2545 ASPECT-BASED SENTIMENT ANALYSIS USING BERT AND DEEP LEARNING MODEL

1201101003 CHIA YU ZHANG

BACHELOR DEGREE IN COMPUTER SCIENCE
B.C.S (HONS) DATA SCIENCE

FACULTY OF COMPUTING AND INFORMATICS
MULTIMEDIA UNIVERSITY

JUNE 2024

2545 ASPECT-BASED SENTIMENT ANALYSIS USING BERT AND DEEP LEARNING MODEL

BY

1201101003 CHIA YU ZHANG

PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE

REQUIREMENT FOR THE DEGREE OF

BACHELOR OF COMPUTER SCIENCE B.C.S (HONS) DATA SCIENCE

in the

Faculty of Computing and Informatics

MULTIMEDIA UNIVERSITY

MALAYSIA

JUNE 2024

DECLARATION

I hereby declare that the work has been done by myself and no portion of the work contained in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institution of learning.

Chia Yu Zhang

Name of candidate: CHIA YU ZHANG

Faculty of Computing & Informatics

Multimedia University

Date: 28: 06 :2024

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my Final Year Project (FYP) supervisor, Dr. Ng Hu for providing me an opportunity to perform the research project and guiding me throughout the entire research process.

Last but not least, sincere thanks to both my parents, family members and friends that has been my back-bone and support me throughout my research project.

Abstract

This research aimed to evaluate the effectiveness of integrating subjective and objective classification into the analysis of textual data. The research was divided into three main phases: dataset collection, subjectivity classification, and comparative analysis of labelling approaches. Initially, datasets tailored for subjectivity analysis, including the SemEval 2014 and MEMD datasets, were compiled. Subsequently, subjectivity classification techniques were applied to these datasets using advanced models like BERT to distinguish subjective and objective elements in the textual data. In the second phase, different labelling approaches were used to classify the MEMD dataset, followed by training the datasets with the BERT model for sequence classification. The results were then compared to evaluate the effectiveness of each labelling approach. Through these steps, this study aimed to refine the classification process, providing valuable insights into the dynamics of subjectivity and objectivity within user-generated content.

Keywords: subjectivity classification, BERT, Different labelling approaches, MEMD dataset, SemEval 2014 dataset.

Table of Contents

List of Figures

List of Tables

1.0 Introduction

1.1 Project Overview

In the project, natural language processing (NLP) involved using computer methods to understand and analyse human language. The primary focus was on subjectivity classification in user reviews to distinguish between subjective and objective elements. BERT (Bidirectional Encoder Representations from Transformers), an advanced NLP model known for considering both left and right context simultaneously, was employed, enhancing its ability to understand language intricacies.

In this context, 'subjective' content reflected personal opinions, feelings, and experiences, while 'objective' or 'non-subjective' content maintained a neutral, factual tone without personal sentiments. To improve the accuracy of subjectivity classification, different labelling approaches were explored. The introduction of a 'subjectivity' attribute in the dataset enabled the model to distinguish between these elements, contributing to more precise classification.

The project delved into the intricate interplay between subjectivity and objectivity in user reviews, aiming to refine the classification process. Understanding subtle degrees of subjectivity and objectivity in textual data is challenging, especially with human-generated content. Reviews are inherently influenced by personal experiences and emotions, making the distinction between subjective and objective content crucial. Through subjectivity-objectivity classification, using advanced models like BERT and deep learning, the project sought to elevate classification accuracy.

1.2 Project Statement

In this project, the primary focus was on subjectivity analysis. The initial phase involved gathering datasets tailored for the classification of subjective and objective (non-subjective) content in user-generated reviews. The aim was to accurately identify subjective and objective parts in the text, providing a basis for further analysis.

The project then applied subjectivity classification to these datasets using different labelling approaches. Advanced models like BERT were employed for sequence classification, with the introduction of a 'subjectivity' attribute to enhance the model's ability to distinguish between subjective and objective elements.

By training the datasets with the subjectivity classification model (BERT) and comparing the results, the project sought to refine the classification process. This approach provided valuable insights into the dynamics of subjectivity and objectivity within user-generated content, contributing to more accurate and nuanced text analysis.

1.3 Project Objectives

In this research project is expected to achieve several objectives. The objectives highlighted are as follows:

i) To collect aspect-based sentiment analysis dataset and subjectivity analysis dataset.

ii) To perform subjectivity classification on aspect-based sentiment analysis dataset.

iii) To utilize different labelling approaches to label the aspect-based sentiment analysis datasets and compare the performance metric across different labelling approaches.

1.4 Project Scope

This project is situated within the expansive domain of Natural Language Processing (NLP), concentrating on subjectivity analysis rather than sentiment analysis. Initially, datasets tailored for this task, including the SemEval 2014 and MEMD datasets, are meticulously collected. These datasets form the foundation for exploring the complex relationship between subjective and objective elements in user-generated content.

The next step involves applying subjectivity classification to the datasets, with a focus on the MEMD dataset in the second part of the project. Advanced models like BERT are employed for detailed sequence classification. By introducing a 'subjectivity' attribute, the goal is to accurately identify and categorize subjective and objective components.

In the second phase of the project, emphasis is placed on different labelling approaches to classify the MEMD dataset. This involves experimenting with various labelling techniques to enhance the accuracy and effectiveness of the subjectivity classification model. By training the datasets with BERT and comparing the results, the project aims to refine the classification process, providing valuable insights into the dynamics of subjectivity and objectivity within user-generated content.

1.5 Project Plan

Table 1. 1 Gantt chart for FYP part 1

PHASE 1 - Trimester 1 2023/2024														
Description	**Week**													
Activity	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Chapter 1: Introduction														
Chapter 2: Literature Review														
Chapter 3: Theoretical Framework														
Chapter 4: Research Methodology														
Chapter 5: Implementation Plan														
Chapter 6: Conclusion														

Table 1. 2 Gantt chart for FYP part 2

PHASE 2 - Trimester 2 2023/2024														
Description	**Week**													
Activity	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Chapter 4: Enhance Methodology														
Chapter 5: Datasets Labelling Approaches and Performance Evaluation														
Chapter 6: Conclusion														
Report Writing and Submission														
Paper Writing														
Presentation														

2.0 Literature Review

2.1 Aspect-Based Sentiment Analysis (ABSA)

Aspect-Based Sentiment Analysis (ABSA) was focuses on understanding people's opinions in a more detailed way (Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam, 2023). For example, ABSA was composed of four emotional elements, namely aspect terms, aspect categories, opinion terms and emotion polarity (Bing Liu, 2022) .Figure 2.1 shows a clear example of the four fundamental sentiment components.

- aspect terms (a) are the specific feature or aspect of the topic.

- aspect categories (c) refer to what is being talked about. For instant, laptop and mouse is the category that under computer shop domain.

- opinion terms (o) denote the expression of the sentiment. For example, “The quality of the mouse in this shop is good”, the “good” is the opinion in this sentence.

- sentiment polarity (p) indicates whether the sentiment expresses positive, negative, or neutral.

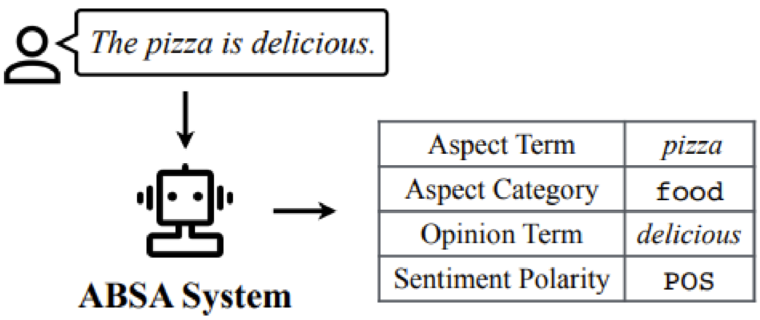


Figure 2. 1. Four fundamental sentiment components (W. Zhang et al., 2023)

The important of Aspect-based Sentiment Analysis (SA) was underscored from both user and enterprise viewpoints. (Mohammad Erfan Mowlaei, Mohammad Saniee Abadeh, Hamidreza Keshavarz., 2020). Prior to making a purchase, reserving accommodations, or engaging with a service, individuals typically want access to various forms of information pertaining to distinct facets. Organisations continually work in the pursuit of identifying the underlying causes of user contentment or discontentment, with the aim of enhancing their products, conducting comprehensive research of these factors can significantly aid companies in this endeavour (Kamal Amarouche, Houda Benbrahim, Ismail Kassou, 2015). Organisations have the ability to leverage customer feedback as a means to modify their sales methods, which may involve implementing price reductions or offering discounts in order to incentivize customers to acquire products that have lower sales volumes.

In contrast to previous ABSA studies that concentrated on a single sentiment element, recent research has delved into compound ABSA tasks, encompassing multiple elements to capture more comprehensive aspect-level sentiment information. ABSA has two main types, single ABSA tasks and compound ABSA tasks. Regarding single ABSA tasks, there are four specific jobs. Aspect Term Extraction (ATE), Aspect Category Detection (ACD), Opinion Term Extraction (OTE) and Aspect Sentiment Classification (ASC).

- Aspect Term Extraction (ATE): ATE involves finding specific aspects like "pizza" or "service". It can be done using different approaches like supervised or unsupervised methods.

- Aspect Category Detection (ACD): ACD identifies the main categories being discussed, using techniques like attention mechanisms.

- Opinion Term Extraction (OTE): OTE focuses on finding opinion expressions related to an aspect, considering the relationship between them.

- Aspect Sentiment Classification (ASC): ASC predicts the sentiment for a specific aspect in a sentence like positive, negative, and neutral.

Moving on to compound ABSA tasks, these involve predicting multiple sentiment elements together. For example, Aspect-Opinion Pair Extraction (AOPE), Aspect Sentiment Triplet Extraction (ASTE) and Aspect Sentiment Quad Prediction (ASQP).

- Aspect-Opinion Pair Extraction (AOPE): AOPE is under pair extraction, and it aims to extract aspects and opinion terms together.

- Aspect Sentiment Triplet Extraction (ASTE): ASPE is under triplet extraction, and it aims to extract aspect, opinion, and sentiment together.

- Aspect Sentiment Quad Prediction (ASQP): ASQP is under quad extraction, and it uses to predict the four elements in the sentence in one time.

These compound tasks are like integrated versions of single ABSA tasks but with a stronger concentrate on understanding the relationships between different elements and has been widely used in different scenario.

Aspect-based sentiment analysis (ABSA) plays a crucial role in understanding sentiments directed towards specific aspects in textual data, presenting a more nuanced approach than traditional sentiment analysis. Recent studies have showcased the effectiveness of utilizing BERT, a state-of-the-art language representation model, in ABSA tasks.

In the paper titled "Aspect-Based Sentiment Analysis Using BERT," the authors emphasize the significance of ABSA in dissecting sentiments towards specific aspects within text (Mickel Hoang, Oskar Alija Bihorac, and Jacobo Rouces., 2019). By leveraging BERT, they achieved state-of-the-art results on SemEval-2015 and 2016 datasets for aspect classification. Their approach demonstrates BERT's prowess in capturing semantic similarities and relations between aspects and textual inputs, particularly excelling in out-of-domain evaluations and displaying potential for generalization across different domains.

Similarly, in "Exploiting BERT for End-to-End Aspect-Based Sentiment Analysis" by (Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam et al., 2019), the authors explore BERT's contextualized embeddings for end-to-end ABSA. Their method, standardized through consistent utilization of a hold-out development dataset (SemEval 2014, 2015, 2016), provides a benchmark for future BERT-based E2E-ABSA research, setting a standard in model selection processes.

In another advancement, "Adversarial Training for Aspect-Based Sentiment Analysis with BERT" introduces the BERT Adversarial Training (BAT) approach (Akbar Karimi, Leonardo Rossi, Andrea Prati., 2020). This technique, conducted in the embedding space, enhances ABSA tasks through adversarial training, making neural networks more resilient and effective. Utilizing datasets from SemEval 2014 and 2016, with a focus on laptops and restaurants, the authors showcase significant improvements in model performance, highlighting the potential of adversarial training in sentiment analysis.

Furthermore, "Utilizing BERT Intermediate Layers for Aspect-Based Sentiment Analysis and Natural Language Inference" explores leveraging BERT's intermediate layers to enhance fine-tuning for ABSA (Youwei Song, Jiahai Wang, Zhiwei Liang, Zhiyue Liu, and Tao Jiang., 2020). Testing on three ABSA datasets (Laptop, Restaurant, Twitter) and the Stanford Natural Language Inference (SNLI) dataset, the study demonstrates that these pooling strategies improve BERT-based model performance, effectively capturing aspect-based sentiments in text.

In "Enhancing Attention-Based LSTM With Position Context for Aspect-Level Sentiment Classification," PosATT-LSTM is introduced (Jiangfeng Zeng, Xiao Ma, and Ke Zhou., 2019). This innovative attention-based LSTM model incorporates position-aware vectors to represent explicit position context between aspects and their contextual words. Outperforming traditional LSTM models on SemEval 2014 datasets for aspect-level sentiment classification, PosATT-LSTM achieves significant accuracy improvements for both Laptop and Restaurant datasets.

Moreover, "Dual Graph Convolutional Networks for Aspect-Based Sentiment Analysis" focuses on improving ABSA using DualGCN (Ruifan Li, Hao Chen, Fangxiang Feng, Zhangyu Ma, Xiaojie WANG, and Eduard Hovy., 2021). Tested on SemEval 2014 restaurant reviews, SemEval 2014 laptop reviews, and tweets, DualGCN outperforms existing methods, particularly excelling in understanding complex and informal text, presenting a promising tool for sentiment analysis.

