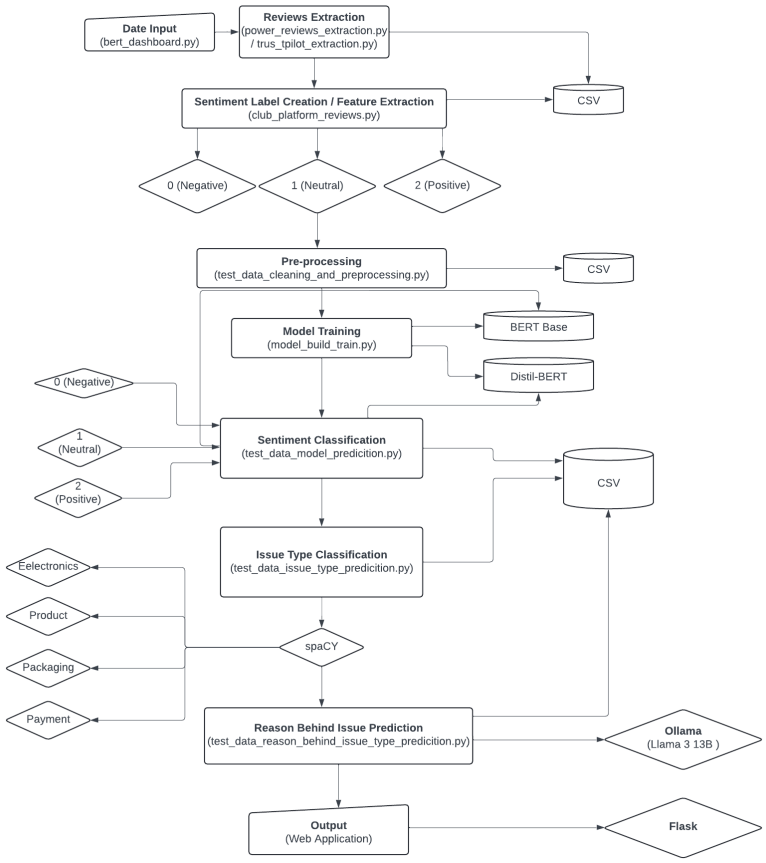

Figure 14: BERT Low Level Flowchart

BERT Flowchart Summary:

- bert_dashboard.py is the central file which invokes rest of the mechanism and displays the results on the webpage hosted locally
- power_reviews_extraction.py and trust_pilot_extraction.py fetches the customer reviews for a given time duration
- Reviews are clubbed and sentiments are created on basis of given rating by club_platform_reviews.py
- Data is pre-processed and trained models are loaded.
- Three DistilBERT models are loaded for label 0, 1 and 2 respectively.
- Model predictions are made and saved into a CSV file.
- Reviews are classified based on different issues: General, Delivery, Packaging, Electronics, Time and Customer Service.
- llama3 8B parameter model is invoked via Ollama library to generate if there is actually an issue behind the reviews which has been classified into an issue category or not

4.4.2 BERT Use-case Diagram

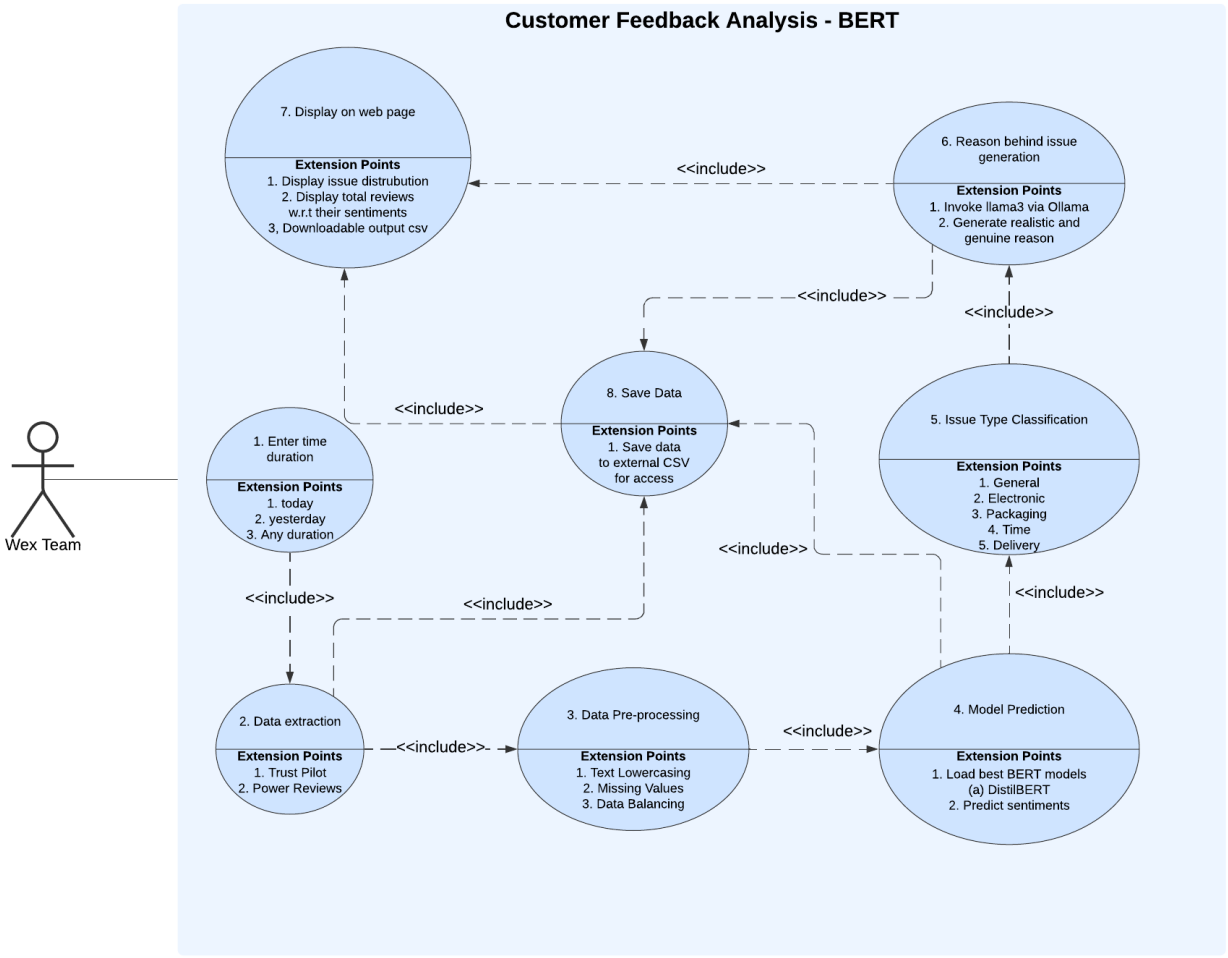

Figure 15: Use Case Diagram - BERT Model Webapp

Use Case Diagram Summary:

- Use Case 1: User enters duration which can be from today to last 12 months
- Use Case 2: Reviews data from Trust Pilot and Power Review is extracted and dumped to a csv
- Use Case 3: Data is pre-processed and sanitized in order to be passed to the saved ML models
- Use Case 4: SVC and Logistic Regression (best) models are loaded and sentiment predictions are made as labels 0 (negative), 1 (neutral) or 2 (positive)
- Use Case 5: spaCy phrase matcher is invoked to classify the reviews as per detected issues
- Use Case 6: llama3 model by Ollama is invoked to generate reasons behind the customer reviews that contains issues
- Use Case 7: Display sentiment distribution, issue distribution and downloadable csv with reasons behind issues on a webpage for easy access

4.4.3 BERT Sequence Diagram

The sequence diagrams are designed per use case explained in 15.

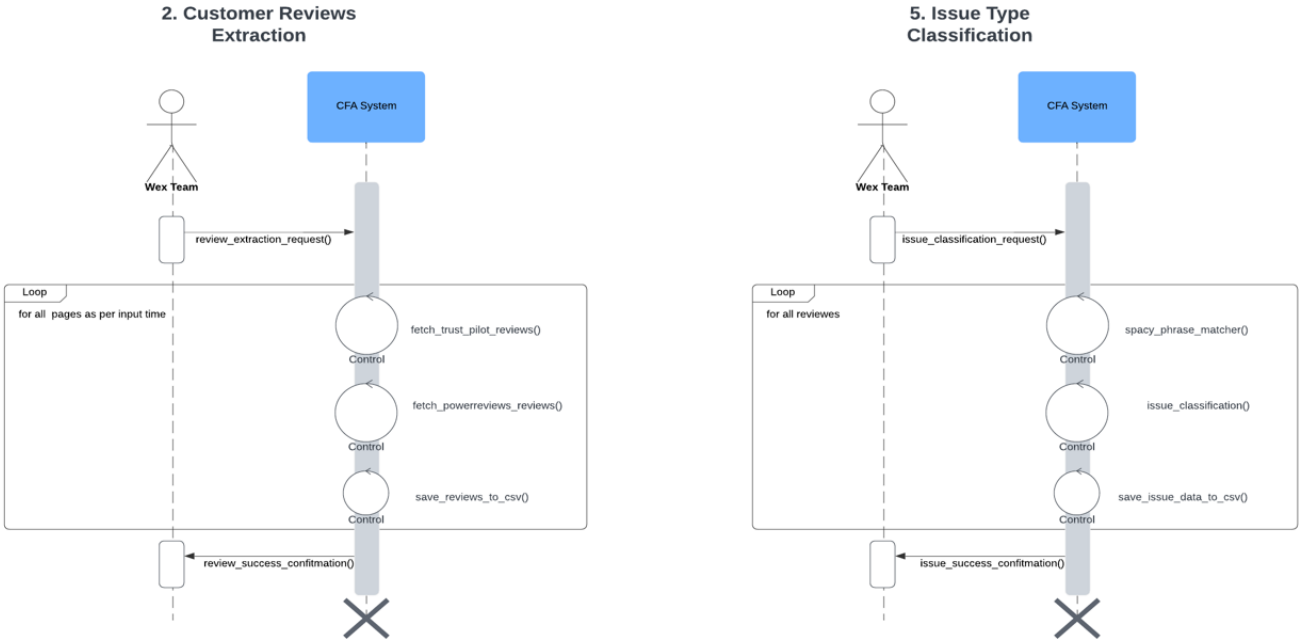

Figure 16: Sequence Diagrams - BERT

Figure 16 Sequence Diagrams Summary:

- Case Number 2: Customer Reviews Extraction
  - Wex team user requests for reviews extraction for a given duration
  - System fetches reviews from Trust Pilot
  - System extracts reviews from Power Reviews
  - System stores the reviews into an external CSV for access
- Case Number 5: Issue Type Classification
  - Wex team user requests to classify reviews into issues
  - System invokes spaCy phrase matcher library to classify the predicted sentiments into cateogorical issues
  - The types of issues classified are: General, Electronics, Packaging, Delivery, Time and Customer Service
  - System stores the reviews with classified issues into an external CSV for access

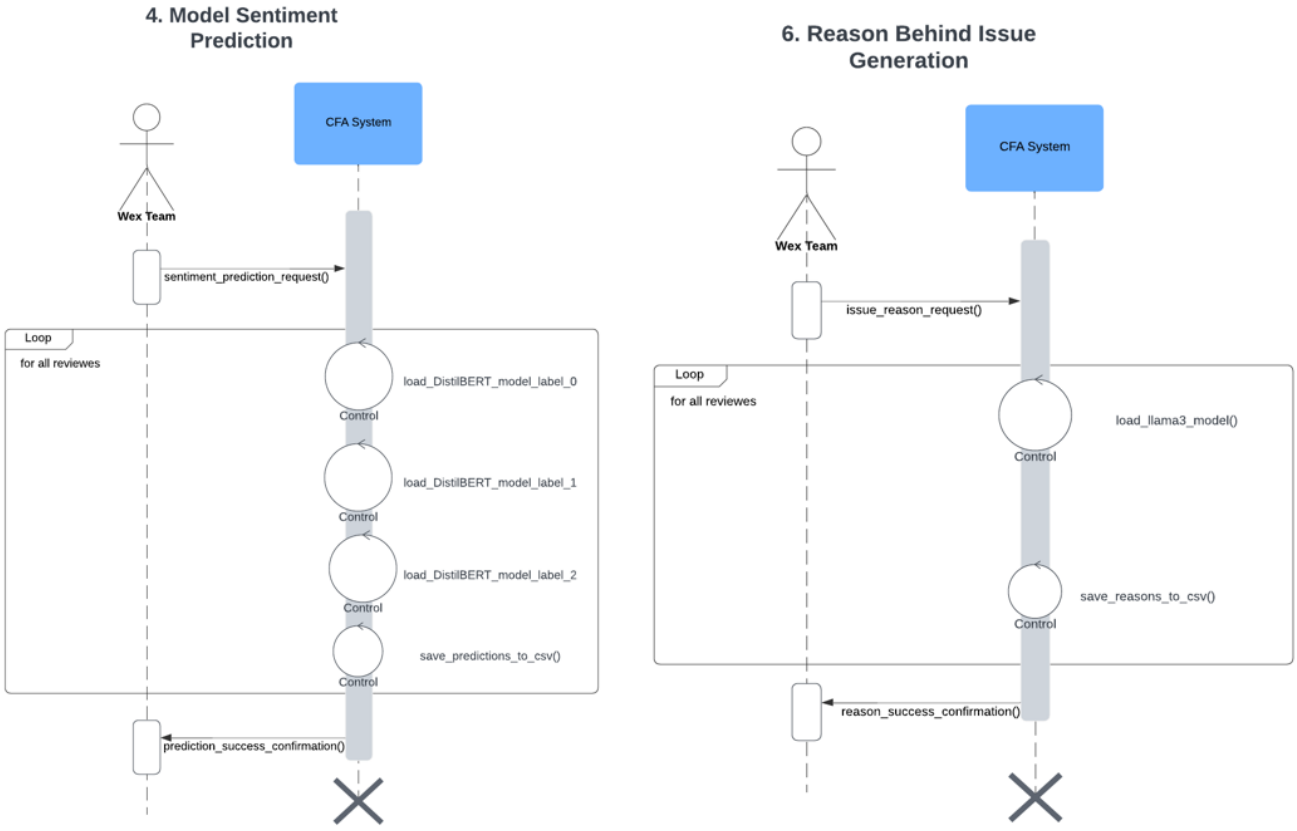

Figure 17: Sequence Diagrams - BERT

Figure 17 Sequence Diagrams Summary:

- Case Number 4: Model Sentiment Prediction
  - Wex team user requests for model prediction once reviews are fetched successfully from their respective platforms
  - System loads the best DistilBERT models as per rating sentiment labels 0, 1 and 2 respectively
  - System sanitize the input data for it to be acceptable to the loaded models
  - System predicts sentiments in form of labels 0 (negative), 1 (neutral) and 2 (positive)
  - System saves the predicted data into it's respective CSV file for easy access
- Case Number 7: Reason Behind Issues Generation
  - Wex team user requests to generate reasons behind the classified issues for the respective reviews
  - System invokes llama3 8B parameter model through Ollama library
  - System generates the potential and genuine reasons behind the issues
  - System saves the data including predictions, issues and generated reasons for respective customer reviews into a CSV file for access

4.5 Web Application Deployment

Figure 18: Web Application - Runs by BERT models

Web Application Summary:

- Figure 1: Input time is required followed by 'Generate Predictions' button click
- Figure 2: Predictions generated. Actual vs Predicted for 3 sentiments (Negative, Neutral and Positive) histogram
- Figure 3: Distribution of sentiments for both Trustpilot and Power Reviews
- Figure 4: Issue type and count histogram for Trustpilot
- Figure 5: Issue type and count histogram for Power Reviews
- Button: *'Download CSV'* button to download CSV file containing predictions, issues and reason behind issues generated text

5 Testing

5.1 Train, Test and Validation Testing

Purpose: To keep an eye on and improve the model's performance as it is being developed. A distinct validation dataset is used to periodically assess the model after it has been trained on a training dataset [18].

5.1.1 ML Models

Approach 1: When data pre-processing was done using TF-IDF random oversampling approach as explained in section 4.2.2

Total Reviews: 21,378

Rating	Total Reviews
1	3607
2	3413
3	7131
4	3628
5	3599

Table 13: Review Distribution / Rating

Train / Test Split: 80/20

Train Data Distribution

Sentiment Label	No. of Reviews
0	5616
1	5705
2	5781

Table 14: Review Distribution / Rating

Model	Accuracy	F1 Score
Logistic Regression	69.8	69.6
Random Forest	68.6	68.3
KNeighbour	47.3	43
SVC	70.9	70.8
Multinomial NB	67	66.3

Table 15: Models Performance on Train Data

Post Hyper-pamater Tuning Best 3 models were chosen for hyper-parameter tuning:-

- SVC
- Logistic Regression
- Random Forest

Model	Accuracy	F1 Score
Logistic Regression	69.8	69.6
Random Forest	68.2	67.8
SVC	69.8	69.8

Table 16: Models Performance post Hyper-param Tuning

Observation: After hyper-parameter tuning on the best 3 ML models, the accuracies degraded by a marginal approx. 0.5 %.