In "Embedding Refinement Framework for Targeted Aspect-Based Sentiment Analysis," a new method, Refining Affective Embedding from Context (RAEC), is introduced to improve sentiment understanding related to specific aspects in text (Bin Liang, Rongdi Yin, Jiachen, Lin Gui, Yulan He, Min Yang, Ruifeng Xu., 2023). Tested on SentiHood and SemEval 2015 datasets, the results demonstrate RAEC's superiority over similar methods, particularly in understanding emotions associated with specific subjects in the text.

In, "Modelling Context and Syntactical Features for Aspect-Based Sentiment Analysis" addresses challenges in understanding opinions on specific topics or aspects in text (Minh Hieu Phan & Philip Ogunbona, 2020). The proposed method, combining part-of-speech embedding, dependency embedding, and contextualized embedding, enhances aspect extractor performance, particularly demonstrated on SemEval 2014 laptop and restaurant reviews datasets, showcasing significant advancements in understanding sentence structures for sentiment analysis.

Lastly, "Towards Generative Aspect-Based Sentiment Analysis," a novel approach to understanding sentiments in text related to specific aspects is explored (Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam., 2021). Approaching sentiment analysis as a text generation challenge, this research outperforms existing methods on SemEval challenges, highlighting the effectiveness of generating text capturing nuanced sentiments related to specific aspects.

As aspect-based sentiment analysis expanded, the corpus gained increased importance for model building and training. (W. Zhang et al., 2023), has provide an overview of common ABSA benchmark datasets. However, according to the research paper above, SemEval (2014, 2015, 2016), Twitter, Stanford Natural Language Inference (SNLI) and SentiHood dataset has been used to test the accuracy of ABSA but only SemEval dataset has been widely used in all research paper. SemEval (2014, 2015, 2016) is widely employed as a benchmark in aspect-based sentiment analysis (ABSA) that combining user reviews from domains such as laptops and restaurants with annotations of aspect categories, aspect terms, and sentiment polarities. Table 2.1 show about the detail for the dataset has been use in research paper above.

Table 2. 1. Dataset Overview (a)

Dataset	Research Paper	URL
SemEval 2014 (Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos., 2014,)	(Karimi et al., 2020; R. Li et al., 2021; X. Li et al., 2019; Phan & Ogunbona, 2020; Y. Song et al., 2020; Zeng et al., 2019; W. Zhang et al., 2021)	https://alt.qcri.org/semeval2014/task4/ https://github.com/avinashsai/ BERT-Aspect/tree/master
SemEval 2015 (Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos., 2015)	(Hoang et al., 2019; X. Li et al., 2019; Liang et al., 2023; W. Zhang et al., 2021)	https://alt.qcri.org/semeval2015/task12/
SemEval 2016 (Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos., 2016)	(Hoang et al., 2019; Karimi et al., 2020; X. Li et al., 2019; W. Zhang et al., 2021)	https://alt.qcri.org/semeval2016/task5/

Table 2. 2. Dataset Overview (b)

Dataset	Research Paper	URL
Twitter (Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, Ke Xu., 2014a)	(R. Li et al., 2021; Y. Song et al., 2020)	https://github.com/napsternxg/ SemEval_Twitter_Data
SentiHood (Marzieh Saeidi, Guillaume Bouchard, Maria, Liakata. Sebastian Riedel., 2016)	(Liang et al., 2023)	https://huggingface.co/datasets /bhavnicksm/sentihood
Stanford Natural Language Inference (SNLI) (Samuel R. Bowman, Gabor Angeli, Christopher, Potts, Christopher D. Mannning., 2015)	(Y. Song et al., 2020)	https://www.kaggle.com/datasets/ stanfordu/stanford-natural-language-inference-corpus

2.2 Lexicons-Based

The lexicon-based approach is a conventional technique employed in the realm of Aspect-Based Sentiment Analysis (ABSA). The lexicon-based approach initially established connections between aspects mentioned in a sentences and their corresponding words or phrases (Haoyue Liu, Ishani Chatterjee, MengChu Zhou, XiaoYu Sean Lu, Adbullah Abusorrah., 2020). Subsequently, it determines the sentiment polarity of each aspect by examining the sentiment polarity of individual words or phrases in the lexicon.

The lexicon-based method employed in the study conducted by Tarbiat Modares University involves the use of two lexicons, namely ABFBSA Lexicon and ABALGA lexicon (Mowlaei et al., 2020). These lexicons are dynamically created and particularly adapted for the purpose of aspect-based sentiment analysis (SA). These lexicons differentiate themselves from traditional ones by demonstrating adaptability to evolving circumstances and incorporating user feedback. Consequently, they possess the capability to classify opinions inside text according to distinct elements, such as the features or attributes of a product.

2.2.1 ABFBSA Lexicon Generation

Aspect-Based Frequency Based Sentiment Analysis (ABFBSA) extends the FBSA approach to aspect-based sentiment analysis (Hamidreza Keshavarz & Mohammad Saniee Abadeh, 2017a). Unlike FBSA, which was designed for general tweet-level analysis, ABFBSA focuses on aspect-level sentiment by considering the context of each aspect within a sentence. This method tracks the frequency of words in proximity to positive and negative aspects, assigning scores based on these frequencies. It employs techniques like lemmatization and the removal of stopwords to improve precision (Mowlaei et al., 2020).

2.2.2 ABALGA lexicon generation

The ABALGA method is an extension of the Frequency-Based Sentiment Analysis (FBSA) algorithm, designed originally for tweet-level opinion mining (Mohammad Erfan Mowlaei, Mohammad Saniee Abadeh, Hamidreza Keshavarz., 2018). This method adapts FBSA for aspect-based opinion mining by modifying how word scores are calculated and by adding new features suitable for aspect-level analysis. In this context, words are associated with the polarity of aspects in customer reviews, considering both sentiment terms and aspects. For instance, in a review saying, "This laptop has a high screen resolution but the keyboard is hard to deal with," the words "high" and "hard to deal with" express sentiments about the aspects "screen resolution" and "keyboard," respectively.

2.2.3 Spectral-Clustering-Based Topic-Specific Chinese Sentiment Lexicon (STCS)

Zhang research team has introduced a new sentiment lexicon called Spectral-Clustering-Based Topic-Specific Chinese Sentiment Lexicon (STCS) (Bo Zhang, Duo Xu, Huan Zhang and Meizi Li., 2019). The STCS lexicon is comprised of three models: the Filtering Text Model (FT Model), the Constructing Sentiment Relationship Graph Model (CRM Model), and the Spectral Clustering Model (SC Model). Figure 2.2 shows an example of the STCS lexicon Framework.

- FT model: The FT model filters out irrelevant comments.
- CRM model: the CRM model calculates base sentiment similarity, topic sentiment similarity.
- SC model: SC model clusters sentiment words based on this graph, generating a lexicon that addresses topic-specific sentiments.

This approach was tested using data from JD.com and eLong on topics like "iPhone6" and "hotel," proving effective in dealing with topic-related sentiment words and improving lexicon accuracy.

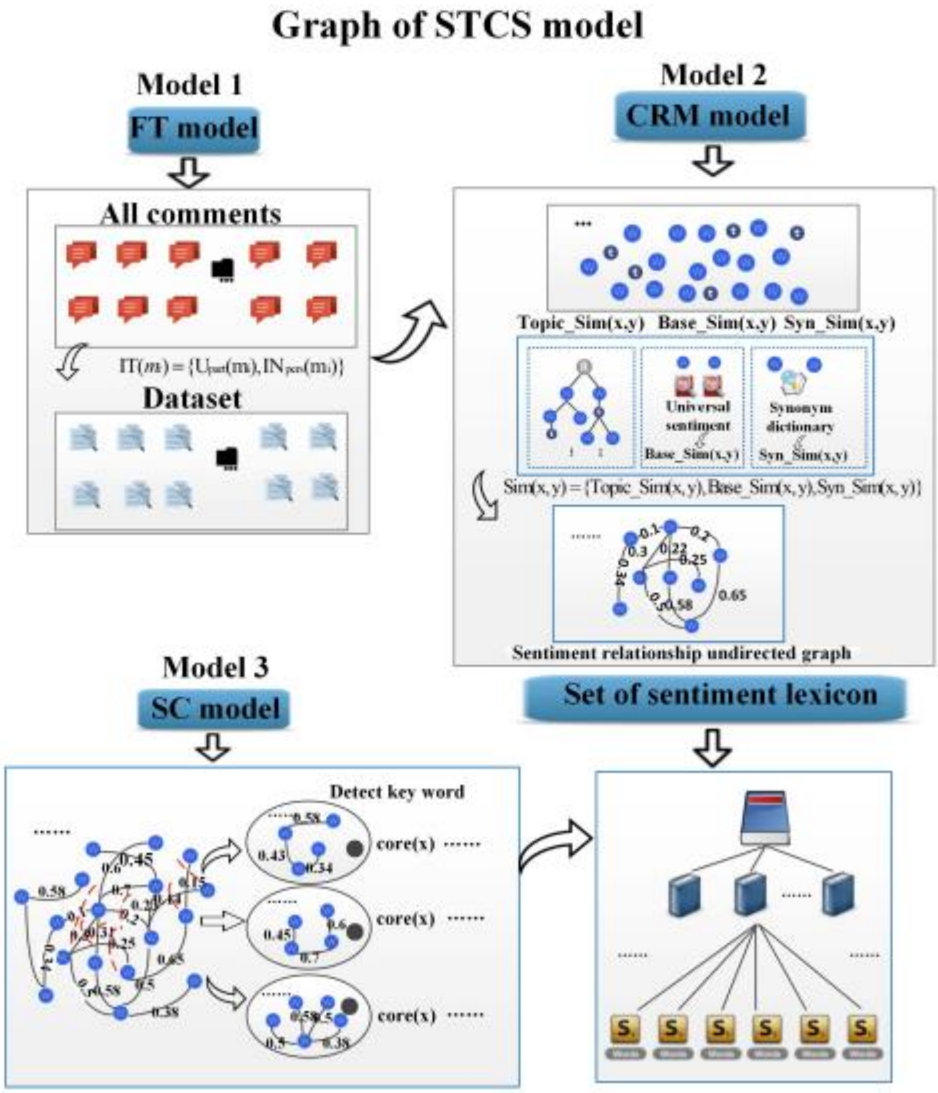


Figure 2. 2. STCS Lexicon Framework (B. Zhang et al., 2019)

2.3 Deep Learning

Deep learning is a specialized area within machine learning that concentrates on crafting and refining artificial neural networks to carry out tasks without explicit programming. These networks, inspired by the way the human brain operates, are composed of layers of interconnected nodes or neurons. The term "deep" signifies the incorporation of multiple layers in these networks, enabling them to autonomously learn hierarchical representations of data. Deep learning has achieved impressive results in tasks such as recognizing images and speech, processing natural language, and excelling at games.

2.3.1 Convolutional neural networks (CNN)

Research from Zhejiang University of Technology elucidated the role of Convolutional Neural Networks (CNN) in Aspect-Based Sentiment Analysis (ABSA) (Haoyue Liu, Ishani Chatterjee, MengChu Zhou, XiaoYu, Sean Lu, Adbullah Abusorrah., 2020). Characterized by their capacity for positional invariance and local feature recognition, CNNs had an architecture comprising an input layer, convolutional layers, pooling layers, and fully connected layers. These layers collectively distilled local and global features from input data, which was crucial for tasks such as sentence classification in ABSA. The CNN model's application to sentence classification involved the conversion of sentences into matrix representations, which were then processed through various layers to extract salient features for sentiment categorization (LiNan Zhu, MinHao Xu, YinWei Bao, YiFei Xu, and XiangJie Kong., 2022). In Figure 2.3, a CNN model for sentence classification was shown, and in Figure 2.4, a CNN model in deep tree was presented.

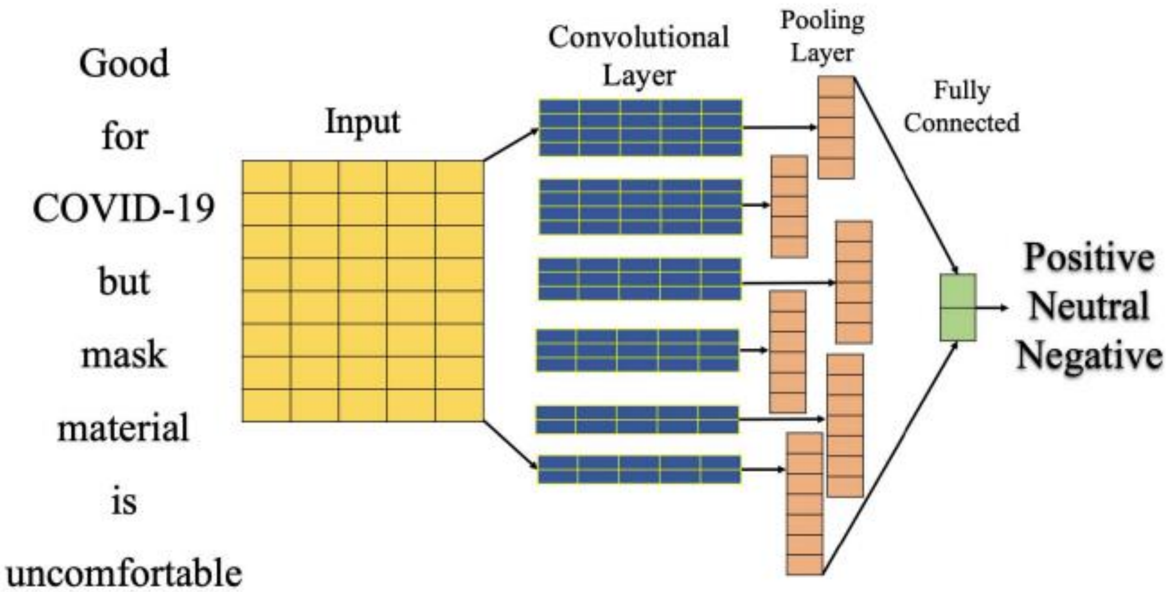


Figure 2. 3. CNN model for sentence classification (H. Liu et al., 2020)

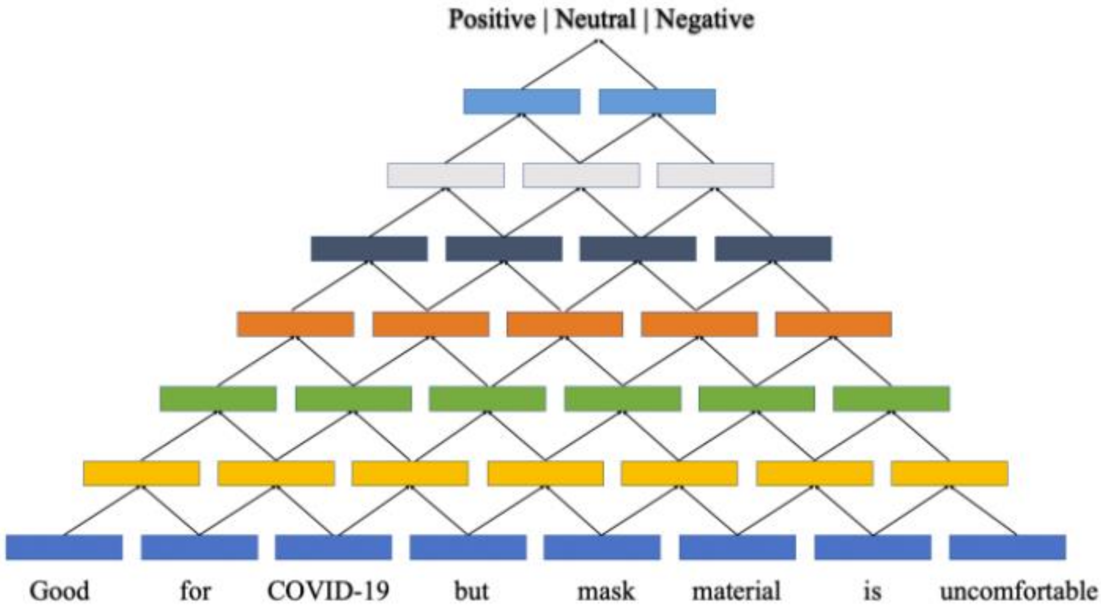


Figure 2. 4. CNN Model in Deep Tree (H. Liu et al., 2020)

2.3.2 Recurrent neural networks (RNN)

Zhejiang University of Technology highlighted Recurrent Neural Networks (RNNs) as essential to ABSA for their prowess in handling sequential data and capturing long-distance dependencies (Zhu et al., 2022). RNNs maintained a hidden state that evolved with sequential input, integrating past knowledge with current information. Bidirectional RNNs (BiRNNs) (Mike Schuster & Kuldip K. Paliwal, 1997) and advanced variants like Long Short-Term Memory (LSTM) (Sepp Hochreiter & Jurgen Schmidhuber, 1997) networks and Gated Recurrent Units (GRUs) (Kyunghyun Cho, Dzmitry Bahdanau, Fethi Bougares, Holger Schewenk, Yoshua Bengio., 2014) were developed to overcome limitations such as the vanishing gradient problem, thereby enhancing the model's memory and predictive capacity across longer data sequences.

Figure 2.5 and 2.6 showed the architecture of the basic RNN model. At each time step, the model took in an input word and processed it along with the previous hidden state. The hidden state represented the network's memory or knowledge of the previous inputs. The model calculated the hidden state at each time step based on the current input and the previous hidden state (H. Liu et al., 2020).

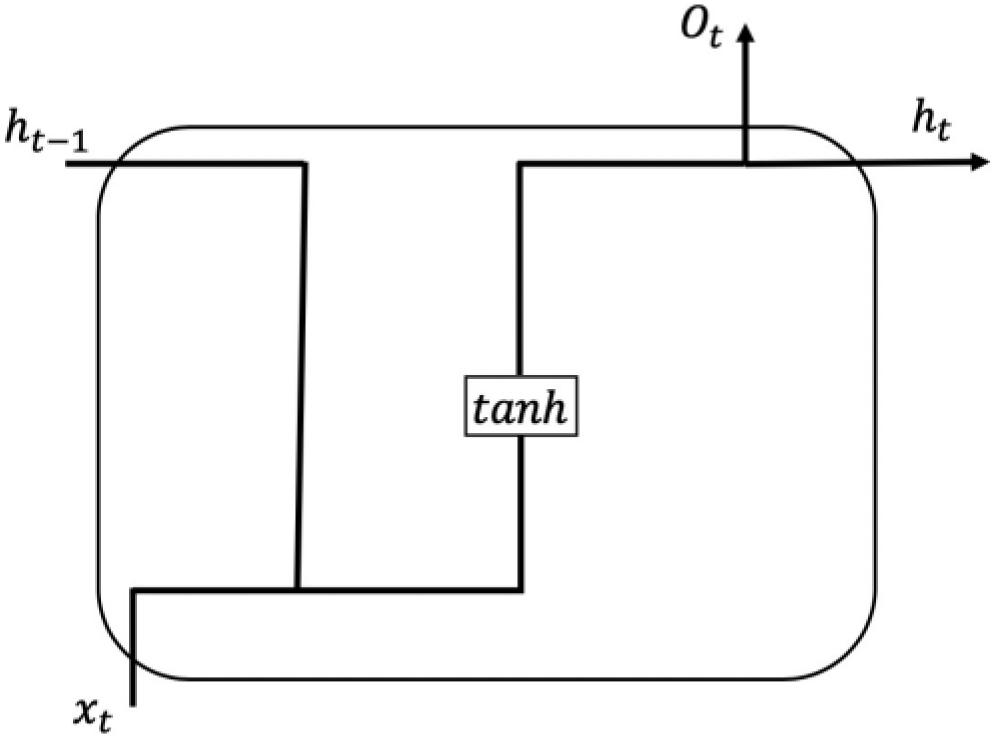


Figure 2. 5. Basic RNN Model (H. Liu et al., 2020)

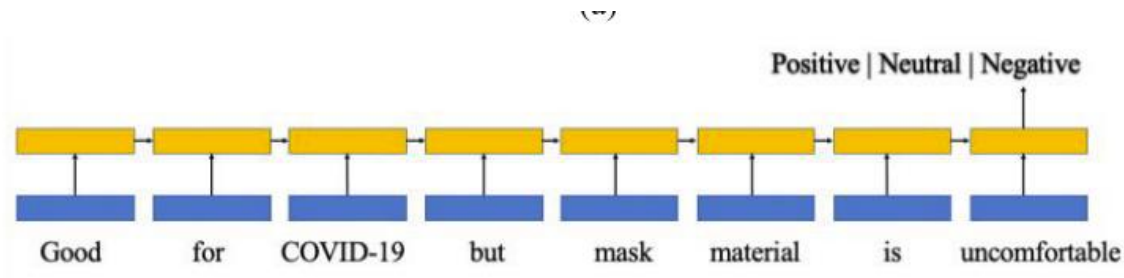


Figure 2. 6. RNN model in Deep Tree. (H. Liu et al., 2020)

2.3.2.1 Long Short-Term Memory (LSTM)

Long short-term memory (LSTM) was a subset of Recurrent Neural Networks (RNNs), adept at processing and retaining extensive sequence information, thereby mitigating the vanishing gradient problem often encountered in traditional RNNs (H. Liu et al., 2020). Characterized by their unique gate mechanism, LSTMs judiciously regulated the flow of information, ensuring a harmonious balance between preserving relevant past data and integrating new inputs. This functionality was particularly beneficial in Aspect-Based Sentiment Analysis (ABSA), as it allowed for a nuanced comprehension of sentence structures and the intricate interplay between aspects and their contextual words. Building upon this, the Bi-directional LSTM (Bi-LSTM) further enhanced these capabilities by learning sequential data from both left-to-right and right-to-left directions. This dual-directional learning approach amalgamated insights for more precise predictions, with studies showing Bi-LSTM's superiority over traditional LSTM models in text data analysis (Kitsuchart Pasupa & Thititorn Seneewong Na Ayutthaya, 2019; Barbara Plank, Anders Søgaard, Yoav Goldberg., 2016; Xin Wang, Yuanchao Liu, Chenjie Sun, Baoxun Wang, Xiaolong Wang., 2015; Peng Zhou, Suncong Zheng, Jiaming Xu, Hongyun Bao, Bo Xu., 2016). Figure 2.7 showed an example of LSTM model.

Zheng research team proposed a new attention-based LSTM model, named Position Attention-based Long Short-Term Memory (PosATT-LSTM) (JiangFeng Zeng, Xiao Ma, Ke Zhou., 2019). The PosATT-LSTM was proposed for aspect-level sentiment classification. It incorporated the explicit position context between the aspect and its context words into the sentiment classification process. The model considered the importance of each context word and used position-aware vectors to represent the position context. It achieved better performance compared to other baselines in terms of classification accuracy.

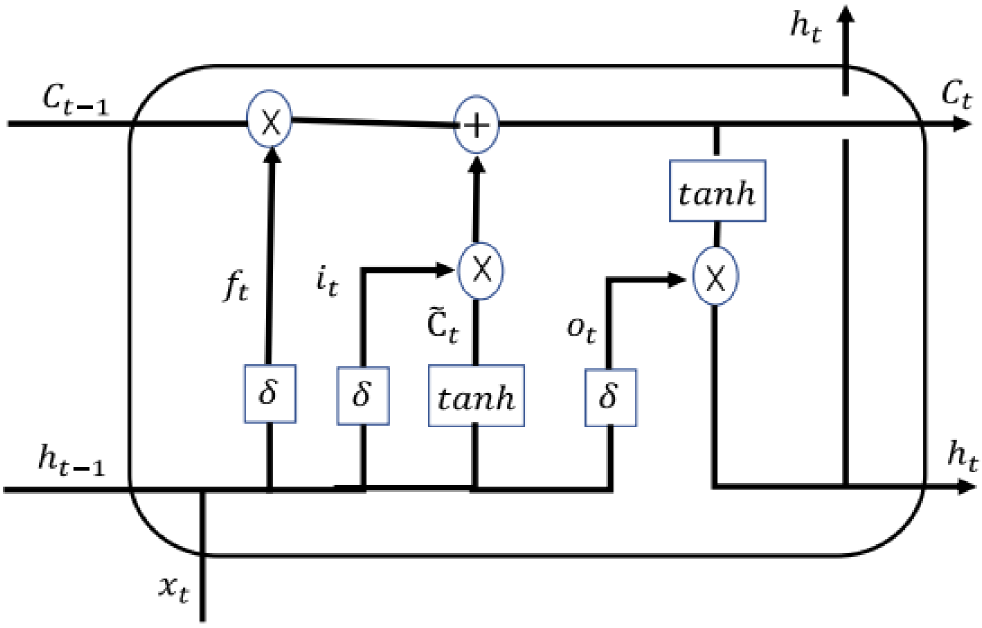


Figure 2. 7. LSTM model (H. Liu et al., 2020)

2.3.3 Memory neural networks (MemNN)

Memory Neural Networks (MemNN) introduced an external memory component, enabling the network to perform self-supervised learning for better representation of embeddings. This enhancement in context processing and awareness proved beneficial in ABSA, where understanding the nuanced sentiment was paramount (Zhu et al., 2022).

2.3.4 Graph neural networks (GNNs)

Graph Neural Networks (GNNs), including Graph Convolutional Networks (GCN) (Thomas N. Kipf & Max Welling, 2016) and Graph Attention Networks (GATs) (Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio., 2017), applied graph theory principles to neural network structures. By representing textual information as graphs, GNNs could capture the complex syntactic and semantic relationships within the text, which was instrumental in discerning sentiment based on aspects. (Zhu et al., 2022).

2.3.5 Recursive Neural Networks (RecNN)

Recursive Neural Networks (RecNN) modeled using a deep tree-like structure and were able to capture hierarchical data relationships. This structure was particularly useful in ABSA for analyzing sentiments in multi-aspect contexts because it allowed for a nuanced understanding of the interplay between different aspects and their associated sentiments (H. Liu et al., 2020).

Some notable RecNN-based methods included AdaRNN (Li Dong, Furu Wei, ChuanQi Tan, Duyu Tang, Ming Zhou, Ke Xu., 2014) and phrase RecNN (Thien Hai Nguyen & Kiyoaki Shirai, 2015). AdaRNN automatically selected composition functions instead of using handcrafted rules, treating context words as equally important give n multiple aspects. Phrase RecNN assigned different weights to each context word, considering their varying contributions toward a given aspect. These methods showed improvements in sentiment analysis accuracy compared to traditional RNN-based approaches. Figure 2.8 shows the RecNN model.