Test Data Distribution

Sentiment Label	No. of Reviews
0	1404
1	1426
2	1446

Table 17: Review Distribution / Rating

Model	Accuracy	F1 Score
Logistic Regression	68.9	68.2
Random Forest	62.6	62.3
KNeighbour	47.3	43
SVC	68.9	68.2
Multinomial NB	63.9	64.3

Table 18: Models Performance on Test Data

Post Hyper-pamater Tuning Best 2 models were chosen for hyper-parameter tuning:-

- SVC
- Logistic Regression

Model	Accuracy	F1 Score
Logistic Regression	69.8	69.8
SVC	69.8	69.6

Table 19: Models Performance post Hyper-param Tuning

Confusion Matrix on Test Data

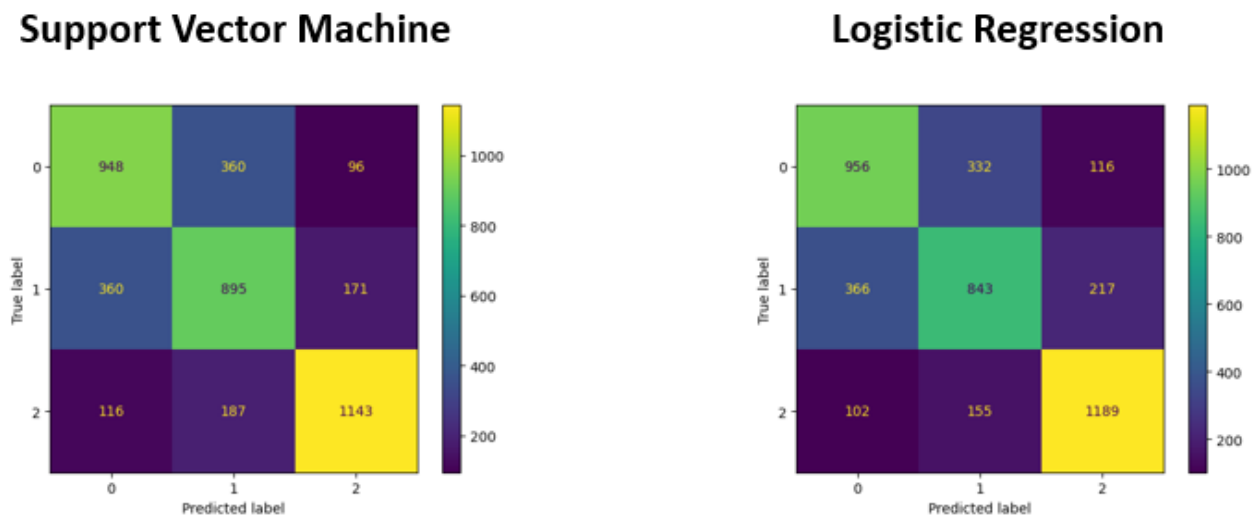

Figure 19: Confusion Matrix on Test Data - ML

Observation: After hyper-parameter tuning on the best 2 ML models, there was hardly any improvement observed on test data. The confusion matrix shows a lot of true negative predictions for all the three labels.

Approach 2 (Best): When data pre-processing was done integrating negative and neutral reviews more external sources as explained in section4.2.2.

Total Reviews: 20,748 (Reduced from approach 1 after applying more pre-processing techniques)

Rating	Total Reviews
1	3559
2	3395
3	6741
4	3586
5	3467

Table 20: Review Distribution / Rating

Train / Test Split: 80/20

Train Data Distribution

Sentiment Label	No. of Reviews
0	5563
1	5392
2	5642

Table 21: Review Distribution / Rating

Model	Accuracy	F1 Score
Logistic Regression	73.1	72.9
Random Forest	69.6	69.4
KNeighbour	44.6	35.5
SVC	73.5	73.4
Multinomial NB	68.9	68.3

Table 22: Models Performance

Post Hyper-pamater Tuning Best 3 models were chosen for hyper-parameter tuning:-

- SVC
- Logistic Regression
- Random Forest

Model	Accuracy	F1 Score
Logistic Regression	73.01	72.6
Random Forest	69.54	69.28
SVC	72.87	72.73

Table 23: Models Performance post Hyper-param Tuning

Observation: After hyper-parameter tuning on the best 3 ML models, the accuracies degraded by a marginal approx. 0.3 %. However, they were improved and showcased better results than what was observed in Approach 1.

Test Data Distribution

Sentiment Label	No. of Reviews
0	1391
1	1349
2	1411

Table 24: Review Distribution / Rating

Model	Accuracy	F1 Score
Logistic Regression	73.1	72.9
Random Forest	69.6	69.4
KNeighbour	44.6	35.5
SVC	73.5	73.4
Multinomial NB	69.9	69.3

Table 25: Models Performance on Test Data

Post Hyper-pamater Tuning Best 2 models were chosen for hyper-parameter tuning:-

- SVC
- Logistic Regression

Model	Accuracy	F1 Score
Logistic Regression	73.01	72.6
SVC	72.8	72.7

Table 26: Models Performance post Hyper-param Tuning

Confusion Matrix on Test Data

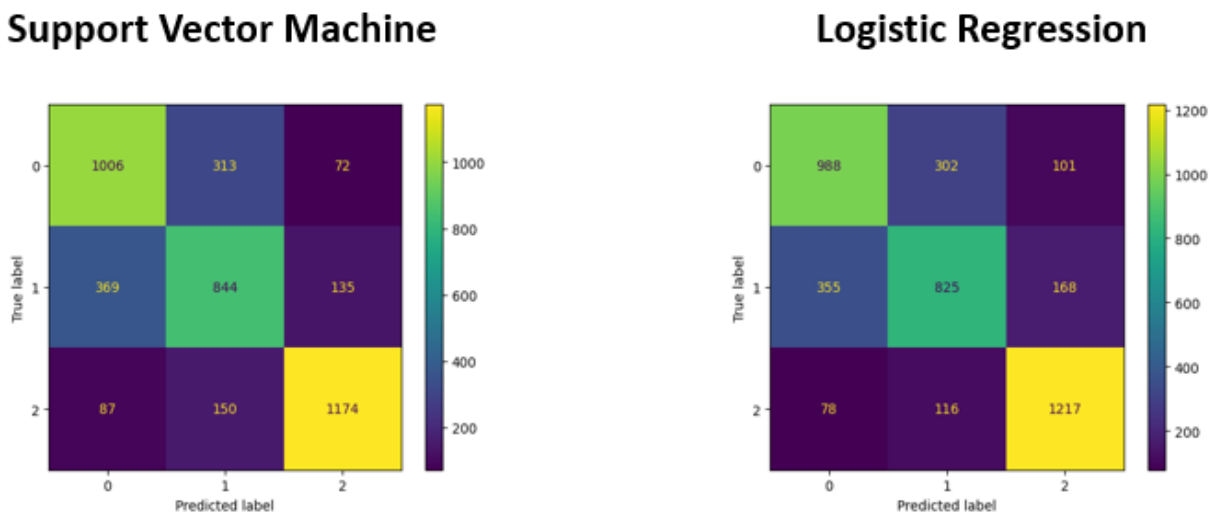

Figure 20: Confusion Matrix on Test Data - ML

Observation: After hyper-parameter tuning on the best 2 ML models, scores dropped down by approx. 0.5%, however the overall results on external test data set were highly encouraging as explained and discussed in section 5.2.

5.1.2 BERT Models

The distilBERT models are directly applied on the Approach 2 as explained in section 4.2.2.

Total Reviews: 21,415
Train / Test Split: 70/15/15

Train Data Distribution

Sentiment Label	No. of Reviews
0	5325
1	4715
2	4950

Table 27: Sentiment Distribution / Rating

Epoch	Batch Size	Learn Rate	Drop Rate	Train Acc	Val Acc	Test Acc	Train Loss	Val Loss
4	64	2e-5	0.3	82.4	72.3	74.2	0.6	0.9

Table 28: BERT Model Performance

Validation Data Distribution

Sentiment Label	No. of Reviews
0	1141
1	1010
2	1061

Table 29: Sentiment Distribution / Rating

Validation Data Classification Report

Label	Precision	Recall	F1
0	73	67	70
1	58	64	61
2	86	86	86

Table 30: BERT - Validation Data Classification Report

Test Data Distribution

Sentiment Label	No. of Reviews
0	1391
1	1349
2	1411

Table 31: Sentiment Distribution / Rating

Test Data Classification Report

Label	Precision	Recall	F1
0	73	70	71
1	60	62	61
2	86	87	87

Table 32: BERT - Test Data Classification Report

Confusion Matrix on Validation & Test Data

Figure 21: Confusion Matrix on Val and Test Data - ML

Observation & Summary: The overall distilBERT model test results were way better than ML conventional models as also discussed and proved in section 5.2, hence BERT model distilBERT was encouraged thereafter and cross-validations and more extensive testing for performed on it rather than ML as showcased in 5.3 as well.

5.2 Testing on Held-out External Test Set

Purpose: This testing is carried out on the new set of reviews extracted from the review websites.

Total External Reviews: 27,596

5.2.1 Deploying Best ML Models

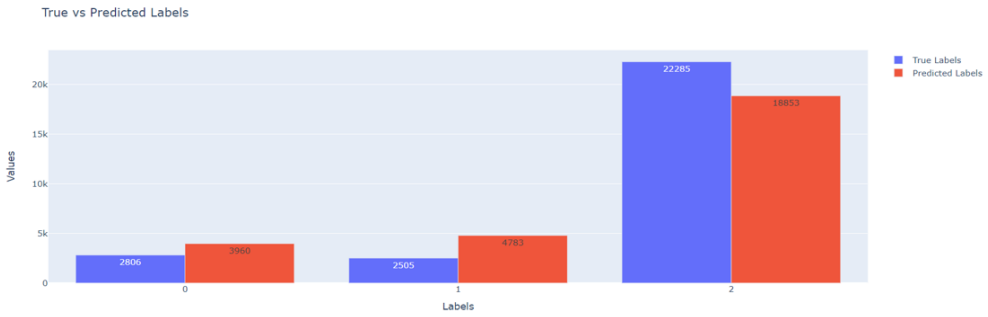

Figure 22: Best ML Models Result on External Dataset

Observation

- Negative (label 0) predictions are more by approximately 18% than actual.
- Neutral (label 1) predictions are more by approximately 32% than actual.
- Positive (label 2) sentiment predictions are lesser by approximately 8% than actual.

5.2.2 Deploying Best BERT Models

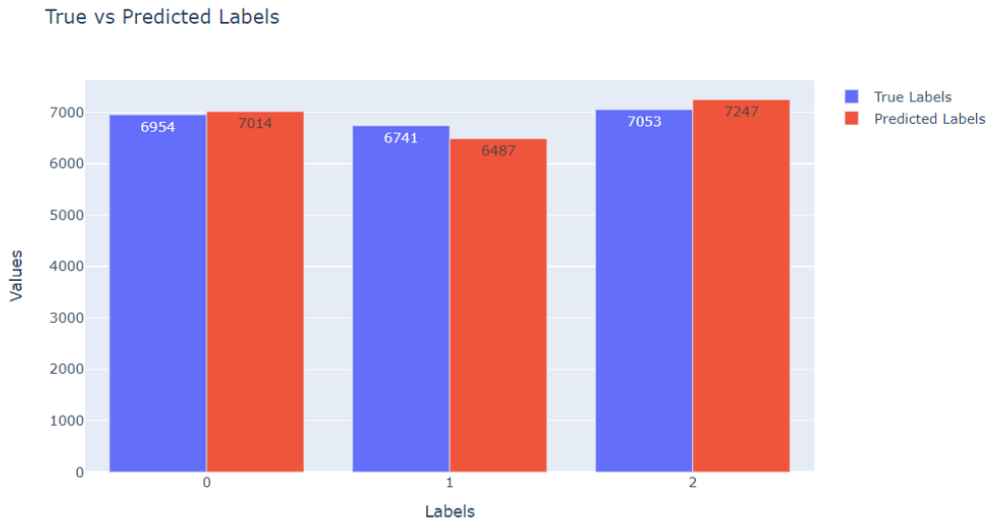

Figure 23: Best BERT Models Result on External Dataset

Observation

- Negative (label 0) predictions are more by only approximately 0.8%.
- Neutral (label 1) predictions are less by only approximately 3.7% than actual.
- Positive (label 2) sentiment predictions are more by approximately 2.7% than actual.

Summary

- Overall, distilBERT performed way better than best ML models in sentiment prediction
- Since, distilBERT performed much better, hence further testing and deployment should be carried only using BERT models
- Further testing sections 5.3, 5.4 and 5.5 are covered using distilBERT models only

5.3 A/B Testing(Split Testing)

Purpose: A/B testing is carried out in which multiple distilBERT model versions will be compared in terms of accuracy and performance. Different **cross-validations** were performed under this testing to figure out the best model for label (0,1 and 2) predictions.

Strategy: The strategy was to train and test DistilBERT model with different split sizes and keeping the rest of the configurations same per split. Below is the strategy explained in enumerated points:-

1. Split the pre-processed data into train and validation dataset only
2. Rest of the configurations such as Batch Size, Token Length, Learning Rate, Dropout Rate, No. of Epoch are set
3. Evaluate the metrics and scores on 5 different seeds for every split and set of configurations. Different seeds used: 42, 101, 789, 1001, 2023
4. Calculate and display the mean train, validation accuracies and losses along with their standard deviation
5. Based on the evaluation of mean accuracies, losses, standard deviation, and other metrics, change the set of configurations and split size
6. Perform the same until expected results are achieved on the training and validation dataset
7. Finally test the model with best results on the external test data

Four different cross validation approaches were performed and results are covered under Appendix section 9.

Overall Summary

- Best model to be deployed in the customer feedback analysis web application was decided on the basis of the results and scores achieved after performing cross validations as explained in Appendix 9 in detail under section 9.1
- Best model was depicted by approach 3 9.1.3 under 9.1
- Below images 24, 25, 26 and 27 showcases first 3 out of 4 total cross validations approaches since the 4th one wasn't relevant but just an experiment, that approach 3 turned out to be the best amongst all in terms of confusion matrix results, training/validation accuracies and losses
- Table 33 shows that the approach 3 scores came out to be better than the rest approaches

- **External Data Testing**: As elaborated under sub-section 5.2.2 within 5.2, deploying best BERT model on the external data produced satisfactory results