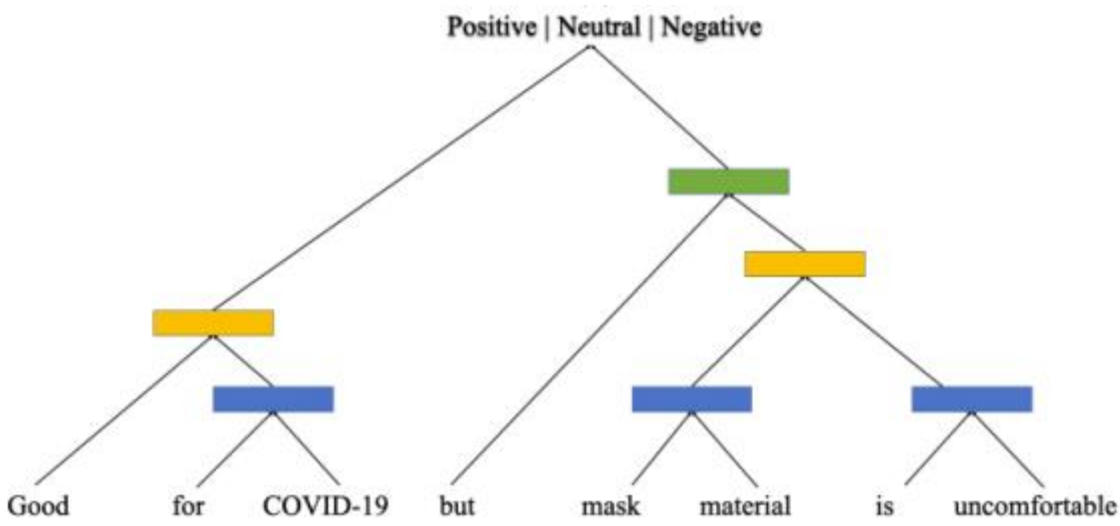


Figure 2. 8. RecNN models. (H. Liu et al., 2020)

2.3.6 Memory Network (MN)

In ABSA, the Memory Network (MN) used a long-term memory to store the information in text such as the conversation context. Comparing to traditional RNN, LSTM, and GRU, the MN had more advantage using MN because it allowed to store large amounts of information but the RNN, LSTM, and GRU used the hidden state to store the information. This made the RNN, LSTM, and GRU not have the ability to process a large and long conversation context information and may lose some information. Figure 2.9 showed a MN model. The MN model was formed by four components: input feature map, generalization, output feature map, and respond (H. Liu et al., 2020).

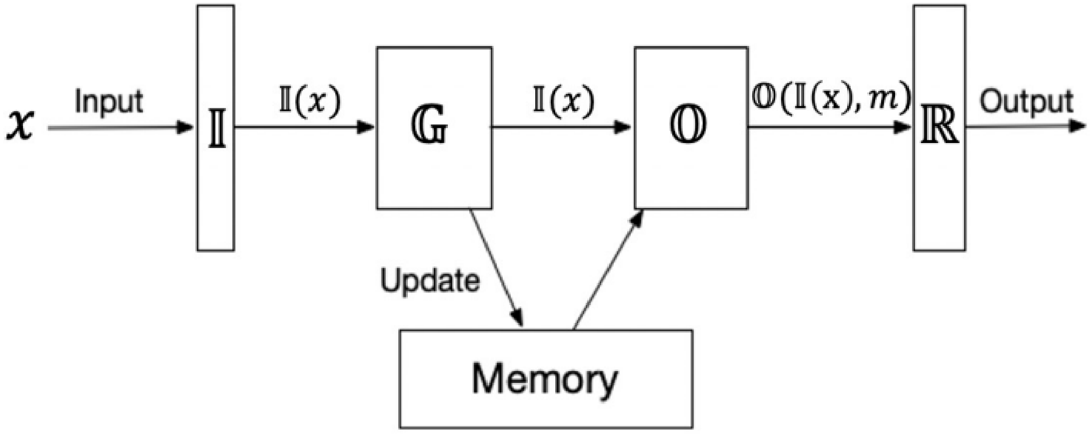


Figure 2. 9. MN Model (H. Liu et al., 2020)

2.4 Tokenization

The process of dividing text into smaller pieces known as tokens. For practically all NLP applications, including sentiment analysis, question answering, machine translation, information retrieval, it was an essential preprocessing step (Xinying Song, Alex Salciaunu, Yang Song, Dave Dopson, Denny Zhou., 2020).

Word-based tokenization process example:

Corpus: “The colour is red.”

After tokenization: ‘The’, ‘colour’, ‘is’, ‘red’

Above was the word-based tokenization process, where the word was separated. Not only using the simple tokenization method as shown above, but there was also another tokenization process named subword tokenization that broke down words into smaller units (subwords) that captured morphological and semantic information, while maintaining a manageable vocabulary size (Ankur A. Patel & Ajay Uppili Arasanipalai, 2021). This approach allowed for efficient representation of words by using a combination of prefixes, suffixes, and root words, thus reducing the vocabulary size and computational costs which were shown in Figure 2.10.

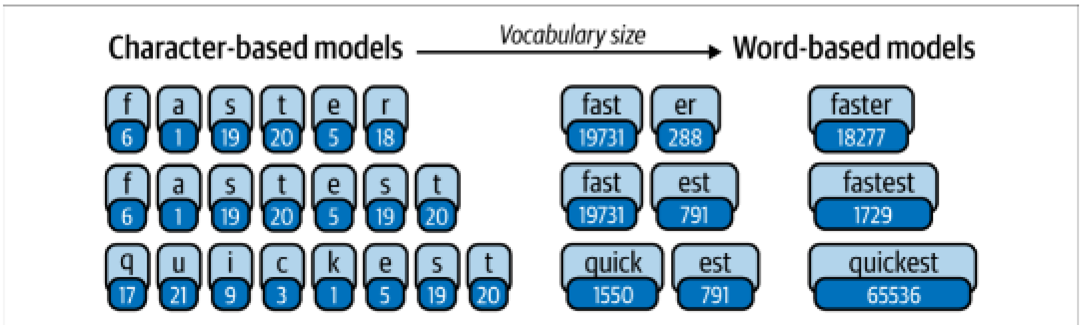


Figure 2. 10. Subtokenization (Patel & Arasanipalai, 2021)

2.5 Word Embedding

Word embedding was a technique in natural language processing (NLP) that represented words as vectors in a high-dimensional space. These vectors captured semantic relationships between words, with similar words having similar vector representations. The resulting word vectors could be used in various NLP tasks, such as text classification, sentiment analysis, and machine translation, to enhance the understanding of relationships and meanings between words.

The word embedding consisted of two main representation methods: one-hot encoding and distributed representation (Zhu et al., 2022). One-hot encoding represented a word as a vector, where the dimension was equal to the size of the vocabulary. In this representation, the word's corresponding position was set to 1, and all other positions were set to 0.

Corpus: “The colour is red.”

Assuming we were using one-hot encoding to define the colour. The sentence was be converted into 0 and 1. Since it was looking for colour, so the red convert to 1 and other text convert to be 0.

Result: “The colour is red.” --> 0 0 0 1

Nevertheless, one-hot encoding failed to capture the semantic correlation between words and might have led to the creation of sparse matrices and dimensionality problems when working with extensive vocabularies. Distributed representation employed low-dimensional vectors to depict words and had the ability to capture the semantic correlation among words. These techniques acquired word embeddings by taking into account the surrounding context in which words were used.

2.6 Transformer

The Transformer was an advanced tool that helped computers understand and create language like humans do. Before it came along, scientists had worked for over a hundred years developing ways for machines to handle language, building on ideas from smart folks like Markov and Turing. Unlike older methods that processed words one by one in order, the Transformer could look at whole sentences at once, making it much quicker and smarter at dealing with language. It came out in 2017 and was a big leap forward, making computers much better at translating languages and summarizing articles, among other things.

Figure 2.11 showed an architecture of the transformer introduced by the paper “Attention is all you need” in 2017 (Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N.Gomez, Lukasz Kaiser, Illia Polosukhin., 2017). The Transformer's model was a stack of 6 layers. On the left was the encoder and on the right was the decoder. Both the encoder and decoder were composed of 6-layer stacks and different numbers of attention sub-layers and FeedForward Networks (FFN) sub-layers.

- Encoder sub-layers
    - Multi-Head Self-Attention
    - Position-wise FeedForward Networks (FFN)
- Decoder sub-layers
    - Masked Multi-Head Self-Attention
    - Multi-Head-Decoder Self-Attention
    - Position-wise FeedForward Networks (FFN)

The encoder processes the input text, while the decoder generates the output text, making the transformer architecture versatile for various sequence-to-sequence tasks.

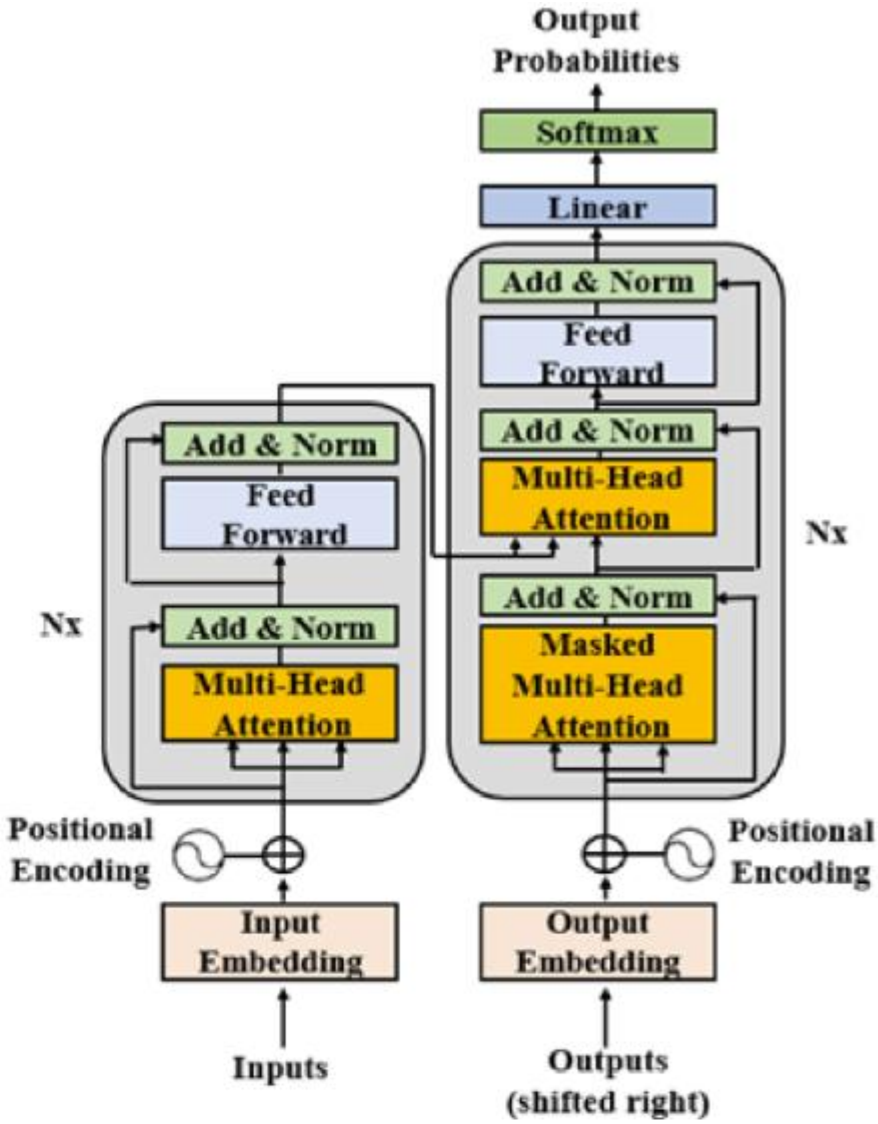


Figure 2. 11. The architecture of transformer (Denis Rothman, 2021)

In the book written by (Rothman, 2021), “Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more,” provided an explanation for Transformer's self-attention mechanism. The transformer could simultaneously consider all the words in a sentence when making predictions or understanding context. It had the ability to attend to all the words at once and capture complex relationships and dependencies in a sequence.

When given a word, such as "cat sat on the mat," the attention computed the dot product between the word and the vectors to discern the strongest connections between the word and its own representation ("cat" and "cat") (Rothman, 2021). The Transformer architecture stood out and transformed activities related to natural language processing because of its parallel processing of data. Figure 2.12 showed an example of Transformer's self-attention mechanism.

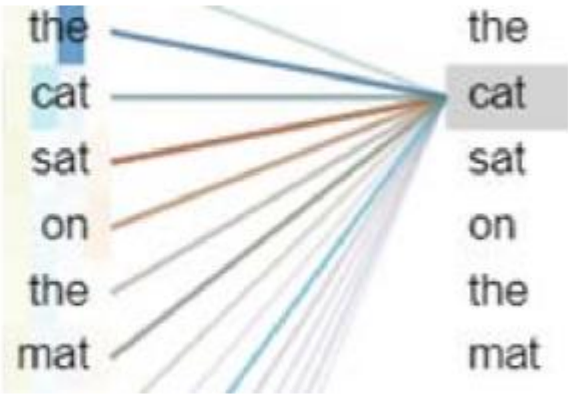


Figure 2. 12. An example of Transformer's self-attention mechanism (Rothman, 2021)

2.7 Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) was a groundbreaking technique in natural language processing (NLP) published by Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova in 2018. BERT belonged to the transformer-based model, which changed how NLP tasks were approached. Unlike previous models, BERT could understand words in context from both directions at once. This means it could consider words that came before and after a word in a sentence, helping it understand the meaning better. BERT was trained on a lot of text data, which helped it learn how words and phrases were used in different contexts. This training made BERT good at various NLP tasks like understanding languages, analysing feelings in texts, and answering questions. BERT's design also included self-attention mechanisms that helped it understand relationships between words better. Because of all these advancements, BERT quickly became one of the most popular tools in the NLP community, pushing NLP research forward and improving how computers understand and use human language.

The University of Technology Sweden conducted research about BERT contextual word representation for ABSA, showing its potential in identifying sentiment and aspects in text, especially in out-of-domain ABSA (Mickel Hoang, Oskar Alija Bihorac, Jacobo Rouces., 2019). The method used fine-tuning BERT with additional generated text, significantly outperforming previous state-of-the-art results on specific tasks. It also provided an in-depth study of aspect and sentiment polarity classification models and their combined use, demonstrating their effectiveness in various evaluations such as SemEval-2016 Task 5.

The Li research group investigated the modelling power of BERT for End-to-End Aspect-Based Sentiment Analysis (E2E-ABSA) (Xin Li, Lidong Bing, Wenxuan Zhang, Wai Lam., 2019). They showed that even though they used a simple neural network, BERT also had features to surpass existing models. Figure 2.13 showed an overview model for E2E-ABS.

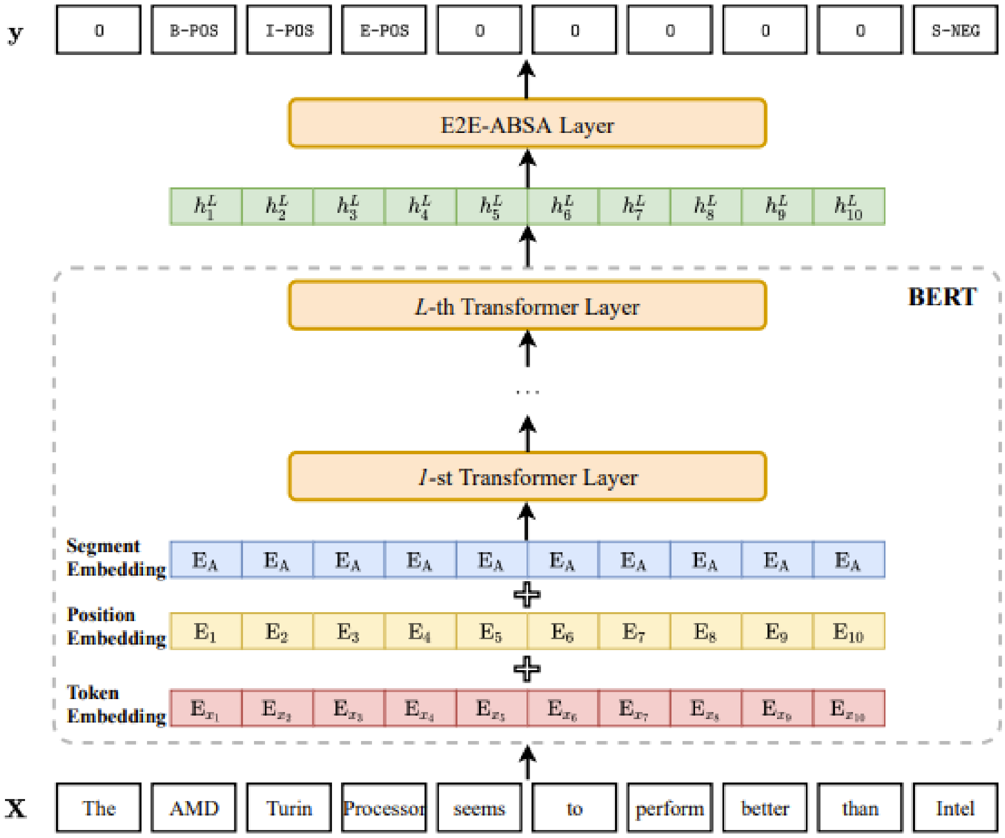


Figure 2. 13. An overview model for E2E-ABSA (X. Li et al., 2019).

Sun Yat-sen University explored the potential of BERT's intermediate layers in ABSA tasks. It introduced two novel pooling strategies: LSTM-Pooling (BERT-LSTM) and Attention-Pooling (BERT-Attention) to enhance BERT's fine-tuning performance and the task in NLP (Youwei Song, Jiahai Wang, Zhiwei Liang, Zhiyue Liu, Tao Jiang., 2020). This study proved the effectiveness of these strategies on various ABSA datasets, as well as their general applicability to other NLP tasks such as natural language reasoning. The results showed that these pooling strategies significantly improved the performance of BERT on the ABSA task. Figure 2.14 showed the BERT-LSTM model. For the BERT-Attention model, only the LSTM module was changed to an attention model.

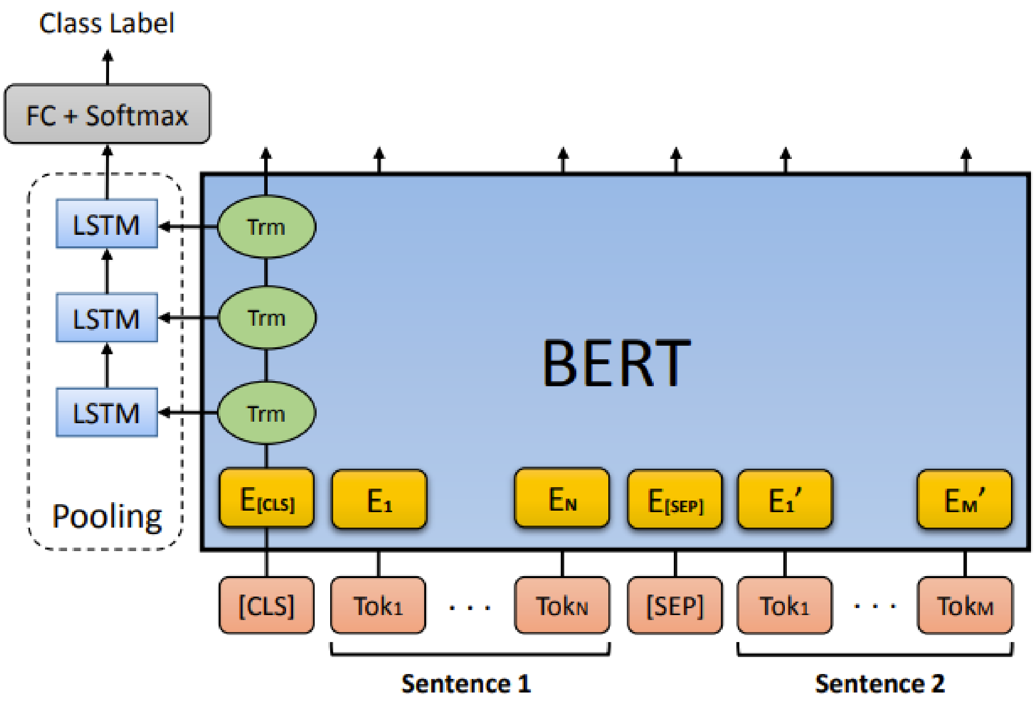


Figure 2. 14. BERT-LSTM model (Y. Song et al., 2020)

The Sarapathy research group discussed the interrelation between polarity detection and subjectivity detection in sentiment analysis, proposing a multitask learning framework using BERT embeddings (Ranjan Satapathy, Shweta Rajesh Pardeshi, Erik Cambria., 2022). It illustrated that BERT's bidirectional training significantly enhanced the performance in these tasks, outperforming baselines and setting a new standard in multitask learning for sentiment analysis. The approach leveraged information across datasets for related tasks, improving task-specific feature understanding. Figure 2.15 showed the multitask learning framework using BERT embeddings.

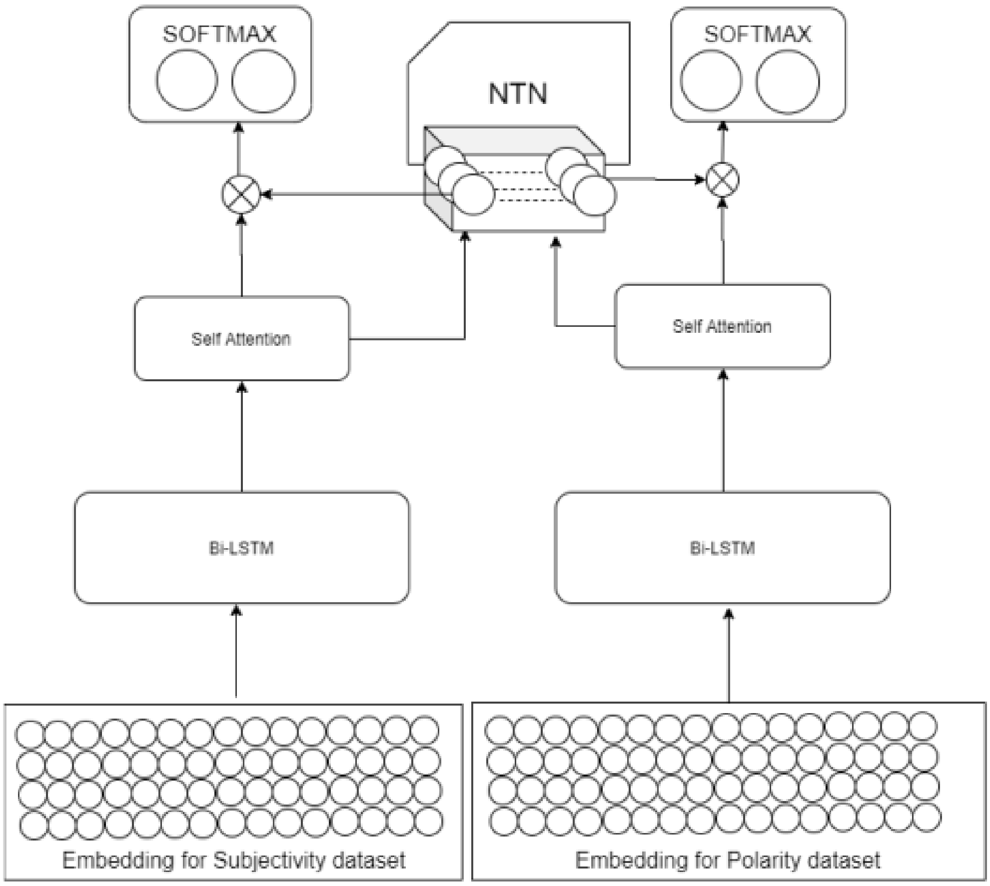


Figure 2. 15. Multitask learning framework using BERT embeddings model (Satapathy et al., 2022).

2.6 Subjectivity

Subjectivity in sentiment analysis referred to the expression of personal opinions, emotions, and preferences. It was about the personal lens through which an individual viewed a topic. For instance, a review saying, "I loved the vibrant colours of this painting" or a user’s personal experience about the food of the restaurant and laptop performance was subjective. This personal perspective was key for understanding how users felt about different aspects of what they were reviewing.

2.7 Objectivity (non-subjective)

Objectivity, on the other hand, dealt with facts and information presented without bias. It was the aspect of language that was factual and unbiased. For example, a statement like "This painting was done in 1920" or a factual detail about a product or service, like a laptop's technical specifications or a restaurant's location, was objective as it stated a verifiable fact without personal feelings. This factual aspect was important for a balanced analysis of different features.

3.0 Theoretical Framework

3.1 Project Stages

The primary steps to be executed in this project included data collecting, performing exploratory data analysis (EDA), data transformation, data cleaning, and data labelling. Data acquisition referred to the process of gathering and analysing data, which involved determining the source of the data and examining its characteristics. EDA referred to the process of identifying general patterns in the data. Data transformation involved extracting the nested table within each column and converting the dataset from JSON format and XML format to CSV format. Data cleaning was the elimination of undesirable rows from datasets and unifying the attribute names across all datasets. Data labelling involved the creation of a new attribute to classify data into two categories: subjective and non-subjective (objective). The project stage framework was depicted in Figure 3.1.

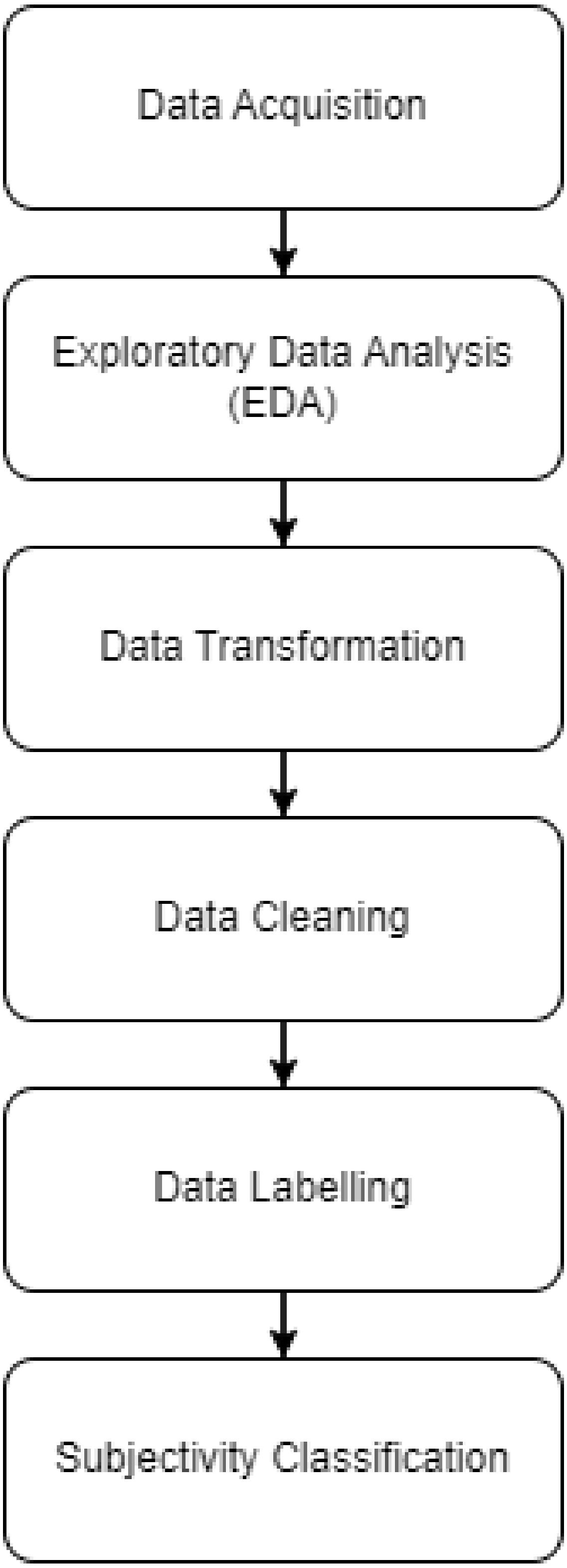


Figure 3. 1. Project stage framework

3.2 Data Acquisition

In this project, three separate datasets were utilized to enable thorough investigation and progress in aspect-based sentiment analysis (ABSA). SemEval 2014 datasets were created by (Pontiki et al., 2014), and MEMD datasets were created by (Hongjie Cai, Nan Song, Zengzhi Wang, QianKun Zhao, Ke Li, Siwei Wu, Shijie Liu, Jianfei Yu, Rui Xia., 2023). These two datasets focused on customer reviews in different domains. This multi-dataset approach aimed to enhance the robustness and generalizability of the aspect-based sentiment analysis model.