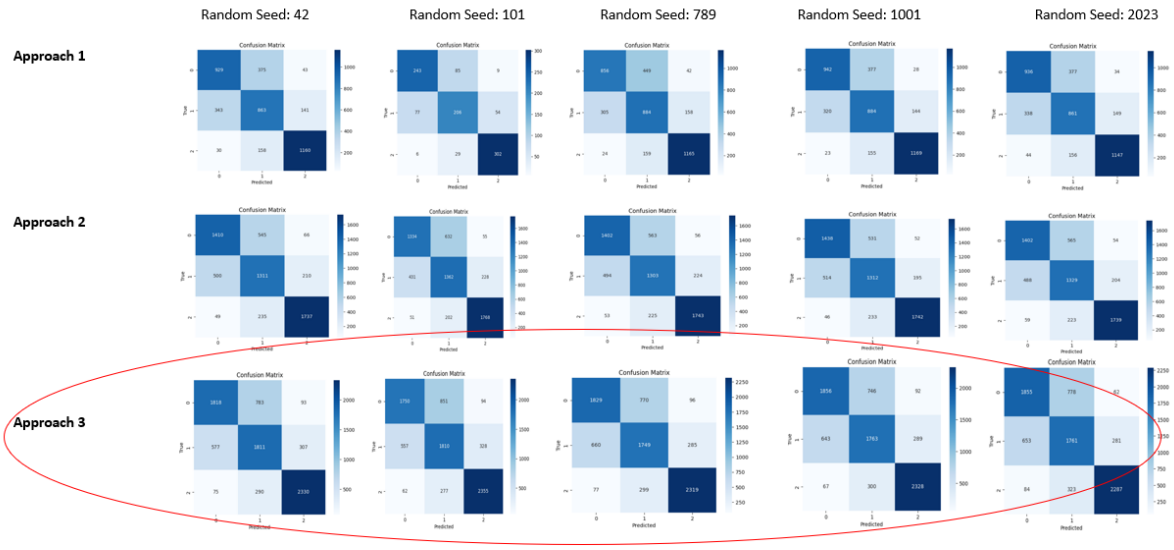

Figure 24: Confusion Matrix - Summary

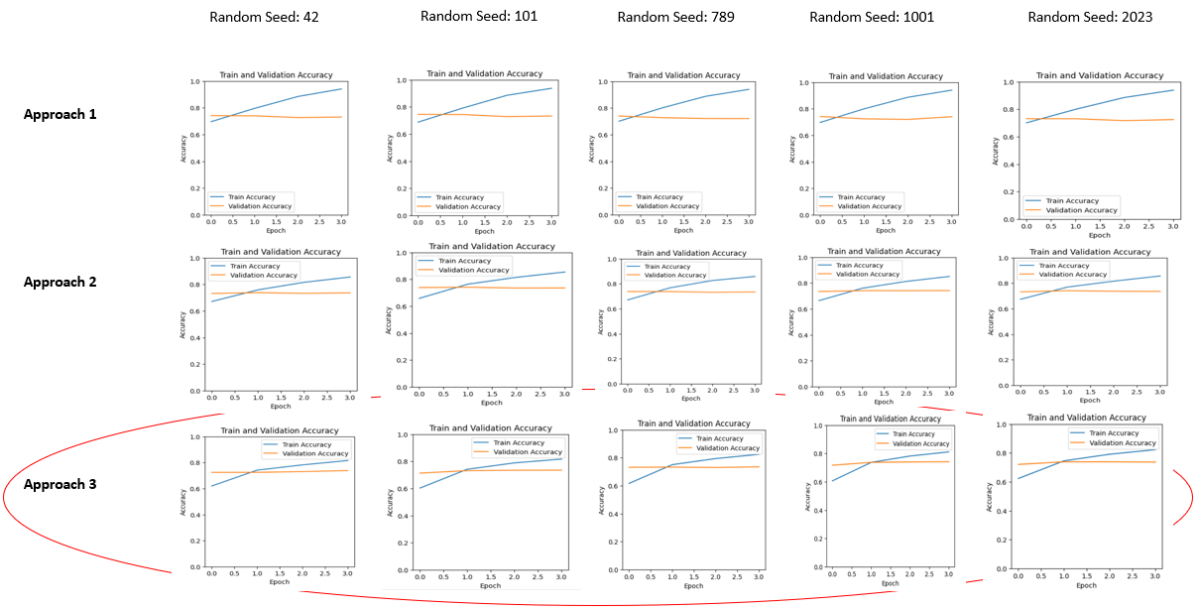

Figure 25: Training & Validation Accuracy - Summary

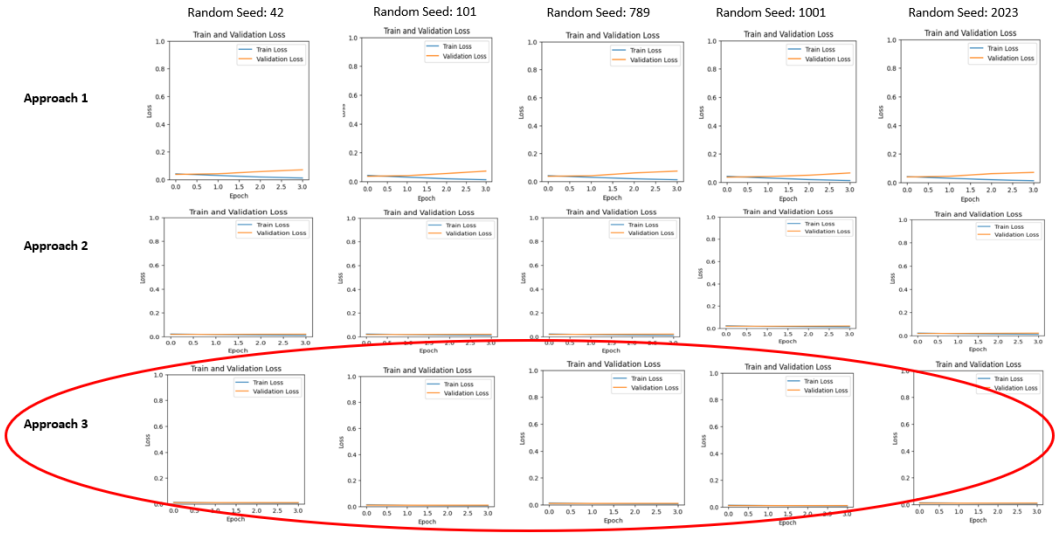

Figure 26: Training & Validation Loss - Summary

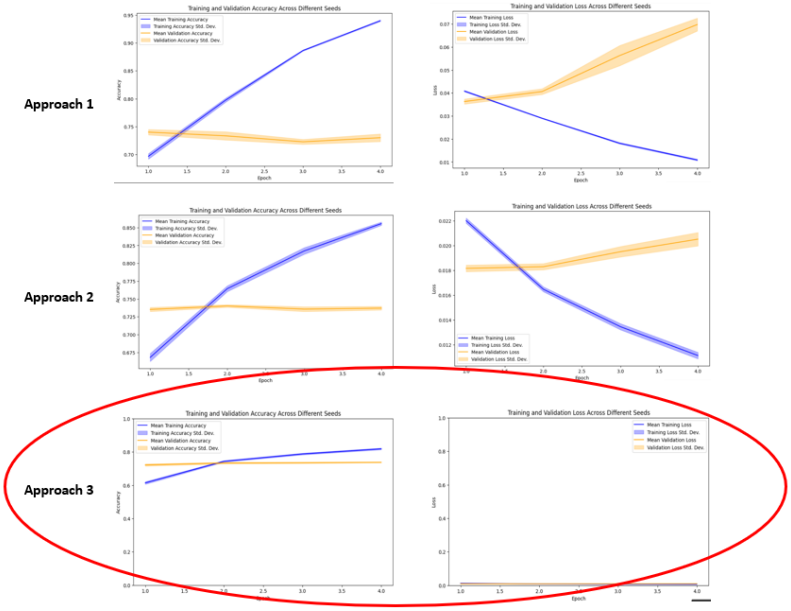

Figure 27: Confusion Matrix - Summary

Approach	Label 0			Label 1			Label 2		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Approach 1	71	69	70	62	64	63	86	86	86
Approach 2	72	70	71	63	65	64	86	86	86
Approach 3	74	74	**74**	68	67	**67.5**	88	86	**87**
Approach 4	72	67	70	61	65	63	86	85	85

Table 33: Precision, Recall, and F1 Score for Labels 0, 1, and 2 across Approaches 1, 2, 3, and 4

5.4 Regression & Integration Testing

Purpose: Regression and Integration testing will be carried out at every stage of new development to make sure the existing code / product does not break or stop functioning because of new features or code integration.

Test ID	Integration point	Test Description	Expected result
P1	Part 1 (Predict sentiments behind reviews)	Test the functionality of predicting the sentiments behind customer reviews for Wex Photo Video on Trust Pilot and Power Reviews. Run bert_dashboard.py and click on the localhost web app link Enter the input time in displayed format in the text field and click Submit button.	Wex CFA web app returns the histogram charts with actual vs predicted values. final_bert_predictions.csv is saved at the backend for the user to view the predictions alongside with reviews and other data.
P2	Part 2 (Issue Type Classification)	Test the issue type classification by the spaCy phrase matcher package for the reviews which are suspected to have issues. Enter the input time same as the previous P1 test case and click Submit button. The system should display histogram charts per review platform for the identified issue category: Electronics, Packaging, Delivery, Time, Customer Service and General	Wex CFA web app returns the histogram bar per review platform per issue type identified.

Table 34 –

Test ID	Integration point	Test Description	Expected result
P3	Part 3 (Reason Behind Issues Generation)	Test the reason generated by Llama3 model for the reviews which contains issues as per the system. Enter the input time same as the previous P2 test case and click Submit button. The system enables a Download CSV button on the webapp which when clicked, downloads the csv file containing the reviews, with predictions, issues identified and reason behind issues.	The CFA web-app once loaded successfully, it enables the Download CSV button which once clicked, downloads the CSV file containing the required data with reasons behind identified issues for the customer reviews.
TC1	Part 1 and Part 2	Test sentiment prediction and issue type classification. Enter input time on web-app and click Submit button. The system should display true vs actual prediction histogram bar charts along with issues identified behind reviews histograms per review platform per issue type.	System displays sentiment behind issues prediction histogram bar charts alongside issue type histograms per review platform.

Table 34 –

Test ID	Integration point	Test Description	Expected result
TC2	Part 2 and Part 3	Test issue type classification and reason behind issues generation. Enter input time on web-app and click Submit button. The system should display the histogram bar charts per review platform for every issue identified related to electronics, packaging, time, customer service, delivery or general category. The system should also enable the Download CSV button which once clicked, should download csv file containing all the relevant review data containing reason behind issues.	CFA system displays the issue classification data and enables Download CSV button accordingly. The button once click, downloads the csv file successfully containing all the relevant review data with possible reasons generated behind issues.
TC3	Part 1 and Part 3	Test sentiment behind issue prediction and reason behind issues generation. Enter input time and click Submit button. The system should generate prediction results along side with Download CSV button which once clicked downloads the csv file containing all the relevant data containing reviews and identified issues with possible reason behind issues.	The CFA system displays the predictions data alongside enables the Download CSV button which once clicks downloads the csv file containing all the relevant data.

Table 34 –

Test ID	Integration point	Test Description	Expected result
TC4	Part 1, Part 2 and Part 3	Test predictions, issue type classification and possible reasons behind issues generation. Enter input time and click Submit button. System should generate model sentiment predictions, issues histogram bar charts and enable Download CSV button which once clicked downloads all csv file containing all the relevant data with reasons behind identified issues.	CFA system displays true vs predicted valaues in the form of histogram bar charts. Alongside, it displays the issues histogram per review platform for the identified issues. It also enables the Download CSV button which once clicked, downloads csv files containing all the relevant review data with possible reasons generated behind the reviews which were issue classified.

5.5 Web Application Usability Testing

Purpose: Usability testing was carried out to test the end product by an external user. The user was one of the existing UEA Postgraduate CMP student. Below are the scenarios tested:-

ID	Description	Preconditions	Test Steps	Expected Results	Actual Results
TC01	Generate sentiment predictions	Flask web-app server is up and running	1. Enter input time 2. Click Submit button	System should display True vs Predicted histogram bar charts	CFA system displays predictions data generated by model in the form of histogram bar charts.
TC02	Generate issue type classification	Flask web-app server is up and running	1. Enter input time 2. Click Submit button	System should display issue type histogram bar charts per review platform	CFA system displays issue type histogram bar charts per review platform
TC03	Generate reason behind issues	Flask web-app server is up and running	1. Enter input time 2. Click Submit button	System should enable Download CSV button which once clicked shown download csv containing all the relevant reviews data with possible reason behind classified issues.	CFA system enables the Download CSV button which once clicked, downloads CSV containing all the relevant reviews data.

ID	Description	Preconditions	Test Steps	Expected Results	Actual Results
TC04	Verify and validate downloaded csv data	Flask web-app server is up and running	1. Enter input time as *today* 2. Click Submit button	System should enable Download CSV button once page is successfully loaded after clicking Download CSV button. The CSV should contain relevant review data on current date. The reviews should be matching with current day's reviews on Trustpilot and Power Reviews for Wex Photo Video.	System enables Download CSV button accordingly. The csv contains valid today's date reviews data as per review platforms Trustpilot and Power Rewviews for Wex Photo Video.
TC05	Verify validation on clicking Submit button on web-app without entering an input time	Flask web-app server is up and running	1. Load web-app server 2. Click Submit button without entering a time	System should display appropriate error message and no data should be displayed	The CFA system displays error message without any data notifying the user to enter input time first.
TC06	Verify validation on clicking Submit button on web-app when an invalid input time is entered	Flask web-app server is up and running	1. Load web-app server 2. Click Submit button by entering a time in unexpected format	System should display appropriate error message and no data should be displayed	The CFA system displays error message without any data notifying the user to enter valid time in expected / appropriate format.
TC07	Verify CFA web-app is loaded successfully on all web browsers	Flask web-app server is up and running	1. Load web-app server on different browsers such as Google Chrome, Safari, IE, Mozilla Firefox	CFA web-app should be loaded on all browsers	The CFA web-app is loaded on all the browsers successfully and returns the data accordingly.
TC08	Verify CFA web-app data is loaded successfully on all web browsers on clicking Submit button	Flask web-app server is up and running	1. Load web-app server on different browsers. 2. Enter valid time and click Submit button	CFA web-app data should be loaded on all browsers	The CFA web-app data is displayed accordingly.

Table 35: Usability Test Cases

6 Evaluation and Discussion

6.1 Sentiment Predictions via ML

Predicting sentiments via Machine Learning models comprised of three different approaches / experiments. The below three approaches discusses and evaluates the performance of ML models in predicting sentiments behind customer reviews. Since the performance and results of ML models were highly dependent upon the pre-processing techniques (as also demonstrated in 4.2) applied over the extracted dataset, therefore it is explained below that how the results of ML models vary in different approaches.

6.1.1 Approach 1

- Data extraction
- Pre-processing
  - Term-Frequency Inverse Document Frequency (TF-IDF) Random Oversampling for data balancing
  - Contractions, emojis, special characters removal
  - Text Lemmatization and STOP words removal
- ML model creation and training
  - Split ratio: 80/20 (Train / Validation)
  - Random Seed: 42

After training various models on a total pre-processed customer reviews data of 30,000 tuples, below are the results:-

Model	Accuracy	F1 Score
SVC	49.1	38
Logistic Regression	50.9	39.7
Random Forest	44	36.6
KNN	45.2	35.4
Naive Bayes	42.2	34.2

Table 36: Models Performance before Hyper-param Tuning

Hyper-parameter tuning was performed on the best two models based on their respective scores.

Model	Accuracy	F1 Score
SVC	52.48	38.96
Logistic Regression	52.83	39.22

Table 37: Models Performance after Hyper-param Tuning

Confusion Matrix

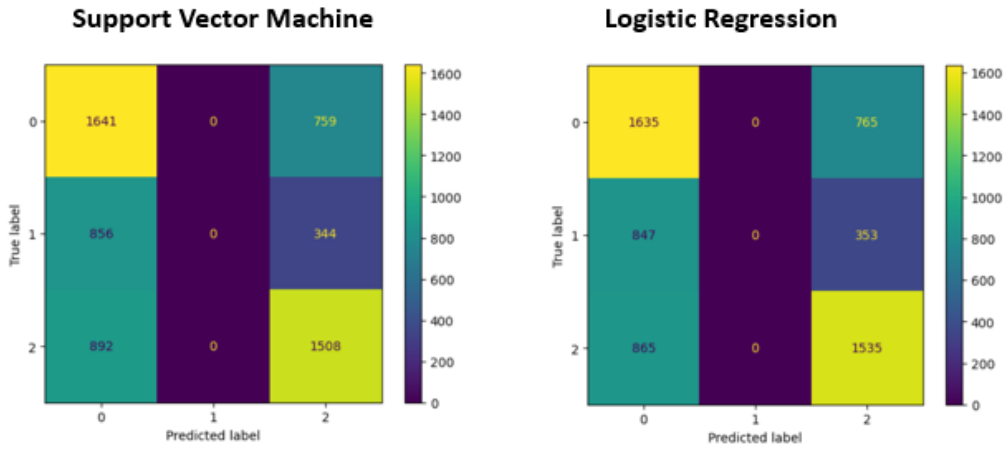

Figure 28: Validation Data Confusion Matrix

Label	SVM		
	Precision	**Recall**	**F1 Score**
0	48	68	57
1	0	0	0
2	58	63	60
Label	**Logistic Regression**		
	Precision	**Recall**	**F1 Score**
0	49	68	57
1	0	0	0
2	58	64	61

Table 38: ML Approach 1 - Scores

Quick Summary: The confusion matrix and scores of top 2 ML models shows a highly poor performance in predicting the sentiment labels, especially for label 1 (neutral) sentiment. Infact, for neutral sentiment, the models could not predict at all.

6.1.2 Approach 2

- Data extraction
- Pre-processing
  - **Data balancing using more neutral sentiments retrieved from external kaggle resources** [16] [17]
  - Contractions, emojis, special characters and **duplicate** removal
  - Text Lemmatization and STOP words removal
- ML model creation and training
  - Split ratio: 80/20 (Train / Validation)
  - Random Seed: 42

After training various models on a total pre-processed customer reviews data of 21,378 tuples (reduced since duplicates are removed in this approach), below are the results:-

Model	Accuracy	F1 Score
SVC	68.9	68.2
Logistic Regression	69.7	69.6
Random Forest	62.6	62.3
KNN	47.3	43
Naive Bayes	63.9	64.3

Table 39: Models Performance before Hyper-param Tuning

Hyper-parameter tuning was performed on the best two models based on their respective scores.