3.2.1 SemEval 2014 Datasets (https://github.com/avinashsai/BERT-Aspect/tree/master, 4/1/2024)

SemEval 2014 dataset was used to propel the progress of research in aspect-based sentiment analysis (ABSA). Se-14 focused on collecting customer reviews from the restaurant and laptop domains, providing separate test and train data for each domain in JSON format (https://github.com/avinashsai/BERT-Aspect/tree/master, 4/1/2024). Table 3.1 showed the SemEval 2014 dataset information. Tables 3.1 and 3.2 showed the maximum and minimum word in the sentences in each domain. Figure 3.2 displayed the file formats and file sizes of each domain in the SemEval 2014 datasets.

Table 3. 1. Sem 2014 dataset information

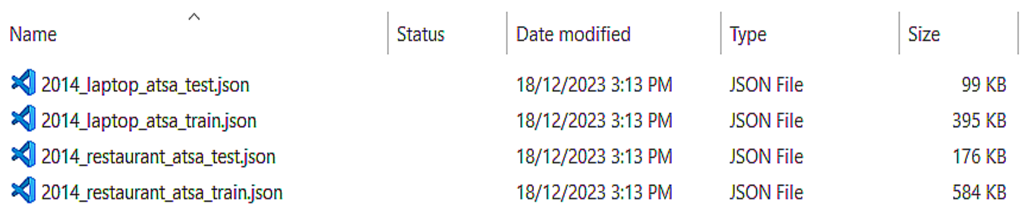
Dataset	Domains	Sentence	Maximum words	Minimum words
SemEval 2014, (https://github.com/avinashsai/BERT-Aspect/tree/master, 4/1/2024)	Laptop	3012	60	2
	Restaurant	4827	55	4

Table 3. 2. Example of max and min sentences in Laptop domain in SemEval 2014.

Maximum word	Minimum word
However , I certainly did enjoy watching Hedstr & ouml ; m and his team conduct t heir investigations into the two deaths , fin ding clues and piecing them together in thi s strongest book of the series so far.If you like to learn a bit about another country , b e a part of intricate investigations , and im merse yourself in the lives of fictional cha racters , then I certainly recommend crime fiction written by Camilla L & auml ; ckb erg .	nice packing.

Table 3. 3. Example of max and min sentences in Restaurant domain in SemEval 2014.

Maximum word	Minimum word
the new menu has a few creative items,the y were smart enough to keep some of the old favorites (but they raised the prices), t he staff is friendly most of the time, but i must agree with the person that wrote abo ut their favorite words: no, can't, sorry..., b oy, they won't bend the rules for anyone.	good food.



Name	Status	Date modified	Type	Size
2014_laptop_atsa_test.json		18/12/2023 3:13 PM	JSON File	99 KB
2014_laptop_atsa_train.json		18/12/2023 3:13 PM	JSON File	395 KB
2014_restaurant_atsa_test.json		18/12/2023 3:13 PM	JSON File	176 KB
2014_restaurant_atsa_train.json		18/12/2023 3:13 PM	JSON File	584 KB

Figure 3. 2. File type and file size for Sem 2014 dataset

3.2.2 MEMD Datasets (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024)

The MEMD dataset collected customer reviews from five domains: book domain, clothing domain, hotel domain, laptop domain, and restaurant domain, and these domains separated the data into dev data, test data, and train data. The MEMD dataset was provided by Nanjing University of Science and Technology in JSON format on GitHub repository (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024). Table 3.4 showed the MEMD dataset information. Tables 3.5, 3.6, 3.7, 3.8, and 3.9 showed the maximum and minimum word in the sentences for each domain. Figure 3.3 showed the file types and file size of each domain in the MEMD datasets.

Table 3. 4. MEMD dataset information

Datasets	**Domains**	**Sentences**	**Maximum words**	**Minimum words**
MEMD (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024)	Book	5978	195	1
	Clothing	3701	114	1
	Hotel	5622	69	1
	Laptop	7382	107	2
	Restaurant	11193	191	1

Table 3. 5. Example of max and min sentences in MEMD book domain.

Maximum word	Minimum word
However , I certainly did enjoy watching Hedstr & ouml ; m and his team conduct their investigations into the two deaths , finding clues and piecing them together in this strongest book of the series so far.If you like to learn a bit about another country , be a part of intricate investigations , and immerse yourself in the lives of fictional characters , then I certainly recommend crime fiction written by Camilla L & auml ; ckberg .	Gift

Table 3. 6. Example of max and min sentences in MEMD clothing domain.

Maximum word	Minimum word
Not the best quality leather , but they are pretty well made , so they will last through a dance season if you are taking a class , and probably much longer depending on use.The Size Question amazon asked stated too small as the only option that made sense , but the aren't really too small - you just have to know that dance shoes are always at least a size smaller than street shoes - sometimes even two sizes .	Comfy

Table 3. 7. Example of max and min sentences in MEMD hotel domain.

Maximum word	Minimum word
Not the best quality leather , but they are pretty well made , so they will last through a dance season if you are taking a class , and probably much longer depending on use.The Size Question amazon asked stated too small as the only option that made sense , but the aren't really too small - you just have to know that dance shoes are always at least a size smaller than street shoes - sometimes even two sizes .	Comfy

Table 3. 8. Example of max and min sentences in MEMD laptop domain.

Maximum word	Minimum word
i hate to knock acer as they ' re the only brand and i ' ve ever purchased and this is the first major issue i ' ve ever had with them , but it was frustrating spending that much time charging and almost finishing the configurations and set - up of a brand - new product and then not being able to turn it on to use it .	good screen .

Table 3. 9. Example of max and min sentences in MEMD restaurant domain.

Maximum word	Minimum word
So , while Junior 's might not be your absolute *favoritest* place , it 's definitely a great choice for those times when everyone needs to compromise a little to find a menu that is flexible enough for a variety of diets.The place is pretty tight , so seating for parties of 5+ is limited to the very back two tables .	Fabulous .

Name	Date modified	Type	Size
memd_book_Dev.json	18/12/2023 3:18 PM	JSON File	201 KB
memd_book_Test.json	18/12/2023 3:18 PM	JSON File	425 KB
memd_book_Train.json	18/12/2023 3:18 PM	JSON File	1,503 KB
memd_clothing.json	18/12/2023 3:18 PM	JSON File	288 KB
memd_clothing_Dev.json	18/12/2023 3:18 PM	JSON File	151 KB
memd_clothingTrain.json	18/12/2023 3:18 PM	JSON File	1,095 KB
memd_hotel_Dev.json	18/12/2023 3:18 PM	JSON File	265 KB
memd_hotel_Test.json	18/12/2023 3:18 PM	JSON File	530 KB
memd_hotel_Train.json	18/12/2023 3:18 PM	JSON File	1,852 KB
memd_laptop_Dev.json	18/12/2023 3:18 PM	JSON File	261 KB
memd_laptop_Test.json	18/12/2023 3:18 PM	JSON File	495 KB
memd_laptop_Train.json	18/12/2023 3:18 PM	JSON File	1,876 KB

Figure 3. 3. File type and file size for MEMD dataset

3.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was a process used by data scientists to analyse the datasets in different patterns such as bar charts or boxplots. In this project, the bar chart was used to analyse the number of three classes in sentiment attributes in three datasets.

3.3.1 SemEval 2014 EDA (https://github.com/avinashsai/BERT-Aspect/tree/master, 4/1/2024)

For the sentiment attributes, the laptop data consisted of 1328 instances of positive sentiment, 994 instances of negative sentiment, and 629 instances of neutral sentiment. For the subjectivity attributes, it consisted of 2322 instances of subjectivity and 629 instances of objectivity. Figures 3.4 and 3.5 showed them respectively.

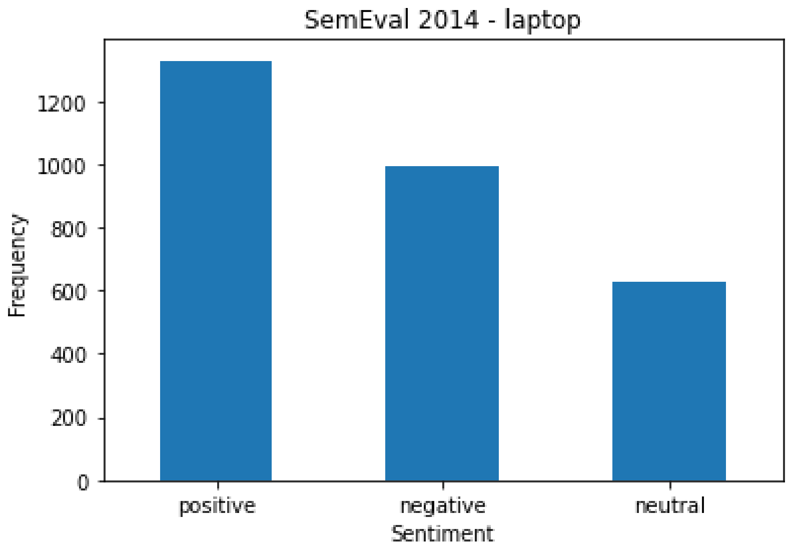


Figure 3. 4. SemEval 2014 Laptop (sentiment attributes)

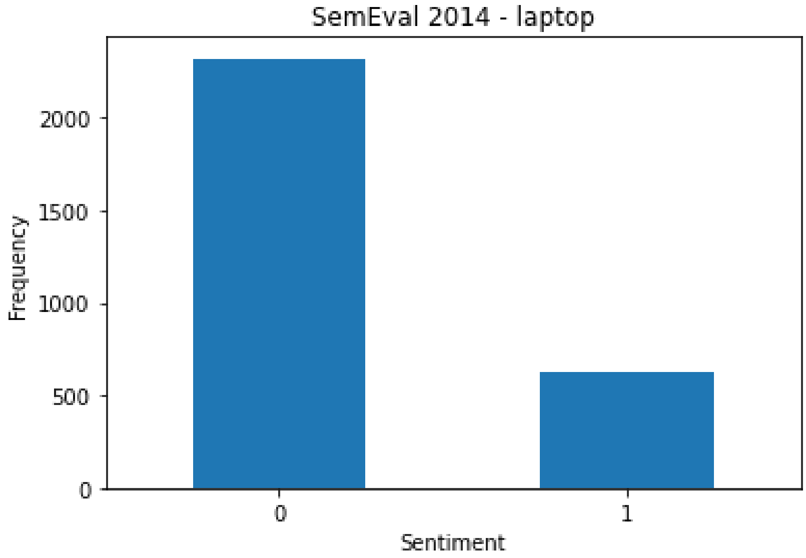


Figure 3. 5. SemEval 2014 Laptop (subjectivity attributes)

In the restaurant data, the sentiment attributes consisted of 2892 positive sentiment instances, 1001 negative sentiment instances, and 829 neutral sentiment instances. For the subjectivity attributes, it consisted of 3893 instances of subjectivity and 829 instances of objectivity. Figures 3.6 and 3.7 showed them respectively.