Model	Accuracy	F1 Score
SVC	69.83	69.83
Logistic Regression	69.87	69.63

Table 40: Models Performance after Hyper-param Tuning

Confusion Matrix

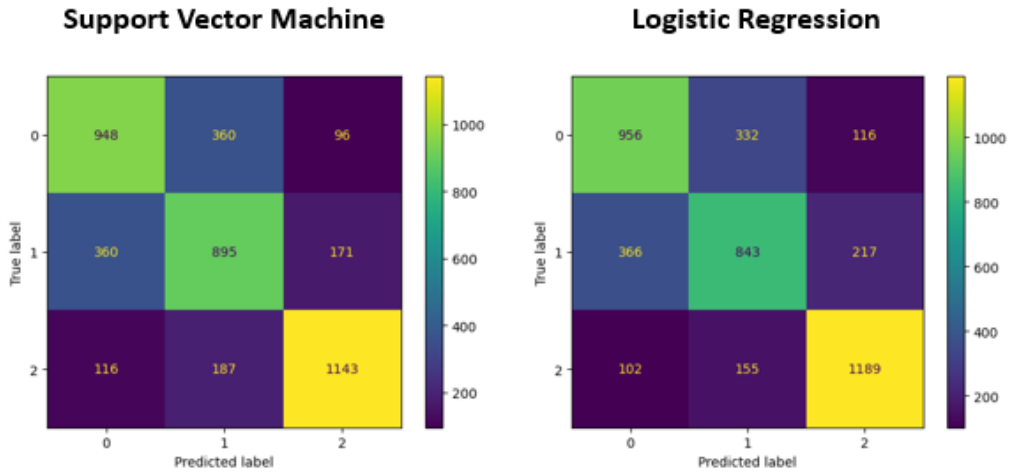

Figure 29: Validation Data Confusion Matrix

Label	SVM		
	Precision	**Recall**	**F1 Score**
0	67	68	67
1	62	63	62
2	81	79	80
Label	**Logistic Regression**		
	Precision	**Recall**	**F1 Score**
0	67	68	67
1	63	59	61
2	78	82	80

Table 41: ML Approach 2 - Scores

Quick Summary: The confusion matrix and scores of top 2 ML models shows a significant improvement in sentiment predictions comparatively to approach 1.

6.1.3 Approach 3

- Data extraction
- Pre-processing
  - Data balancing using more neutral sentiments retrieved from external kaggle resources [16] [17]
  - Contractions, special characters and duplicates removal
  - **Emoji replacement**
  - **Exclusion of Text Lemmatization and controlled STOP word removal**
- ML model creation and training
  - Split ratio: 80/20 (Train / Validation)
  - Random Seed: 42

After training various models on a total pre-processed customer reviews data of 20,748 tuples (reduced since more records were dropped in this approach during pre-processing phase), below are the results:-

Model	Accuracy	F1 Score
SVC	73.5	73.4
Logistic Regression	73.1	72.9
Random Forest	69.6	69.4
KNN	44.6	35.5
Naive Bayes	69.9	69.3

Table 42: Models Performance before Hyper-param Tuning

Hyper-parameter tuning was performed on the best two models based on their respective scores.

Model	Accuracy	F1 Score
SVC	72.86	72.73
Logistic Regression	73.01	72.65

Table 43: Models Performance after Hyper-param Tuning

Confusion Matrix

Figure 30: Validation Data Confusion Matrix

Label	SVM		
	Precision	**Recall**	**F1 Score**
0	69	72	71
1	65	63	64
2	85	83	84
Label	**Logistic Regression**		
	Precision	**Recall**	**F1 Score**
0	70	71	70
1	66	61	64
2	82	86	84

Table 44: ML Approach 3 - Scores

Quick Summary: The confusion matrix and scores of top 2 ML models shows even more improvement in sentiment labels prediction after even more refined and micromanaged pre-processing techniques in approach 3.

6.1.4 Testing Best Approach on External Test Dataset

Total Reviews: 20,703

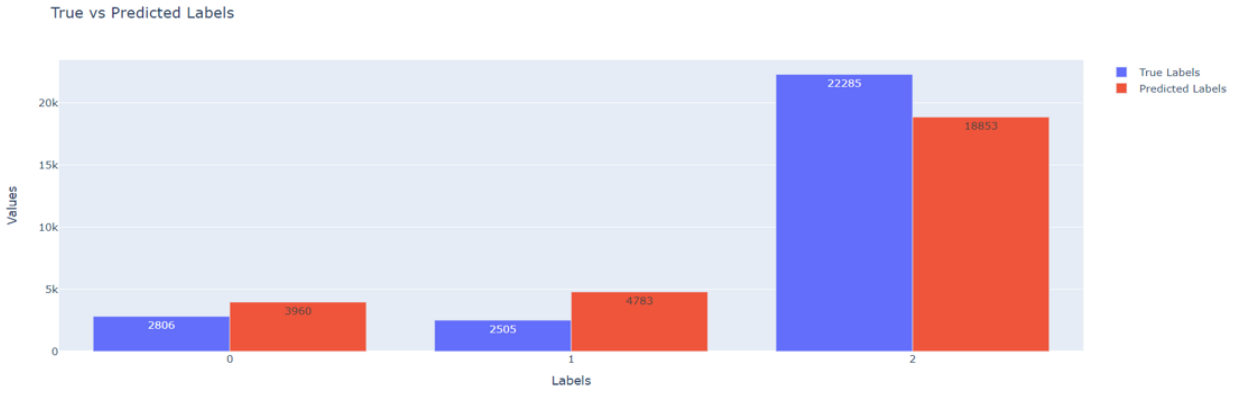

Figure 31: Best ML Models Result on External Dataset

6.1.5 Summary

- Accuracy and F1 scores increased by approximately 20% from approach 1 to approach 2.
- Accuracy and scores increased by another approximately 4% after gibberish reviews removal from approach 2 to 3.
- Random oversampling in approach 1 6.1.1 created a lot of redundant synthetic data which lead to bias towards label 2.

6.2 Sentiment Predictions via BERT

The sentiment predictions by BERT based model called distilBERT are evaluated and discussed in detail under sub-sections 5.1, 5.2 and 5.3 with testing section 5.
To summarise again, there were 4 different cross validations approaches performed to figure out the best suited BERT based distilBERT model for sentiment labels prediction as demonstrated in section 9.1. The best suited model came out to be the one derived via 9.1.3. Here is the quick summary of the best evaluated and suited bert-based distilBERT model for sentiment prediction :-
Configurations and Parameters

- Train/Val Split Ratio: 60 (Train) / 40 (Validate)
- Random Seed: 42, 101, 789, 1001, 2023
- Epoch: 4
- Batch Size: 64
- Learning Rate: 2e-5
- Dropout Rate: 0.3
- Token Length: 512

The model was trained upon a total of 20,208 records with 5 different random states and concluded to be the best suited based on metrics and scores such as precision, recall, f1 scores, confusion matrix, mean training and validation accuracies and losses respectively and standard deviation for the same.

6.2.1 Testing Best Approach on External Test Dataset

Total Reviews: 20,748

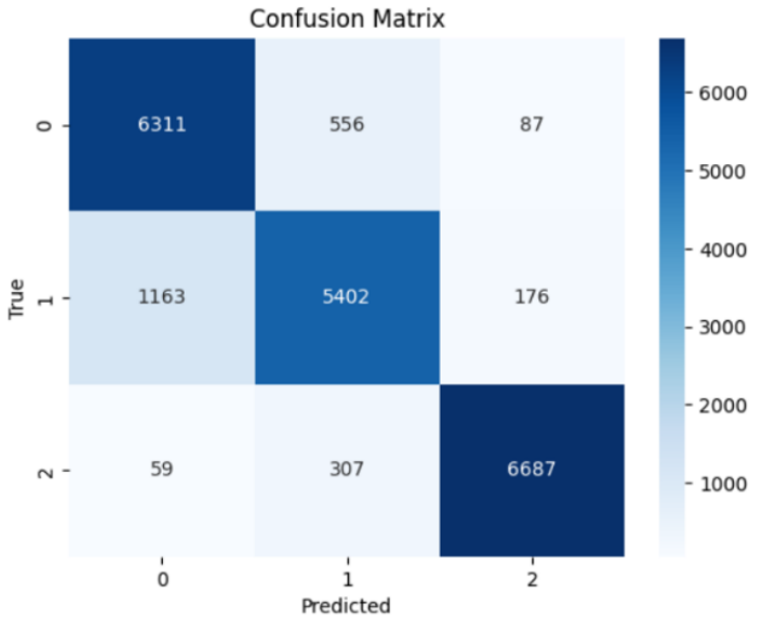

Figure 32: Best BERT Model Confusion Matrix

External Data Classification Report

Label	Precision	Recall	F1
0	78	80	79
1	74	71	72
2	90	92	91

Table 45: BERT - External Data Classification Report

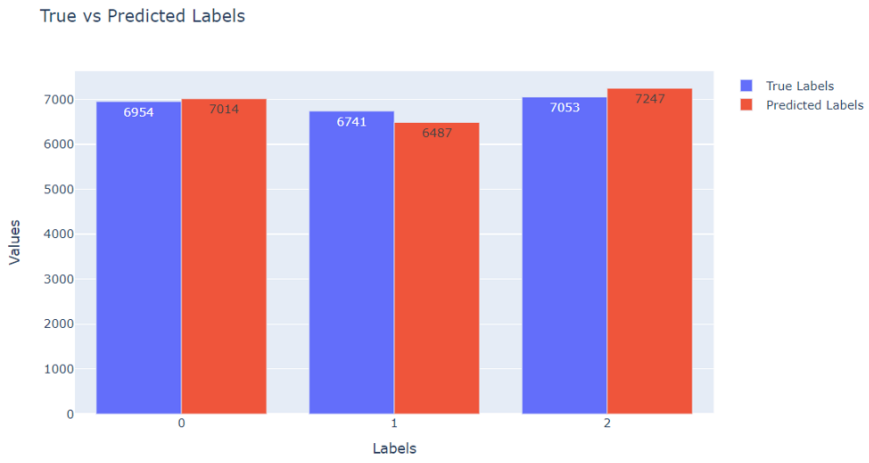

Figure 33: Best BERT Models Result on External Dataset

6.2.2 Summary

- Same as 5.3 within testing section 5
- The above bullet point refers to the extensive cross validation testing performed on the training and validation customer reviews data to depict best suited model
- In addition to above point, distilBERT best model performed really well on the external test dataset as well as demonstrated above
- Negative predictions made more by only approximately 2% than actual on the external dataset.
- Neutral predictions made less by only about 1.5% than actual on the external dataset.
- Positive predictions made less by only about 1.7% than actual on the external dataset.

6.3 ML vs BERT

Comparing the best approaches to identify best suited models for ML and BERT respectively would give a better vision on which model performed better in terms of scores and results on the validation data.
Classification Reports of ML vs BERT on Validation Data

Label	Precision	Recall	F1
0	69	72	71
1	65	63	64
2	85	83	84

Table 46: ML - Validation Data Classification Report

Label	Precision	Recall	F1
0	74	74	**74**
1	68	67	**67.5**
2	88	86	**87**

Table 47: BERT - Validation Data Classification Report

Confusion Matrix of ML vs BERT on Validation Data

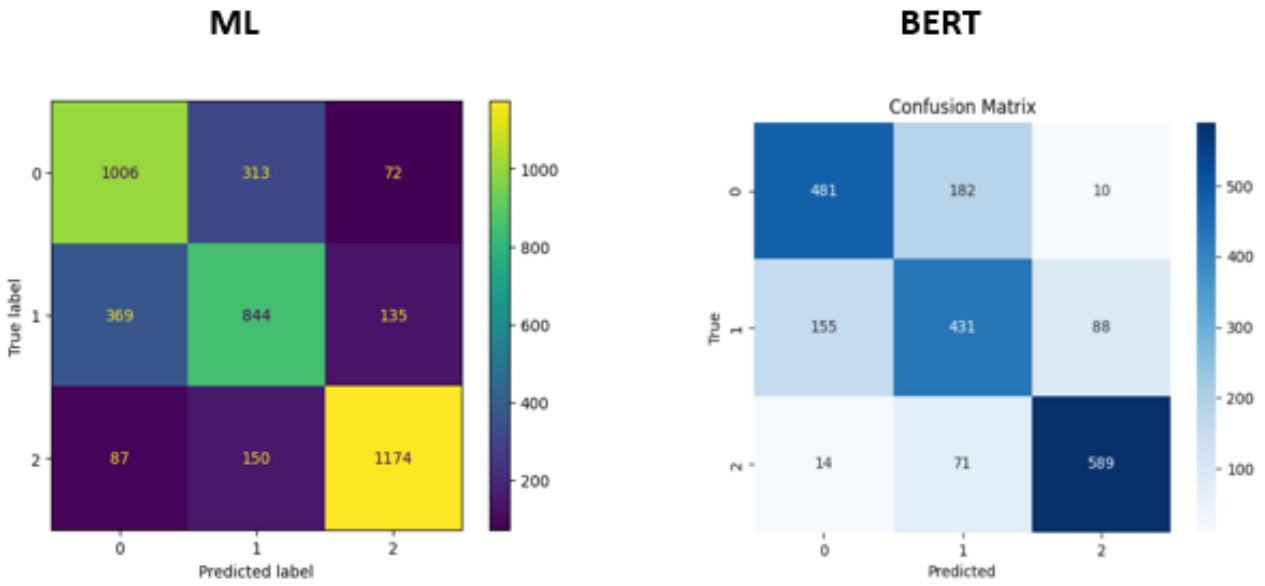

Figure 34: ML vs BERT - Validation Data Confusion Matrix

6.3.1 Why distilBERT performed better than ML ?

1. Contextual Understanding

ML: Traditional machine learning approaches, such as logistic regression and Support Vector Classifiers (SVC), rely on simple features like word counts (bag-of-words) or TF-IDF (Term Frequency-Inverse Document Frequency). These methods fail to capture the context or deeper meaning of words within phrases. For example, they treat the word "good" in "not good" as a positive adjective, ignoring the negative sentiment conveyed by the phrase.

DistilBERT: In contrast, DistilBERT, a transformer-based model, excels at understanding context. It recognizes that the word "good" in "not good" should be interpreted negatively, thanks to its ability to analyze entire phrases and understand word relationships. DistilBERT is a smaller, faster variant of BERT, reducing the model size by 40% and increasing speed by 60%, while retaining 97% of BERT's language understanding capabilities [19].

2. Deep Representation of Text

ML: ML models usually work with superficial features. For instance, word order and more subtle linguistic aspects are disregarded by TF-IDF while frequency data is captured. They consequently have a limited capacity to distinguish between minute variations in mood.

DistilBERT: DistilBERT generates rich, multi-dimensional text representations by applying deep neural network layers. Because it can recognize more sophisticated patterns in language, it can interpret sentiment cues that traditional models might miss, such sarcasm or faint emotional undertones. The **self-attention** mechanism allows the model to capture the relationships between words, even if they're far apart in the sentence, helping it understand the full meaning more effectively [20].

3. Pre-trained Knowledge

ML: On the input dataset given, traditional ML models are typically trained from scratch. Their comprehension of language is restricted to the particular dataset they are trained on. A limited or undiversified dataset will make it harder for the model to generalize.

DistilBERT: DistilBERT is a pre-trained model that has been extensively trained using text data from webpages, books, and articles on the internet. It contains 66 million parameters. As a result, it has a solid fundamental grasp of language, including sentiment and subtle connotations. It begins with this deep language understanding and tailors it to your particular activity, resulting in better performance when fine-tuned on your sentiment dataset [19].

6.4 Issue Type Classification via spaCy PhraseMatcher

The PhraseMatcher in spaCy is a powerful tool used to efficiently match exact phrases in a text. It works by first converting the phrases you want to match into tokenized patterns, and then using a fast and efficient algorithm to find those patterns in the text [21].

The objective was that once sentiment is predicted by the model, then also to identify the possible issues especially behind the reviews which are predicted as label 0 or 1, meaning negative or neutral. Not only for label 0 and 1 but even for label 2 which are predicted as positive sentiments but Wex Photo Video claims that they encounter quite many reviews on daily basis which are rated as 4 or 5 stars but still contains some issue or complaint mentioned either related to the product or service of Wex.

These are the possible issues identified by talking to their team and also by going through the reviews on the review platforms :-

- Electronics
- Product
- Packaging
- Time
- Delivery
- Customer service
- General

In order to capture these issues, an NLP (Natural Language Processing) pre-trained large web model was used: *en_core_web_lg*. Using the *PhraseMatcher* library, basic words and phrases associated with these issues were identified and hard-coded in the form of a dictionary. Below is the dictionary containing these issue types and their related phrases:

```
issue_phrases = {
    "Electronics": ["laptop", "camera", "lens", "mobile phone", "smartphone",
                    "tripod", "tablet", "headphone", "charger", "battery"],
    "Product": ["product issue", "defective product", "faulty item",
                "damaged", "damage", "torn", "broken", "break", "gouged"],
    "Packaging": ["packaging", "package", "pack"],
    "Time": ["late", "delayed", "time issue", "too long", "long to arrive"],
    "Delivery": ["delivery issue", "shipment problem", "shipping issue", "arrive",
                 "arrival", "late delivery", "delivery late", "late", "time", "times"],
    "Customer service": ["customer service", "support issue", "service complaint",
                         "rude", "aggressive", "tone", "behaviour", "nonsense",
                         "not understand", "staff", "insult", "salesperson"],
    "Payment": ["payment", "pay", "money"],
    "General": ["issue", "problem", "complaint"]
}
```

Once any of the phrases match in any of the customer review, that review falls within that specific issue category. It is possible that a single review falls into multiple categories.