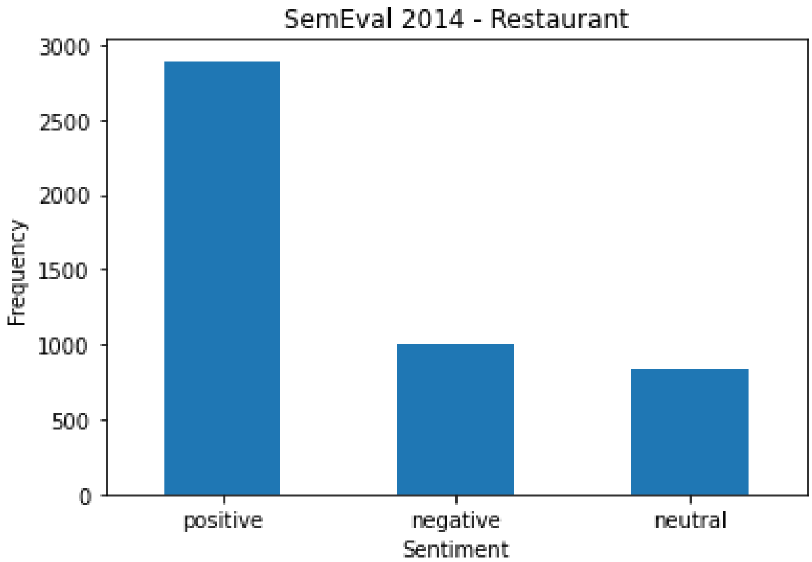


Figure 3. 6. Sem 2014 Restaurant (sentiment attributes)

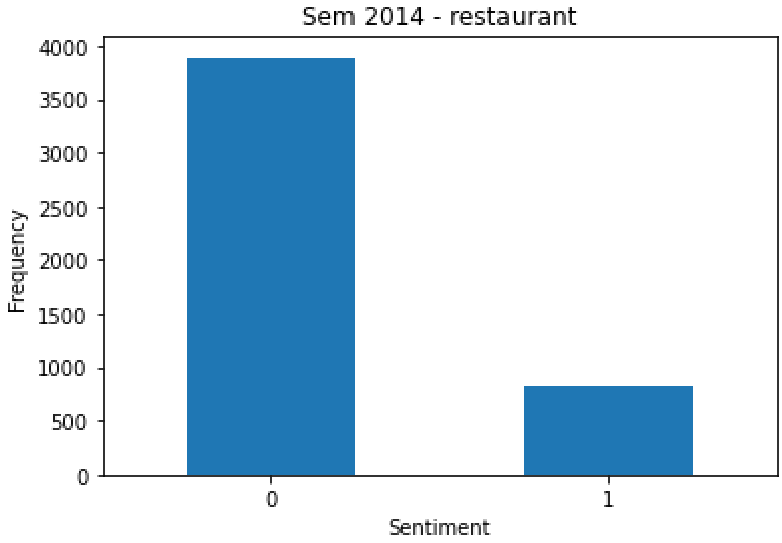


Figure 3. 7. Sem 2014 Restaurant (subjectivity attributes)

3.3.2 MEMD EDA (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024)

In the MEMD book sentiment attributes, the positive class contained 3672 instances, the negative class contained 1393 instances, and the neutral class contained 155 instances. In the MEMD book subjectivity attributes, the subjectivity class contained 5065 instances, and the objectivity class contained 155 instances. Figures 3.8 and 3.9 showed them respectively.

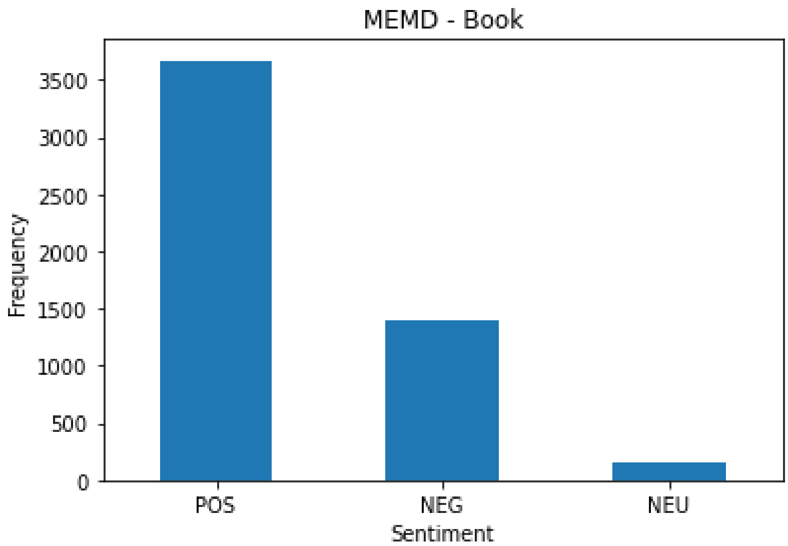


Figure 3. 8. MEMD Book (sentiment attributes)

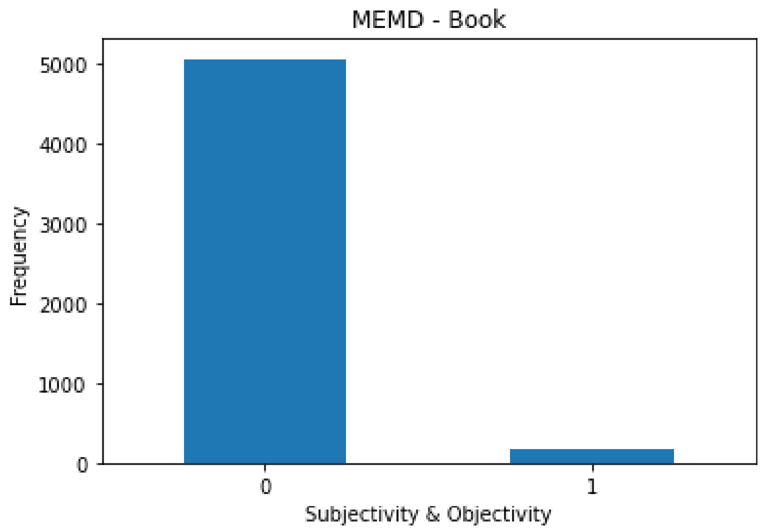


Figure 3. 9. MEMD Book (subjectivity attributes)

In the MEMD clothing sentiment attributes, the positive class contained 2220 instances, the negative class contained 850 instances, and the neutral class contained 63 instances. In the MEMD clothing subjectivity attributes, the subjectivity class contained 3070 instances, and the objectivity class contained 63 instances. Figures 3.10 and 3.11 showed them respectively.

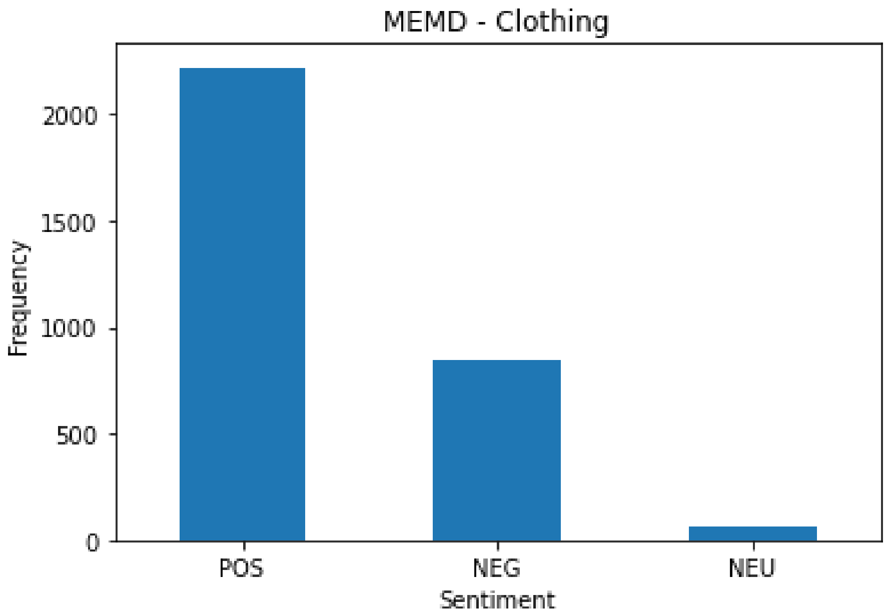


Figure 3. 10. MEMD Clothing (sentiment attributes)

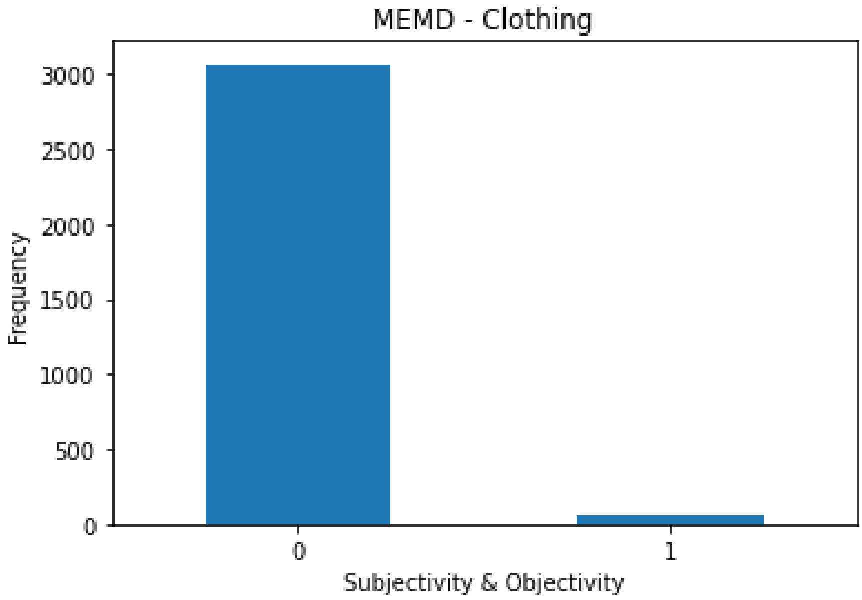


Figure 3. 11. MEMD Clothing (subjectivity attributes)

In the MEMD hotel sentiment attributes, the positive class contained 5314 instances, the negative class contained 288 instances, and the neutral class contained 20 instances. In the MEMD hotel subjectivity attributes, the subjectivity class contained 5602 instances, and the objectivity class contained 20 instances. Figures 3.12, and 3.13 showed them respectively.

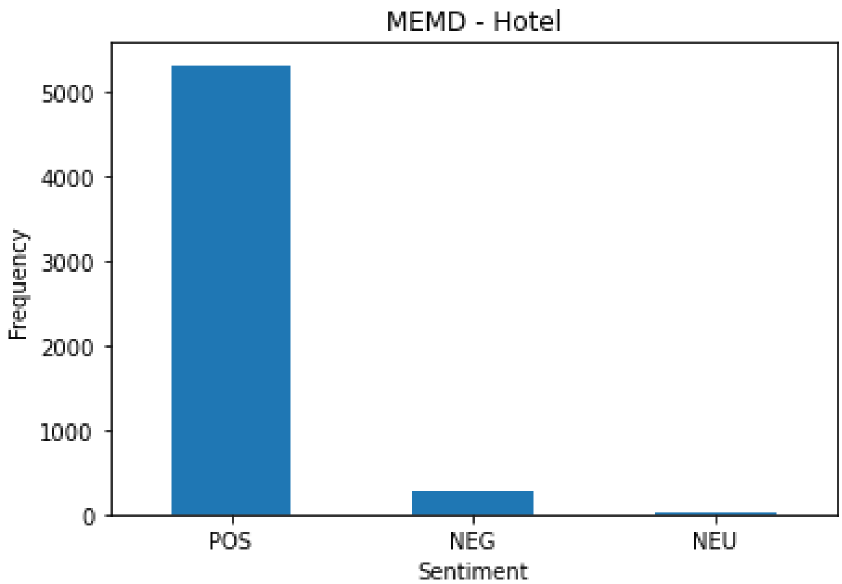


Figure 3. 12. MEMD Hotel (sentiment attributes)

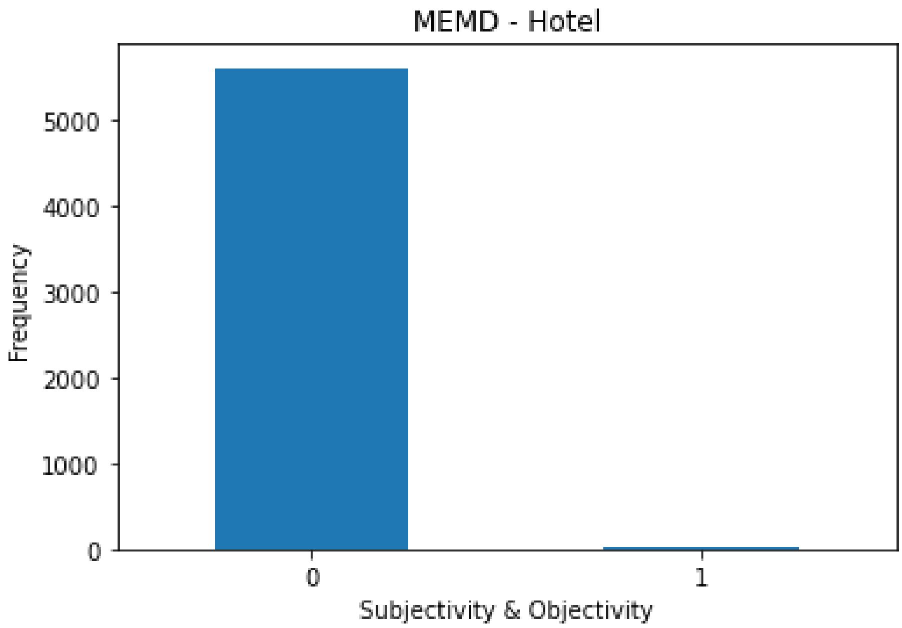


Figure 3. 13. MEMD Hotel (subjectivity attributes)

In the MEMD laptop sentiment attributes, the positive class contained 3714 instances, the negative class contained 2452 instances, and the neutral class contained 394 instances. In the MEMD laptop subjectivity attributes, the subjectivity class contained 6166 instances, and the objectivity class contained 394 instances. Figures 3.14, and 3.15 showed them respectively.