Here is the sample customer review data with issues depicted:-

Source	Content	Issues
Trustpilot	love my new **camera**, got a nikon zf from here and the service in-store and online was really good. belfast branch let me see and use the display model they had, and answered all my questions in a helpful and friendly way, the also pointed out key features and things to consider plus alternatives if needed.	Electronics
Power Reviews	i have just wasted 90 minutes driving to and from moor allerton as i needed a fast sony lens for a wedding tomorrow. there were two **staff** in the branch and one other customer when i went in. for some bizarre reason both staff members saw fit to **ignore** me and both dealt with the one **customer**. one said he would be with me in a minute and went out back to find something for the one customer they had, he was actually gone about 5 or 10 minutes, came back with something like a **lens** cap, then they both spent even more time with the same customer. the same **staff member** then walks past me again on his way to the store and says, i will be with you in a minute, you know, like he would said 10 minutes before. by the his point i would been stood by the counter for 15-20 mins and i told him not to bother as i was leaving. so wex lost a decent sale and i will never shop there again. appalling and **rude customer service**!	Electronics, Customer Service, Delivery
Trustpilot	initial offer was fine but reduced by around 10% on a boxed **lens** in good condition. took quite a few days to make the offer after the lens was received. however the **money** owed was paid promptly. i do not plan to sell anything else through wex	Electronics, Payment

Table 49: Customer Reviews with Issues

6.5 Possible reason behind issues generation via Llama3

The previous subsection 6.4 shows that how spaCy pre-trained large model and PhraseMatcher library is used to depict the possible issues behind the customer reviews by matching the hard-coded set of phrases in the reviews. However, there are two major drawbacks in phrase matching technique. The first drawback is limited phrases matching. The *dictionary* 6.4 maintained is hard-coded and needs time to time revision and update so that the system can depict issues by matching the newly added phrases. This limits the functionality and requires a time to time manual intervention by going through the reviews and checking for new or unseen phrases related to the issues and finally incorporate them in the dictionary. The second drawback is that even for the reviews in which customer has expressed complete delight and happiness but has mentioned the phrases which may match the phrases / issues dictionary, even then the system would make that review fall into an issue or multiple issues category. Now this could be a genuine case as well, where despite of giving 5 ratings to the review, the customer has however actually mentioned a review or complaint related to any of the product or services of Wex Photo Video. But it might not be a good idea to categorise the review with an issue even when there isn't any. Additionally, the Wex team also demanded that if there an issue depicted with reviews, there should be a functionality which helps in predicting or generating the possible reason behind the issues. This feature would not only generate the possible reason behind the issue of the review but also verify the genuineness of it. Below is a vivid example demonstrating the same by using the same table above within section 6.4:-

Source	**Content**	**Issues**	**Reason**
Trustpilot	love my new **camera**, got a nikon zf from here and the service in-store and online was really good. belfast branch let me see and use the display model they had, and answered all my questions in a helpful and friendly way, the also pointed out key features and things to consider plus alternatives if needed.	Electronics	The review appears genuine, well-written, and provides specific details about the customer's experience with the store and online service.

Table 50 –

Source	Content	Issues	Reason
Power Reviews	i have just wasted 90 minutes driving to and from moor allerton as i needed a fast sony lens for a wedding tomorrow. there were two **staff** in the branch and one other customer when i went in. for some bizarre reason both staff members saw fit to **ignore** me and both dealt with the one **customer**. one said he would be with me in a minute and went out back to find something for the one customer they had, he was actually gone about 5 or 10 minutes, came back with something like a **lens** cap, then they both spent even more time with the same customer. the same **staff member** then walks past me again on his way to the store and says, i will be with you in a minute, you know, like he would said 10 minutes before. by the his point i would been stood by the counter for 15-20 mins and i told him not to bother as i was leaving. so wex lost a decent sale and i will never shop there again. appalling and **rude customer service**!	Electronics, Customer Service, Delivery	The review appears legitimate, detailing frustrating experiences with poor customer service and staff prioritizing another customer over the writer's urgent request.
Trustpilot	initial offer was fine but reduced by around 10% on a boxed **lens** in good condition. took quite a few days to make the offer after the lens was received. however the **money** owed was paid promptly. i do not plan to sell anything else through wex	Electronics, Payment	The main issue seems to be the initial offer being reduced by 10%, which may impact trust and satisfaction with Wex's service.

Table 50: Reason Behind Issues via Llama3

One of the drawback we discussed above was related to the genuineness of the depicted issue, that whether there was actually in issue within the customer review or not. This feature suffices the problem and reflects on the validity of the issue and generates the possible reason behind the classified issues within the customer review. In the first review within the above table where the review / content expresses complete satisfaction with Wex service, but system depicted issues related to electronics. The pre-trained llama3 model analysed the review and generates the reason text that the review seemed genuine and actually contained no issue. The customer infact shares a very positive experience with Wex Photo Video product and services. Whereas the other two issues do express some dissatisfaction by the customers related to customer service, product and services of Wex. The llama3 model generates the correct reason behind those issues where the customers talk about poor customer service and related to payments and offers published by Wex Photo Video.

Here are the different approaches that were tried and tested in order to generate reason behind the reviews which were categorised into issues:-

Approach 1: Open AI
Gpt4 model was tried and tested using the private api key provided by the open AI [22]. The customer review was given in the input prompt and was asked to identify the issues with it to check the genuineness of the issues identified by the spaCy phrasematcher. Here is the input prompt given: *"Could you tell me if there is any issue with this customer review: '{review}'. Keep the answer within 20 words."*

The '{review}' is the actual customer review which is passed an argument. Here is one of the sample reviews from Trust Pilot which was passed to the prompt :-
initial offer was fine but reduced by around 10% on a boxed lens in good condition. took quite a few days to make the offer after the lens was received. however the money owed was paid promptly. i do not plan to sell anything else through wex

Below is the response generated by openAI API :-
"The issue is a reduction in the initial offer by 10%, potentially impacting customer trust and satisfaction with Wex's service."

Response Time: 0.8 seconds
The response from openAI seems realistic and genuine and the response time was less than a second.

Limitation: Although the response from openAI API was genuine and quick in response but it was paid. OpenAI grants 5 dollars initially to access GPT4.0 for a limited amount of tokens and then 15 dollars is charged per 1M tokens [22]. Hence it was not feasible due to monetary constraints. Therefore, this approach was not considered to integrate in the Wex CFA automation framework.

Approach 2: Llama2
Next were the llama2 models comprising of 7B and 13B parameters, used to test the validity of the issues behind the customer reviews [23]. The same input review was given to the llama2 prompt as used while testing opneAI gpt4 model above 6.5.

Below is the response generated by llama2 7B model:-
"The customer review highlights a concern about trust and satisfaction due to a 10% reduction in the initial offer."
Response Time: 3.6 seconds The response generated by llama2 7B model also appeared to be genuine and realistic but the response time was more than 3.5 seconds on an average when tested for about 10 reviews.

Below is the response generated by llama2 13B model:-
"The 10% drop in the initial offer raises questions in the customer review on happiness and trust."
Response Time: 5.1 seconds
Limitations: The responses from llama2 models consisting of 7B and 13B parameters respectively also appeared valid and accurate but the only major concern was the response times per review. Llama2 7B model was initially thought to be used but then later it was observed that the responses generated by it weren't as accurate for more complex or long reviews. Therefore, even llama2 models were not considered.

Approach 3 (Best): Llama3
Llama3 8B model [24] came out to be the best suited to be integrated into the framework. Reason being, although it took roughly 4 seconds on an average to generate a response for a review, but it gave highly accurate and valid responses even for the complex and longer reviews which were more than 500 words and contained multiple issues.

Below are a few customer reviews which were passed to test the responses generated by llama3 model:-
Review: *"i have just wasted 90 minutes driving to and from moor allerton as i needed a fast sony lens for a wedding tomorrow. there were two staff in the branch and one other customer when i went in. for some bizarre reason both staff members saw fit to ignore me and both dealt with the one customer. one said he would be with me in a minute and went out back to find something for the one customer they had, he was actually gone about 5 or 10 minutes, came back with something like a lens cap, then they both spent even more time with the same customer. the same staff member then walks past me again on his way to the store and says, i will be with you in a minute, you know, like he would said 10 minutes before. by the his point i would been stood by the counter for 15-20 mins and i told him not to bother as i was leaving. so wex lost a decent sale and i will never shop there again. appalling and rude customer service!"*

Response: *"The review appears legitimate, detailing frustrating experiences with poor customer service and staff prioritizing another customer over the writer's urgent request."*

Review: *"super easy no hassle ordering and great quality lens would highly recommend wex"*

Response: *"No apparent issues; the review seems genuine, concise, and highlights positive experiences with Wex's ordering process and product quality."*

Response Time: 4.1 and 4.5 seconds respectively
Although the response time per review on an average is slightly over 4 seconds but the responses appeared genuine and acceptable and works as expected for complex reviews and reviews which didn't have any issues but spaCy phrasematcher identified issues in them. Hence, this was considered to be the best suited model and eventually was integrated in the CFA framework.

Limitations: The responses generated by llama3 were better than llama2 models. However the response time taken by 8B llama3 model was about 4 - 4.5 seconds on an average per review. If N number of reviews are passed to the model, the total amount of time is N * 4. The user has to wait for the system to generate the reasons for all the reviews based on the input time given. If the input time give is to fetch the reviews of the past 30 days or 60 days, the system takes a fairly long time to produce the final CSV.

6.6 User Experience (UX)

- **Ease of Use:** As per the usability testing in 5.5 carried out by different UEA students, the feedback was positive about CFA web framework being easy to use.
- **Response Time:** As discussed in 6.5, the response time to generate the possible reason behind issues is a time consuming process since llama3 8B model takes roughly 4 seconds to generate reason per review. Despite multi-threading being implemented in attempt to make the reason generation process a bit faster by opening multiple threads using the virtual cores of the local machine and assigning N no. of reviews per thread, but even then the response time taken by the model remains the same.
- **User Satisfaction:** The overall response from both of the UEA students who carried usability testing of the CFA framework 9.2 was positive and satisfactory.

7 Areas of Improvement & Future Enhancements

7.1 Part A: Sentiment Prediction

One area for improvement in this research is the ability to extensively test the machine learning and BERT models. Due to memory constraints, the models could not be thoroughly evaluated on much larger datasets or across more extensive hyperparameter tuning. Future work should focus on overcoming these limitations by leveraging more computational resources, such as high-memory GPUs or cloud-based infrastructure, to fully exploit the capabilities of BERT and other deep learning models. This would enable more comprehensive experimentation, resulting in potentially improved model accuracy and robustness in sentiment analysis. Currently, BERT based model called distilBERT comprising of 66M parameter has been used. But given more resources, other models suchas roBERTa, BERTweet, or original BERT model can be trained and evaluated as well.

7.2 Part B: Issues Classification

An area of improvement in the current issue classification approach using the spaCy 'PhraseMatcher' is the reliance on a hard-coded dictionary containing potential issues, words, and phrases to be matched in customer reviews. This static dictionary requires manual updates, which involves continuously monitoring new reviews and adding new phrases or variations over time. This process is labor-intensive and may not capture emerging issues promptly. Future work could focus on automating this issue classification process by leveraging machine learning (ML) or deep learning (DL) techniques. These models could dynamically learn from the data, reducing the need for manual updates and improving the accuracy and adaptability of issue prediction over time.

7.3 Part C: Reason Behind Issues Generation

An area of improvement for this sub section could involve optimizing the response time and resource usage for the current setup. Currently, the LLaMA 3 (8B parameter) model is used via Ollama, which, while providing good results, is slow and consumes significant memory on the local machine. The response time for generating reasons for a single review of around 100 tokens ranges from 4-10 seconds. A potential improvement would be to switch to using the GPT API from OpenAI, which is notably faster, producing results in 1-2 seconds. However, this would require a paid subscription, as the GPT API is a paid service.

7.4 User Experience (UX)

An area of improvement of the current web application involves addressing stability issues. The application, built using Flask in Python, sometimes experiences server connection breaks when fetching data, impacting its reliability. Future work could involve developing a more robust web application using HTML, CSS, and JavaScript, running on a Node.js or AngularJS server. This shift could mitigate the server connectivity issues faced with Flask and allow for a more stable and interactive user experience, enhancing overall usability and responsiveness. Additionally, another potential enhancement would be to add functionality that allows users to click on histogram bars (such as those showing Actual vs. Predicted sentiments or issue histograms from Trust Pilot and Power Reviews). Upon clicking, the relevant customer reviews would be displayed in a tabular format directly on the web page, providing easy access to the underlying data.

8 Conclusion

The project aimed to analyze customer feedback for Wex Photo Video across various review platforms, primarily focusing on Trust Pilot and Power Reviews. By leveraging both traditional supervised Machine Learning (ML) models and advanced BERT-based models, the goal was to predict sentiments behind customer reviews and classify potential issues highlighted in the feedback. Through extensive research, model training, and evaluation, significant insights were gained into the effectiveness of different approaches for sentiment analysis. In terms of sentiment prediction, two supervised ML models—Support Vector Machine (SVM) and Logistic Regression—proved to be the most effective. These models consistently delivered strong performance in categorizing reviews as Positive, Neutral, or Negative. SVM, known for its robust classification capabilities, demonstrated efficiency in handling the often nuanced and varied nature of customer reviews. Similarly, Logistic Regression, a simpler yet powerful algorithm, showed solid results in sentiment classification.

Alongside these traditional models, a deep learning approach was explored using a lightweight BERT-based model, DistilBERT. As a transformer-based model, DistilBERT is capable of understanding the context and meaning of words in a review, allowing it to outperform traditional ML models in sentiment prediction. During evaluation, DistilBERT showed superior performance in terms of accuracy and other evaluation metrics on both validation and external test data. Its ability to capture the context of the entire review, rather than relying on isolated words or phrases, made it especially effective for sentiment analysis tasks in this domain. However, being a deep learning model, DistilBERT required more computational resources compared to SVM and Logistic Regression, which posed some challenges in terms of scalability and deployment.

In addition to sentiment prediction, the project implemented an issue classification system using the spaCy PhraseMatcher. This system relied on a predefined dictionary of potential issues and phrases to match with customer reviews, effectively categorizing feedback based on common problems identified in the reviews. This approach worked well for predefined issues but required manual updates to the dictionary as new issues emerged. To address this limitation, future work could focus on automating issue prediction using more advanced ML or deep learning techniques, which would reduce the need for manual dictionary updates.

The project also explored generating potential reasons behind identified issues using the LLaMA3 8B model. While this model produced relevant and meaningful reasons, it was relatively slow and consumed significant memory resources on a local machine, impacting the user experience. Future enhancements could involve using faster, more efficient models, such as GPT-based APIs, to reduce response times and improve scalability as mentioned in section 7.

In summary, all the objectives outlined at the beginning of the project were successfully achieved. Sentiment analysis was performed using a combination of ML and deep learning models, with DistilBERT showing the best performance. Issue classification and reason generation were also implemented, with room for further improvements. The future work and enhancements identified in Section 7 provide a clear roadmap for building upon the successes of this project, improving performance, and expanding functionality with required resources. By addressing these areas, the system can be made more robust, scalable, and user-friendly, ultimately providing Wex Photo Video with deeper insights into their customer feedback and enabling them to make more informed business decisions.

9 Appendix

9.1 Cross Validations

9.1.1 Cross Validation Approach 1

- Train/Val Split Ratio: 80 (Train) / 20 (Validate)
- Random Seed: 42, 101, 789, 1001, 2023
- Epoch: 4
- Batch Size: 16
- Learning Rate: 5e-5
- Dropout Rate: 0.1
- Token Length: 256

Total Reviews: 20,208

Train Data Distribution

Sentiment Label	No. of Reviews
0	5389
1	5389
2	5388

Table 51: Sentiment Distribution / Rating

Validation Data Distribution

Sentiment Label	No. of Reviews
0	1347
1	1347
2	1348

Table 52: Sentiment Distribution / Rating

Confusion Matrix Per Seed

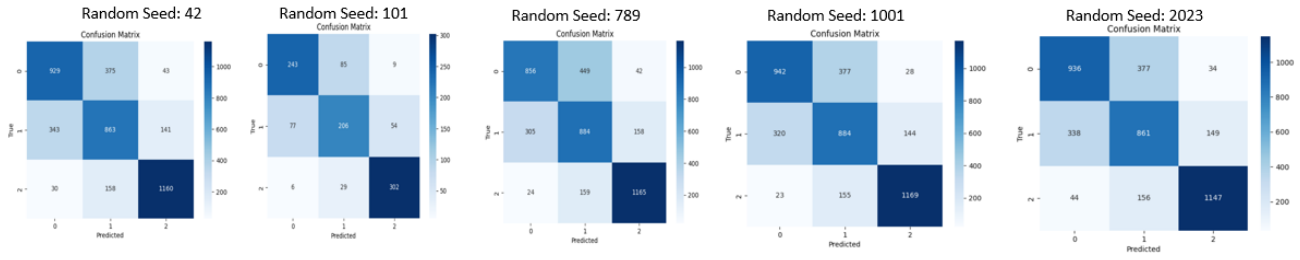

Figure 35: Train Accuracy & Val Loss

Train, Validation Accuracy Per Seed

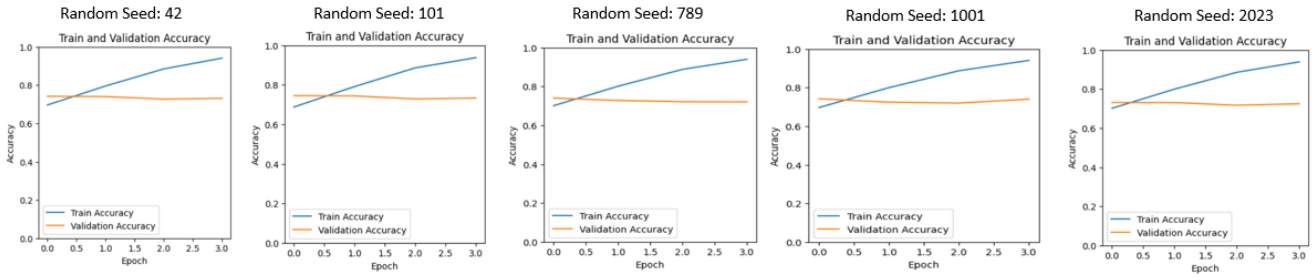

Figure 36: Train Accuracy & Val Loss

Train, Validation Loss Per Seed

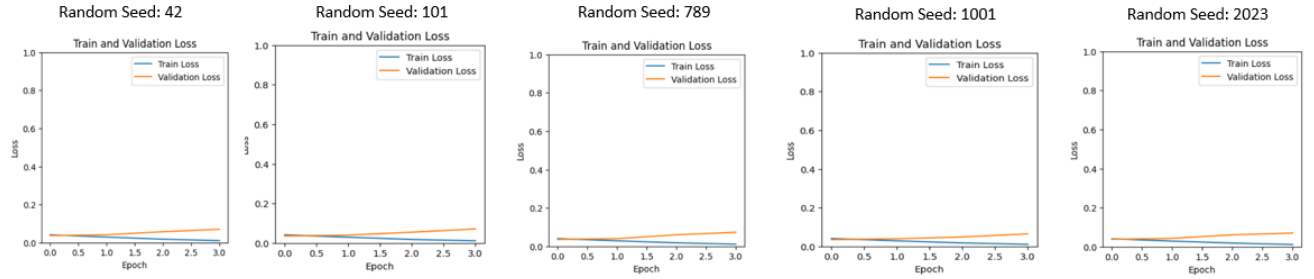

Figure 37: Train Accuracy & Val Loss

Train, Validation Mean Accuracy & Standard Deviation Across Different Seeds

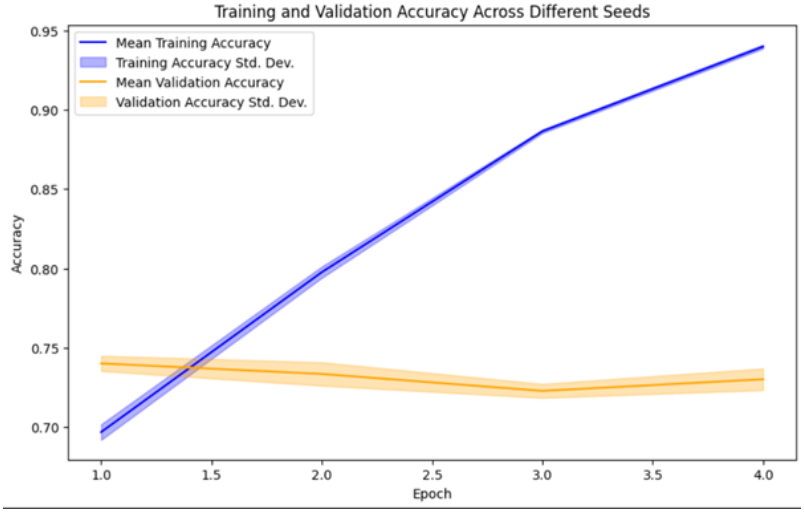

Figure 38: Train Accuracy & Val Loss

Train, Validation Mean Loss & Standard Deviation Across Different Seeds

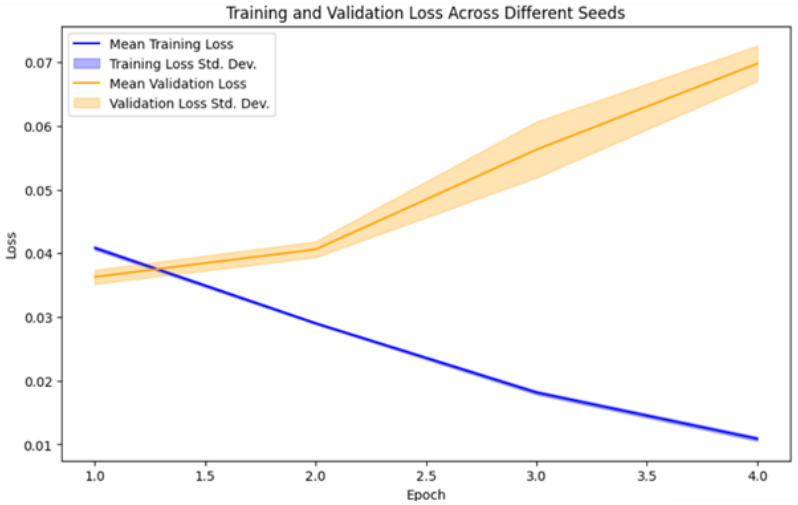

Figure 39: Train Accuracy & Val Loss

Quick Summary

- Mean Training Accuracy: 83.25% | Mean Validation Accuracy: 73.16%
- Mean Training Loss: 0.0247 | Mean Validation Loss: 0.203
- The training accuracy increases with every epoch
- The validation accuracy decreases with every epoch upto last 4th epoch
- Similarly, the training loss decreases per epoch and validation loss increases
- The mean and standard variation plots shows potential over-fitting [25] of the model when trained on given set of configurations across different random states

9.1.2 Cross Validation Approach 2

- **Train/Val Split Ratio: 70 (Train) / 30 (Validate)**
- Random Seed: 42, 101, 789, 1001, 2023
- Epoch: 4
- **Batch Size: 32**
- **Learning Rate: 2e-5**
- **Dropout Rate: 0.2**
- Token Length: 256

Total Reviews: 20,208

Train Data Distribution

Sentiment Label	No. of Reviews
0	4715
1	4715
2	4715

Table 53: Sentiment Distribution / Rating

Validation Data Distribution

Sentiment Label	No. of Reviews
0	2021
1	2021
2	2021

Table 54: Sentiment Distribution / Rating

Confusion Matrix Per Seed

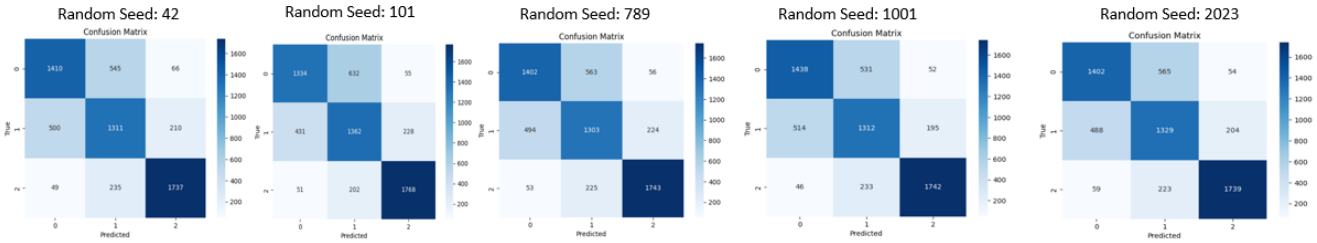

Figure 40: Train Accuracy & Val Loss

Train, Validation Accuracy Per Seed

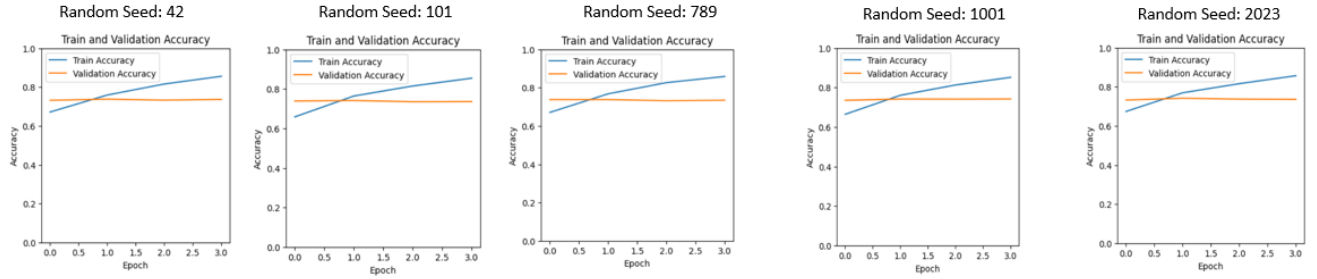

Figure 41: Train Accuracy & Val Loss

Train, Validation Loss Per Seed

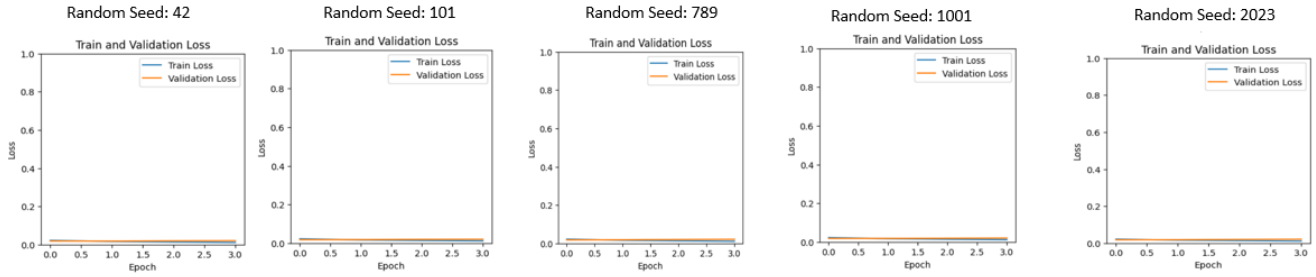

Figure 42: Train Accuracy & Val Loss

Train, Validation Mean Accuracy & Standard Deviation Across Different Seeds

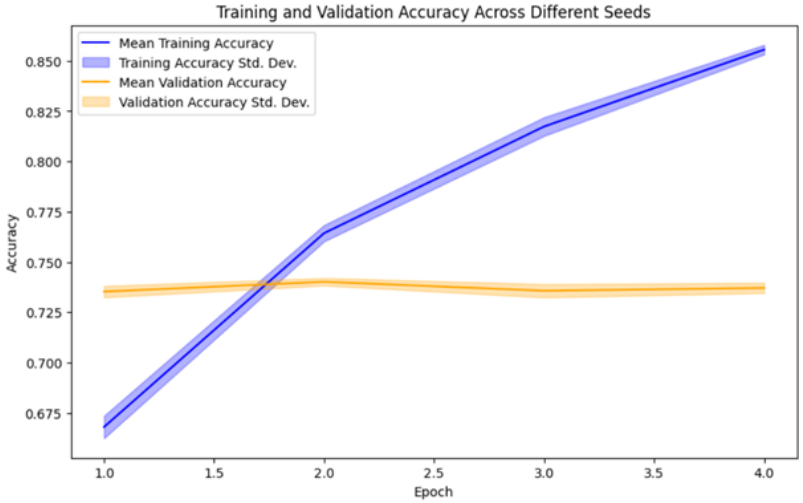

Figure 43: Train Accuracy & Val Loss

Train, Validation Mean Loss & Standard Deviation Across Different Seeds

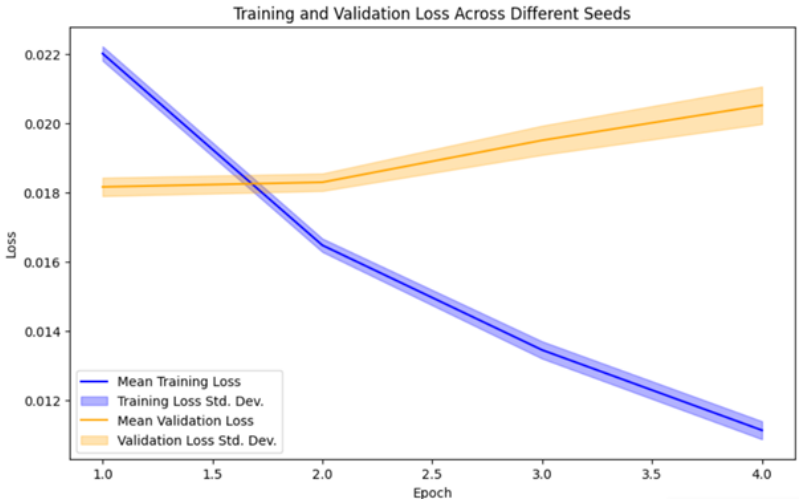

Figure 44: Train Accuracy & Val Loss

Quick Summary

- The batch size was increased to 32 from 16
- Learning rate was reduced from 5e-5 to 2e-5
- Dropout rate was increased from 0.1 to 0.2
- Split ratio was changed from 80/20 to 70/30. More validation data was introduced to expose the model to more unseen data during validation
- Mean Training Accuracy: 77.63% | Mean Validation Accuracy: 73.71%
- Mean Training Loss: 0.015 | Mean Validation Loss: 0.019
- The validation accuracy only improved by roughly 0.7% comparatively to fig. 38

- The mean validation loss decreased by approximately 200% as compared to approach 1 shown in fig. 39

9.1.3 Cross Validation Approach 3

- **Train/Val Split Ratio: 60 (Train) / 40 (Validate)**
- Random Seed: 42, 101, 789, 1001, 2023
- Epoch: 4
- **Batch Size: 64**
- Learning Rate: 2e-5
- **Dropout Rate: 0.3**
- **Token Length: 512**

Total Reviews: 20,208

Train Data Distribution

Sentiment Label	No. of Reviews
0	4042
1	4041
2	4041

Table 55: Sentiment Distribution / Rating

Validation Data Distribution

Sentiment Label	No. of Reviews
0	2694
1	2695
2	2695

Table 56: Sentiment Distribution / Rating

Confusion Matrix Per Seed

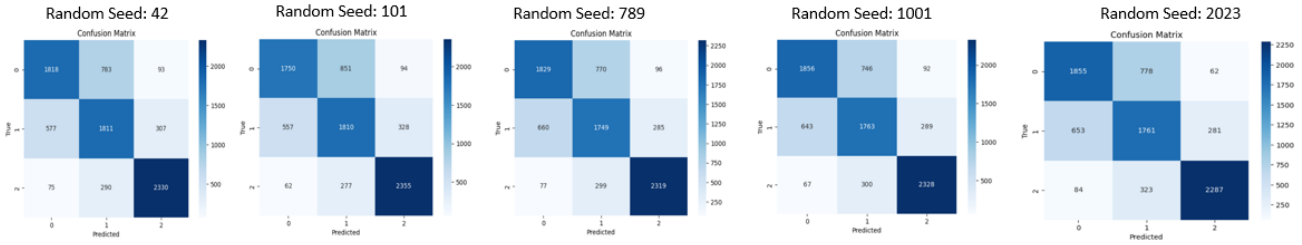

Figure 45: Train Accuracy & Val Loss

Train, Validation Accuracy Per Seed

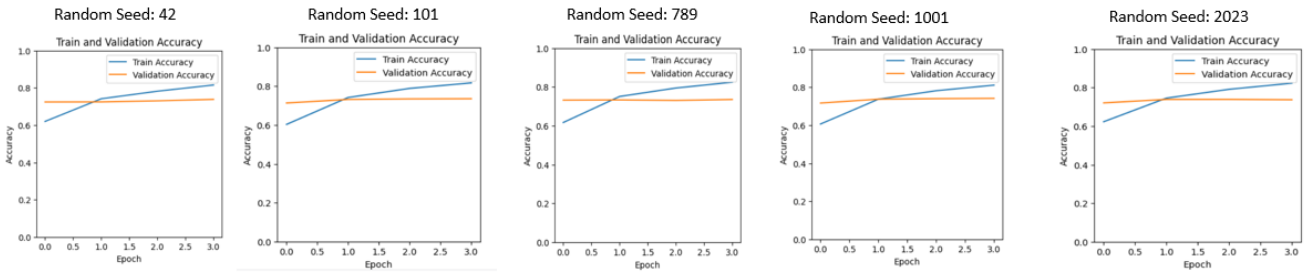

Figure 46: Train Accuracy & Val Loss

Train, Validation Loss Per Seed

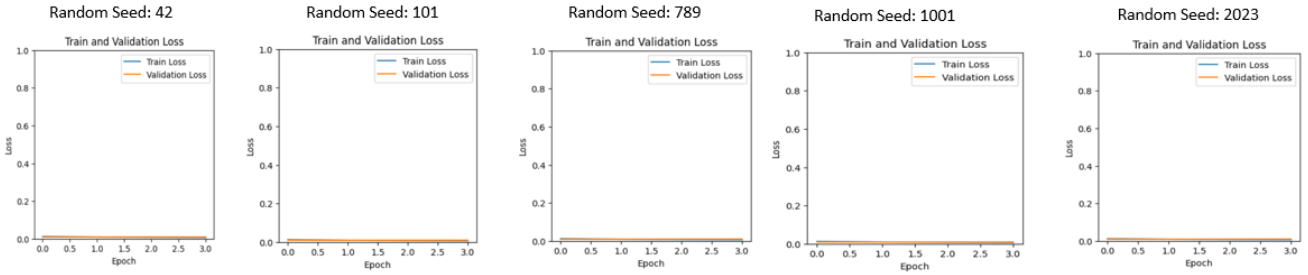

Figure 47: Train Accuracy & Val Loss

Train, Validation Mean Accuracy & Standard Deviation Across Different Seeds

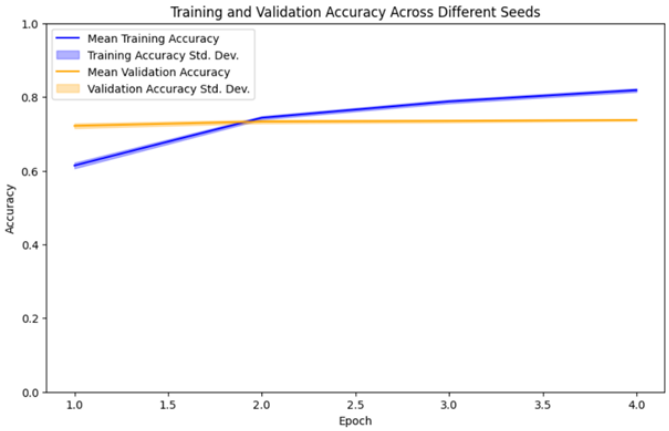

Figure 48: Train Accuracy & Val Loss

Train, Validation Mean Loss & Standard Deviation Across Different Seeds

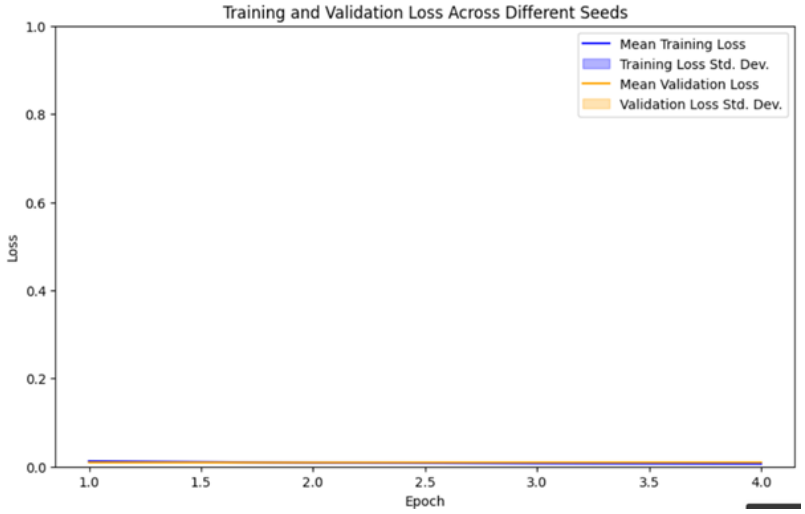

Figure 49: Train Accuracy & Val Loss

Quick Summary

- Changed split size to 60/40 from 70/30 by increasing the validation split ratio even more by 10% to introduce more unseen data to model to validate upon as training has been performing well.
- Batch size increased from 32 to 64
- Dropout rate increased from 0.2 to 0.3
- Token length increased from 256 to 512
- Mean Training Accuracy: 74.08% | Mean Validation Accuracy: 75.17%
- Mean Training Loss: 0.008 | Mean Validation Loss: 0.009
- The validation accuracy only improved by roughly 2% comparatively to fig. 43
- The mean validation loss decreased by another 180% as compared to approach 2 shown in fig. 44

9.1.4 Cross Validation Approach 4

- Train/Val Split Ratio: 60 (Train) / 40 (Validate)
- Random Seed: 42, 101, 789, 1001, 2023
- **Epoch: 8**
- Batch Size: 64
- Learning Rate: 2e-5
- Dropout Rate: 0.3
- Token Length: 512

Total Reviews: 20,208

Train Data Distribution

Sentiment Label	No. of Reviews
0	4042
1	4041
2	4041

Table 57: Sentiment Distribution / Rating

Validation Data Distribution

Sentiment Label	No. of Reviews
0	2694
1	2695
2	2695

Table 58: Sentiment Distribution / Rating

Confusion Matrix Per Seed

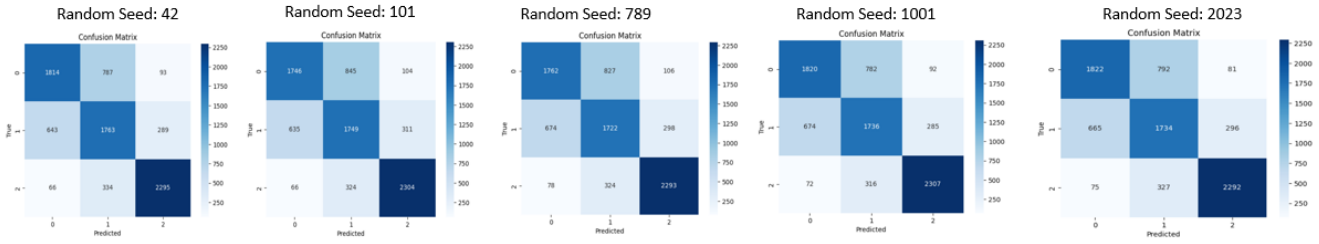

Figure 50: Train Accuracy & Val Loss

Train, Validation Accuracy Per Seed

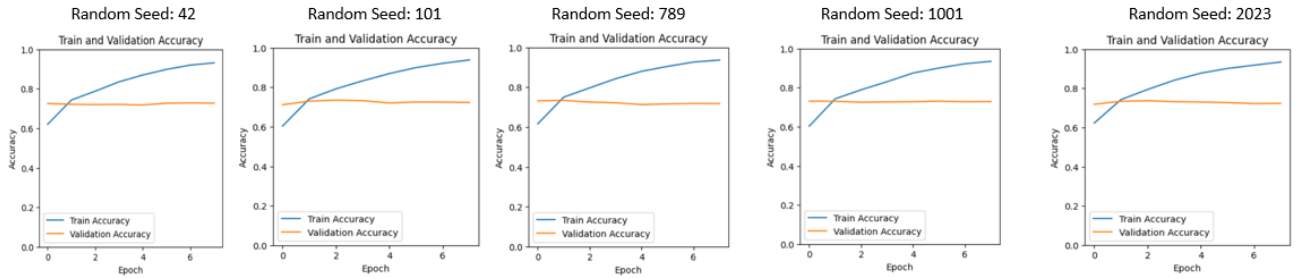

Figure 51: Train Accuracy & Val Loss

Train, Validation Loss Per Seed

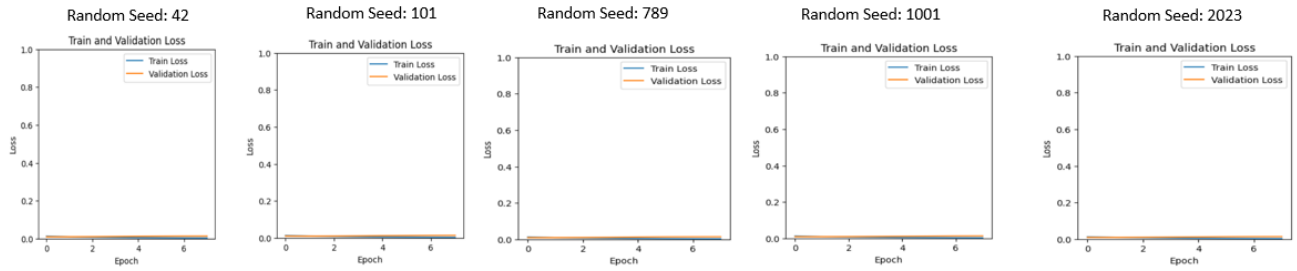

Figure 52: Train Accuracy & Val Loss

Train, Validation Mean Accuracy & Standard Deviation Across Different Seeds

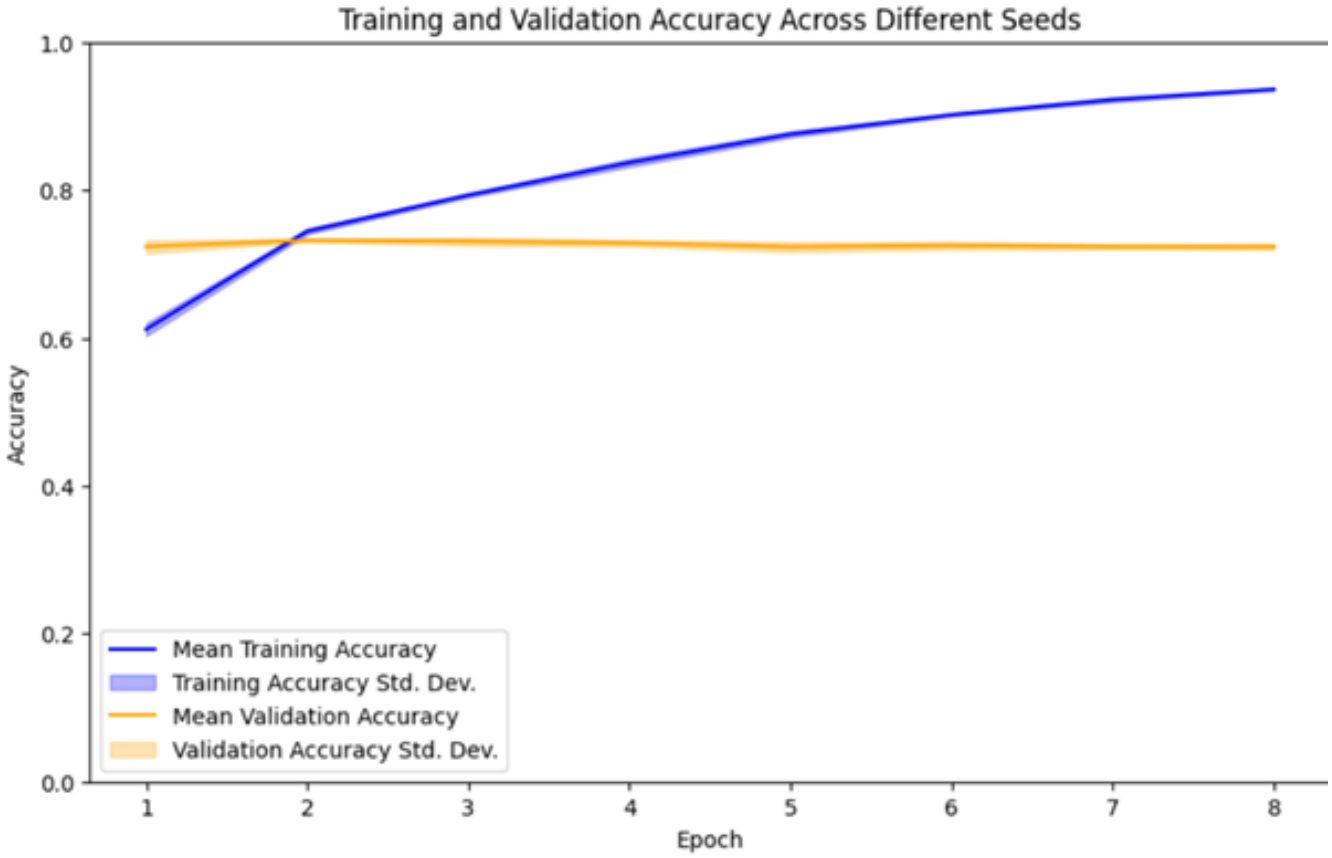

Figure 53: Train Accuracy & Val Loss

Train, Validation Mean Loss & Standard Deviation Across Different Seeds

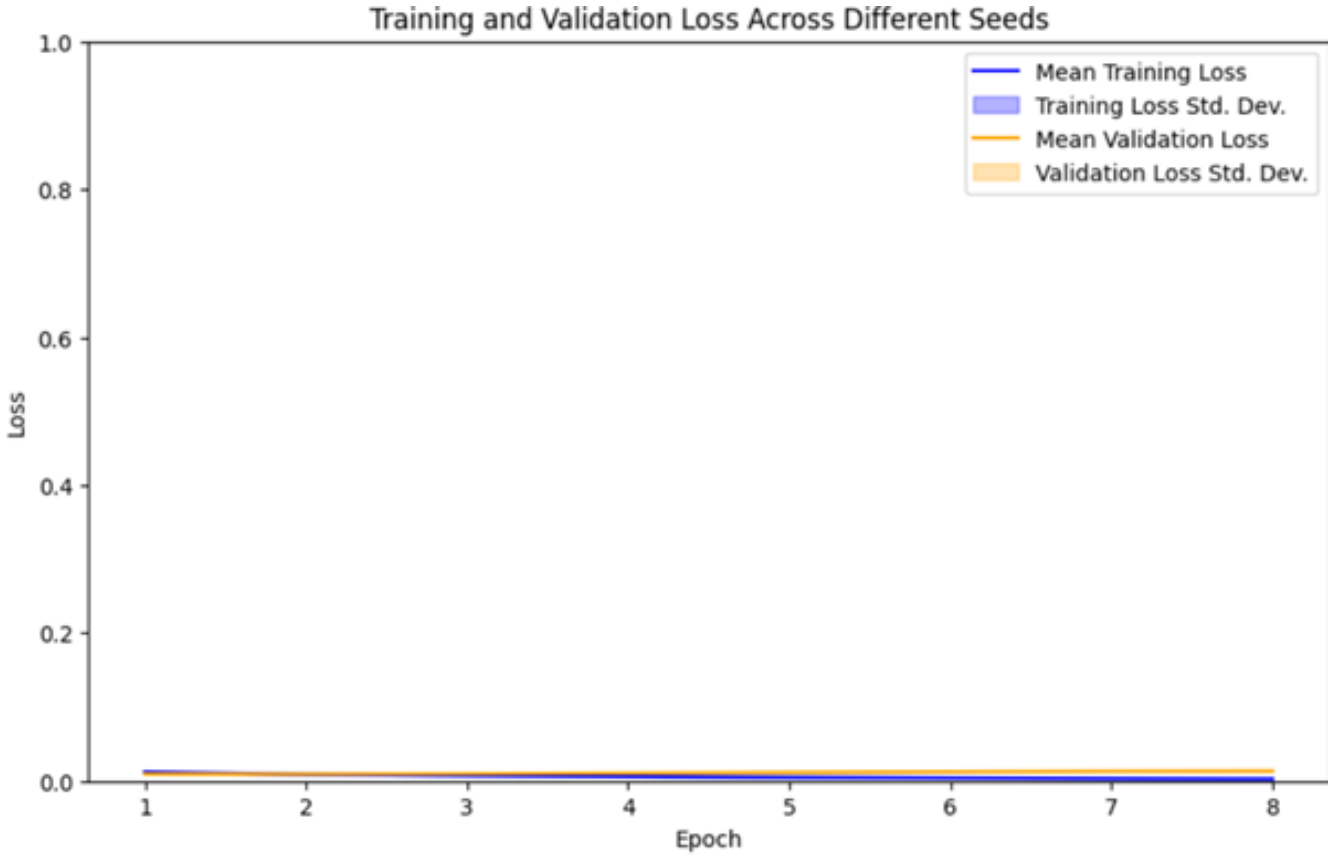

Figure 54: Train Accuracy & Val Loss

Quick Summary

- All the configurations were kept same as defined in approach 3, but only the no. of epoch are increased from 4 to 8 to observe if there was anymore increase and decrease in accuracies and losses respectively
- Mean Training Accuracy: 82.77% | Mean Validation Accuracy: 72.61%
- Mean Training Loss: 0.006 | Mean Validation Loss: 0.011
- There is hardly any improvement in accuracies and losses on increasing no. of epoch from 4 to 8 and leaving rest of the configurations exactly the same as approach 3

9.2 User Experience (UX)

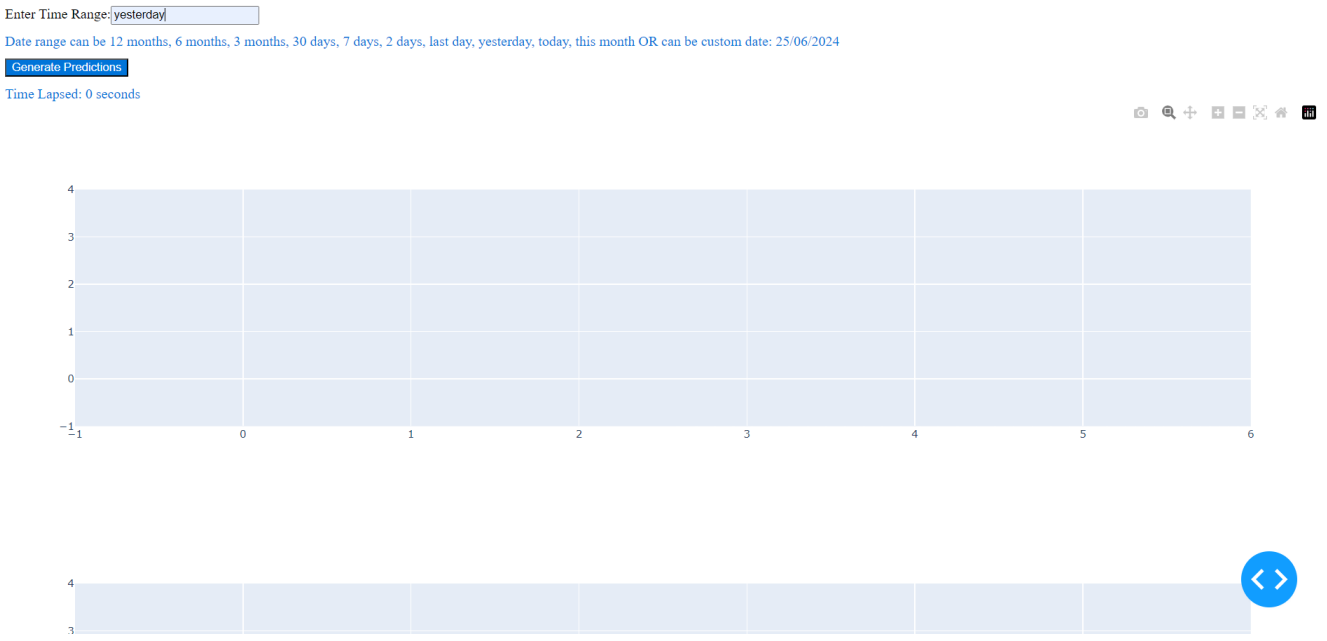

Figure 55: UX - Home Page

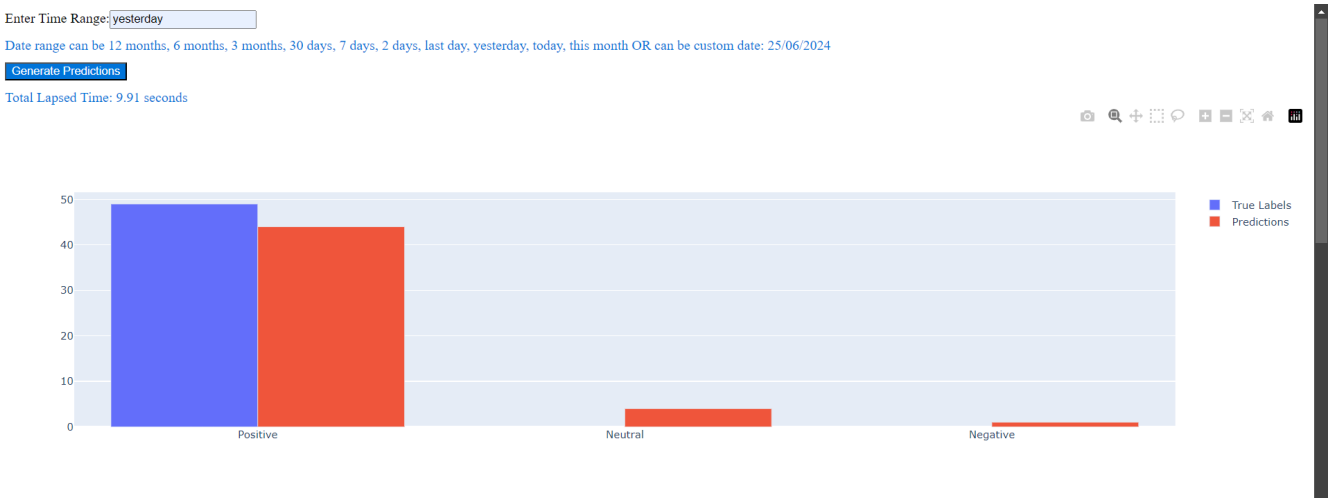

Figure 56: UX - Actual vs Predicted Sentiments

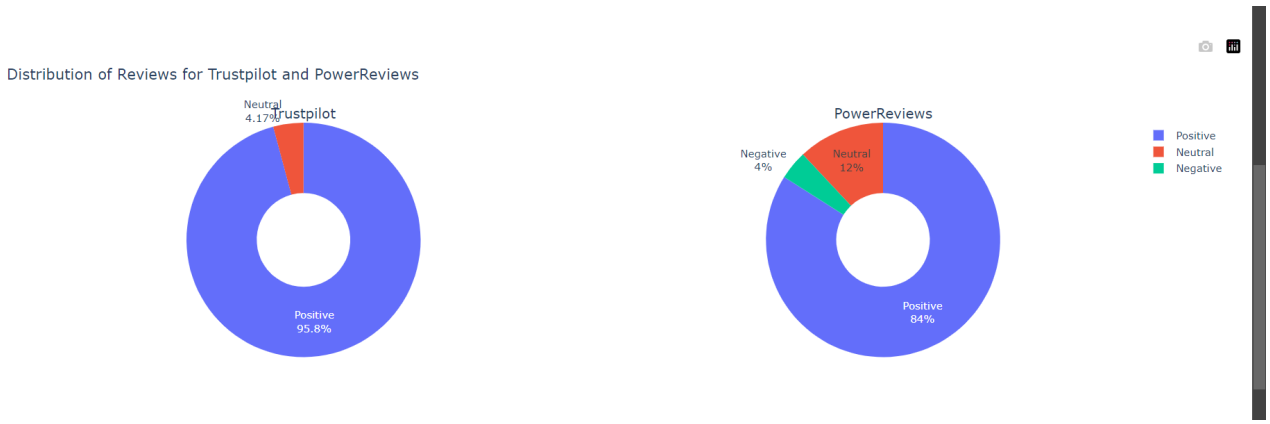

Figure 57: UX - Sentiments Distribution Pie Chart

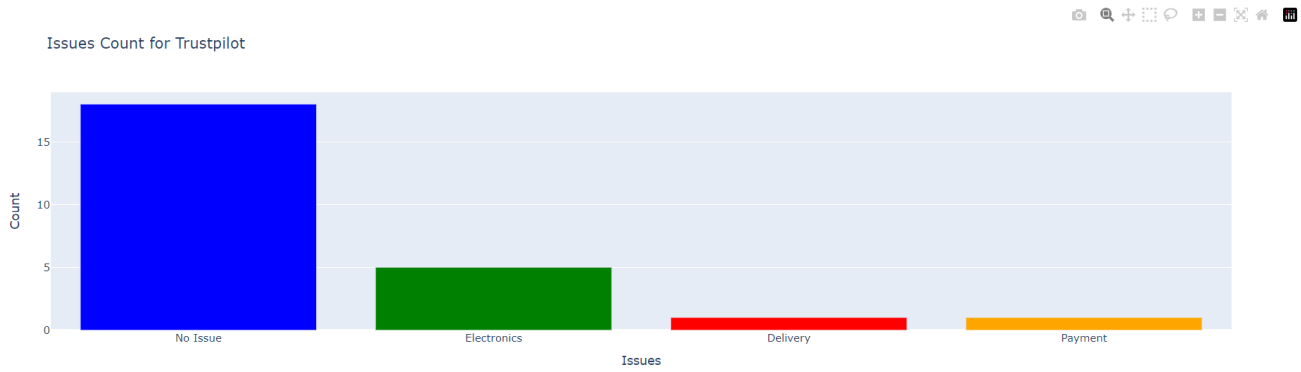

Figure 58: UX - Trust Pilot Issue Distribution

Figure 59: UX - Power Reviews Issue Distribution

9.3 Project Management on Trello

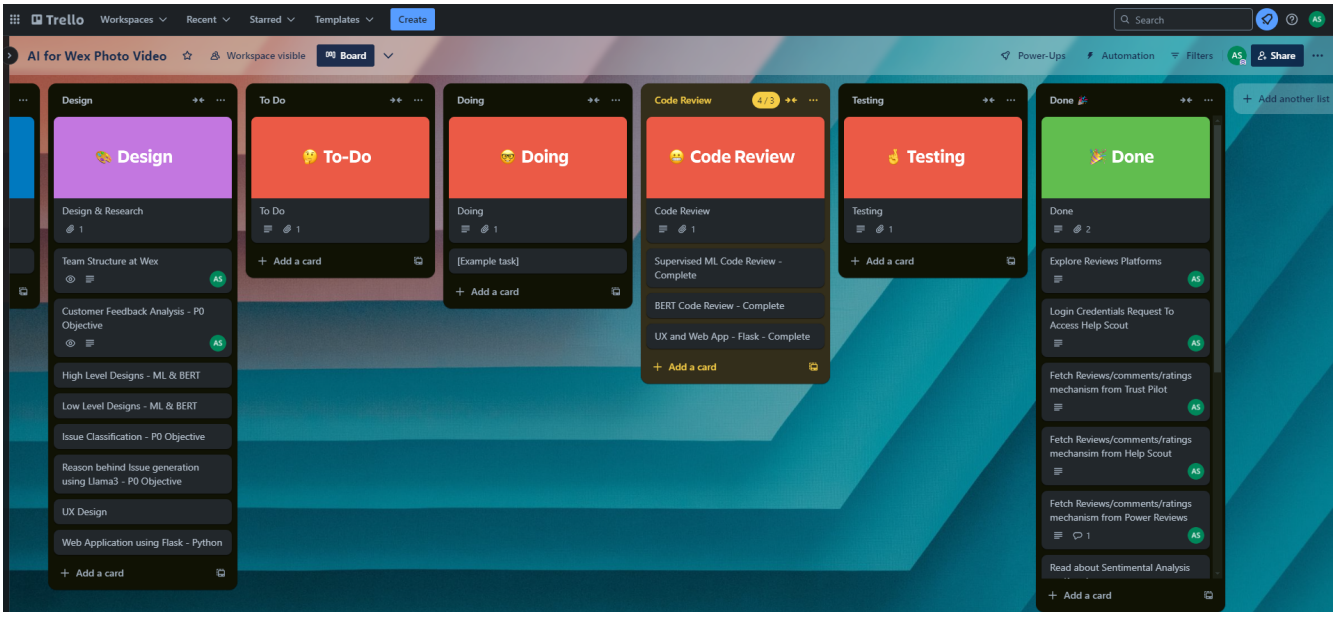

Figure 60: Trello Home Page

9.4 Github Code Tracking & Maintenance

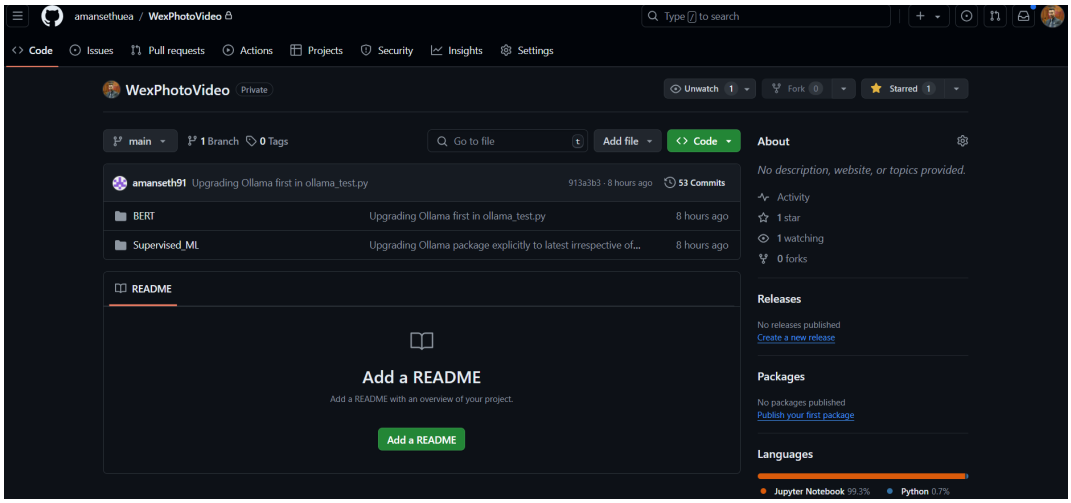

Figure 61: Github - Code Commits

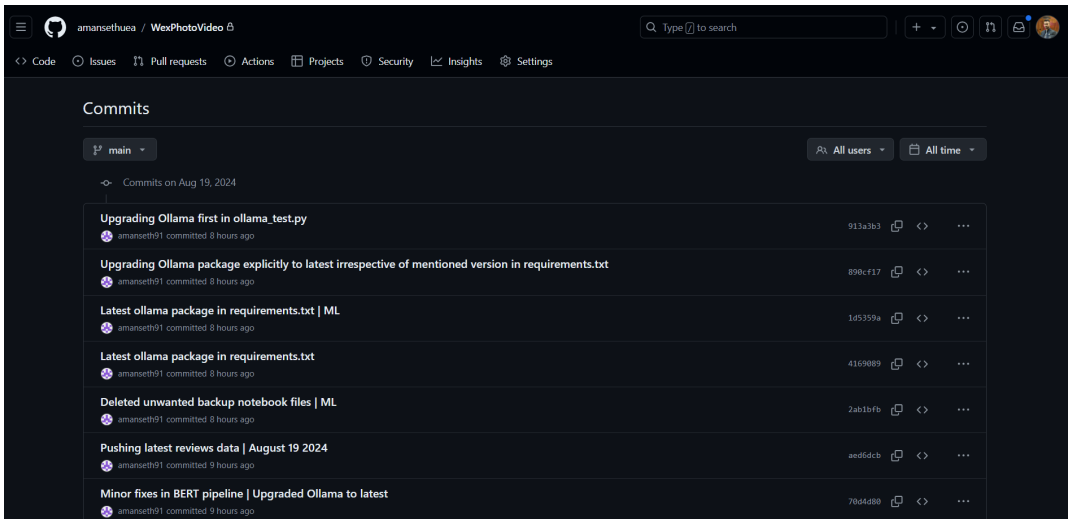

Figure 62: Github - Code Commits

References

[1] Malhar Anjaria and Ram Mohana Reddy Guddeti. A novel sentiment analysis of social networks using supervised learning. *Social Network Analysis and Mining*, 4:1–15, 2014.

[2] Bert language model. `https://www.techtarget.com/searchenterpriseai/definition/BERT-language-model`. Accessed: 2024-05-01.

[3] Bert language model. `https://www.kaggle.com/code/parulpandey/eda-and-preprocessing-for-bert#2.-General-EDA`. Accessed: 2024-05-01.

[4] Krishna Karoo and Mr Vikas Chitte. Ethical considerations in sentiment analysis: Navigating the complex landscape. *International Research Journal of Modernization in Engineering Technology and Science*, 2023.

[5] Yongfeng Ma. A study of ethical issues in natural language processing with artificial intelligence. *Journal of Computer Science and Technology Studies*, 5(1):52–56, 2023.

[6] Jochen L Leidner and Vassilis Plachouras. Ethical by design: Ethics best practices for natural language processing. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 30–40, 2017.

[7] Brendan O'Connor, Ramnath Balasubramanyan, Bryan Routledge, and Noah Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the international AAAI conference on web and social media*, volume 4, pages 122–129, 2010.

[8] B. K. Bhavitha, Anisha P. Rodrigues, and Niranjan N. Chiplunkar. Comparative study of machine learning techniques in sentimental analysis. In *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 216–221, 2017.

[9] Waqar Muhammad, Maria Mushtaq, Khurum Nazir Junejo, and Muhammad Yaseen Khan. Sentiment analysis of product reviews in the absence of labelled data using supervised learning approaches. *Malaysian Journal of Computer Science*, 33(2):118–132, 2020.

[10] Nusrat Jahan Prottasha, Abdullah As Sami, Md Kowsher, Saydul Akbar Murad, Anupam Kumar Bairagi, Mehedi Masud, and Mohammed Baz. Transfer learning for sentiment analysis using bert based supervised fine-tuning. *Sensors*, 22(11):4157, 2022.

[11] Bert language model. `https://huggingface.co/sagorsarker/bangla-bert-base`. Accessed: 2024-05-01.

[12] B Selvakumar and B Lakshmanan. Sentimental analysis on user's reviews using bert. *Materials Today: Proceedings*, 62:4931–4935, 2022.

[13] Bert language model. `https://www.analyticsvidhya.com/blog/2021/06/amazon-product-review-sentiment-analysis-using-bert/`.

[14] Konstantinos I Roumeliotis, Nikolaos D Tselikas, and Dimitrios K Nasiopoulos. Llms in e-commerce: a comparative analysis of gpt and llama models in product review evaluation. *Natural Language Processing Journal*, 6:100056, 2024.

[15] Transformers - hugging face. `https://huggingface.co/docs/hub/transformers`.

[16] Sentiment analysis - company reviews. `https://www.kaggle.com/competitions/sentiment-analysis-company-reviews/rules`.

[17] Women's e-commerce clothing reviews. https://www.kaggle.com/datasets/nicapotato/womens-ecommerce-clothing-reviews.

[18] Train test validation split. https://www.v7labs.com/blog/train-validation-test-set.

[19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[20] Ashish Vaswani. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[21] spacy phrasematcher. https://spacy.io/api/phrasematcher.

[22] Open ai. https://openai.com/api/.

[23] Llama2 huggingface. https://huggingface.co/meta-llama/Llama-2-7b-hf.

[24] Llama3 - ollama. https://ollama.com/.

[25] What is overfitting? https://www.ibm.com/topics/overfitting.