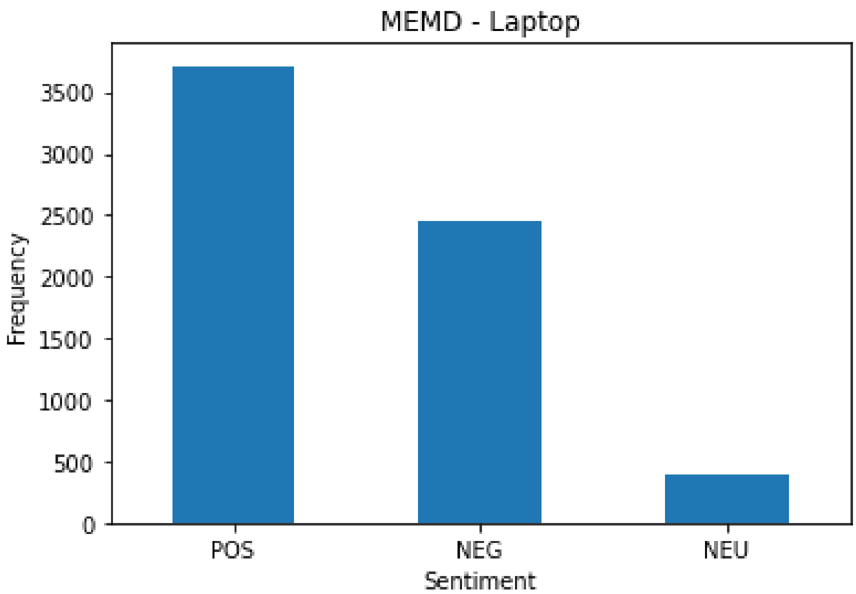


Figure 3. 14. MEMD Laptop (sentiment attributes)

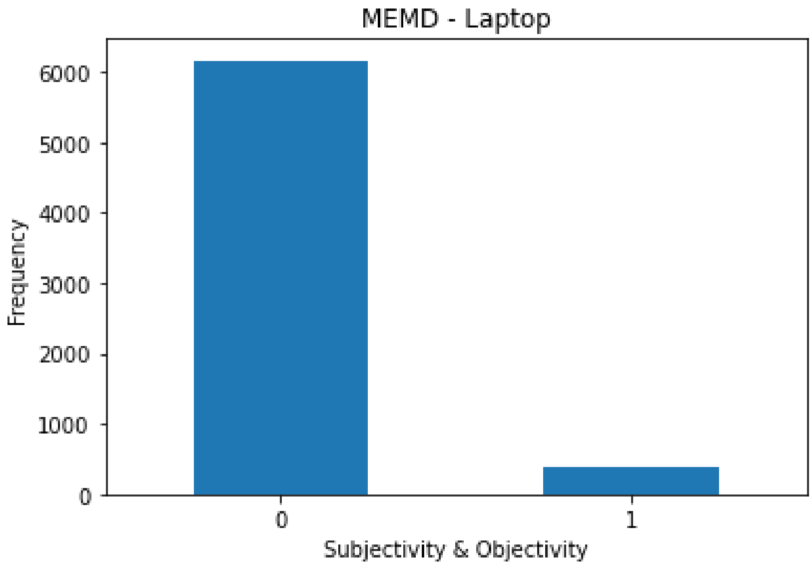


Figure 3. 15. MEMD Laptop (subjectivity attributes)

In the MEMD restaurant sentiment attributes the positive class contains 7821, negative class contains 2119, neutral class contains 140. In the MEMD hotel subjectivity attributes, the subjectivity class contains 9400, and objectivity class contains 140. Figures 3.16 and 3.17 show the respectively.

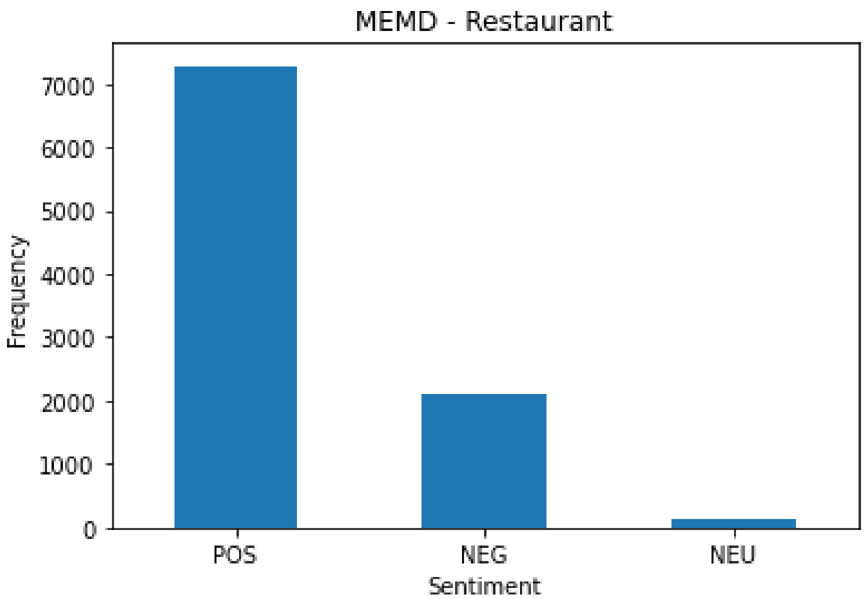


Figure 3. 16. MEMD Restaurant (sentiment attributes)

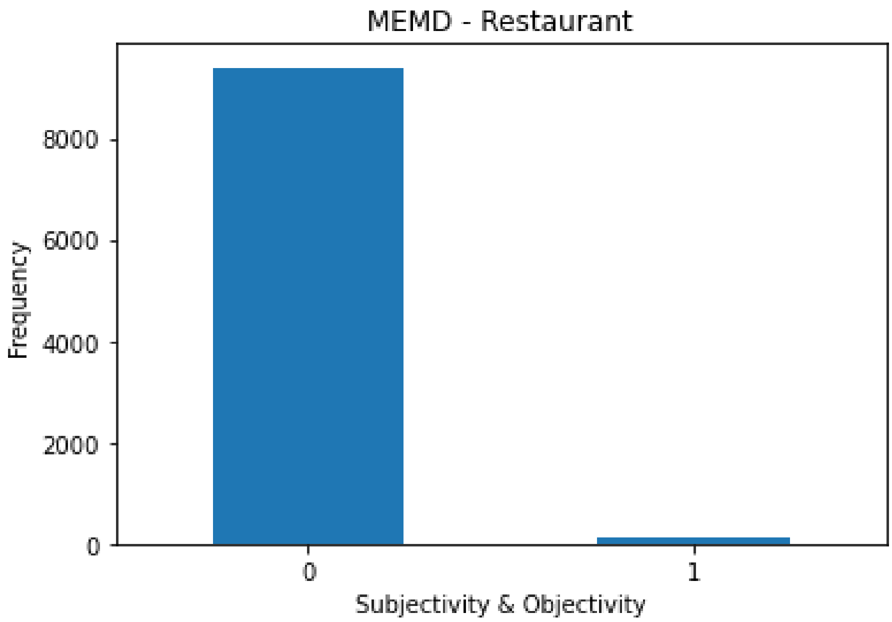


Figure 3. 17. MEMD Restaurant (subjectivity attributes)

3.4 Data Transformation

In this project, the main goal of data transformation was to expand the nested tables inside the attributes and convert the dataset from JSON format or XML format to CSV format. In this process, Power BI, a Microsoft product, was used to expand the nested tables and convert the datasets into CSV format, with the process of using Power BI displayed in Figure 3.16.

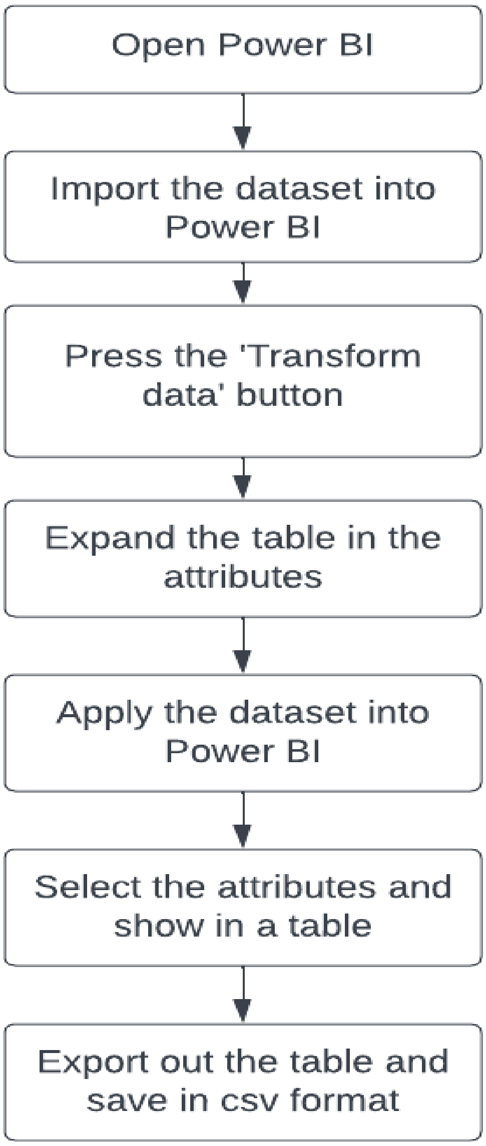


Figure 3. 18. Flowchart for using Power BI

3.4.1 SemEval 2014 (https://github.com/avinashsai/BERT-Aspect/tree/master, 4/1/2024)

The GitHub repository already had the dataset without any nested tables. Hence, the only requirement for this dataset was to convert it from JSON format to CSV format. The dataset was successfully converted from JSON format to CSV format. Figure 3.17 displayed the Se-14 datasets in CSV format.

Name	Date modified	Type	Size
laptop_2014_test.csv	29/12/2023 12:25 PM	Microsoft Excel C...	71 KB
laptop_2014_train.csv	29/12/2023 12:25 PM	Microsoft Excel C...	294 KB
restaurant_2014_test.csv	29/12/2023 12:25 PM	Microsoft Excel C...	128 KB
restaurant_2014_train.csv	29/12/2023 12:26 PM	Microsoft Excel C...	425 KB

Figure 3. 19. SemEval 2014 dataset in csv format

3.4.2 MEMD (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024)

The MEMD dataset contains one or multiple nested tables under the aspect attribute because one sentence can have multiple aspect terms. For example, in the MEMD dataset laptop domains, the sentence "I am pleased with the fast log on, speedy wifi connection, and long battery life (>6 hrs)." contains three aspect terms and three sentiments: "log on" with the sentiment positive, "wifi connection" with the sentiment positive, and "battery life" with the sentiment positive.

Figure 3.18 shows the data in a JSON file, where the aspect and sentiment are under quadruples attributes. Using Power BI to expand the aspect and sentiment out from quadruples attributes and make them as two new attributes. The attributes are automatically named as quadruples.aspect.term and quadruples.sentiment.

```
"raw_words": "( this was really at odds with her personality in the book )
". Confusing and just plain awful to read .",
"task": "ACOS",
"quadruples": [
  {
    "aspect": {
      "from": -1,
      "to": -1,
      "term": [
        "NULL"
      ]
    },
    "category": "Content#Plot",
    "opinion": {
      "from": 16,
      "to": 19,
      "term": [
        "just",
        "plain",
        "awful"
      ]
    },
    "sentiment": "NEG"
  },
  {
    "aspect": {
      "from": -1,
      "to": -1,
      "term": [
        "NULL"
      ]
    },
    "category": "Content#Plot",
    "opinion": {
      "from": 14,
      "to": 15,
      "term": [
        "Confusing"
      ]
    },
    "sentiment": "NEG"
```


Figure 3. 20. An example of nested table in JSON format

In MEMD datasets, many unnecessary attributes were present, so only the dataset with raw_words, quadruples.aspect.term, and quadruples.sentiment attributes were converted into a CSV file. Figure 3.19 showed the total number of attributes in MEMD datasets, and Figure 3.20 showed the dataset in CSV format.

```
before = memd_laptop_dev_original.columns
after = memd_laptop_dev.columns

print(f'Before --> {before}')
print("-----------------------------------------------------------------------------------------------------")
print(f'After -->{after}')
```

```
Before --> Index(['raw_words', 'task', 'quadruples.aspect.from', 'quadruples.aspect.to',
       'quadruples.aspect.term', 'quadruples.category',
       'quadruples.opinion.from', 'quadruples.opinion.to',
       'quadruples.opinion.term', 'quadruples.sentiment'],
      dtype='object')
-----------------------------------------------------------------------------------------------------
After -->Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
```

Figure 3. 21. Number of columns in MEMD dataset (before and after)

Name	Date modified	Type	Size
memd_book_Dev.csv	18/12/2023 4:09 PM	Microsoft Excel C...	63 KB
memd_book_Test.csv	18/12/2023 4:11 PM	Microsoft Excel C...	138 KB
memd_book_Train.csv	18/12/2023 4:12 PM	Microsoft Excel C...	485 KB
memd_clothing_Dev.csv	18/12/2023 4:12 PM	Microsoft Excel C...	28 KB
memd_clothing_Test.csv	18/12/2023 4:12 PM	Microsoft Excel C...	47 KB
memd_clothing_Train.csv	18/12/2023 4:13 PM	Microsoft Excel C...	194 KB
memd_laptop_Dev.csv	18/12/2023 4:15 PM	Microsoft Excel C...	61 KB
memd_laptop_Test.csv	18/12/2023 4:15 PM	Microsoft Excel C...	105 KB
memd_laptop_Train.csv	18/12/2023 4:15 PM	Microsoft Excel C...	441 KB
memd_restaurant_Dev.csv	18/12/2023 4:16 PM	Microsoft Excel C...	100 KB
memd_restaurant_Test.csv	18/12/2023 4:17 PM	Microsoft Excel C...	192 KB
memd_restaurant_Train.csv	18/12/2023 4:17 PM	Microsoft Excel C...	792 KB

Figure 3. 22. MEMD dataset in csv format

3.5 Data Cleaning

Data cleaning was the procedure of eliminating errors and correcting inconsistencies, duplicates, or data that was not properly prepared prior to analysis. Data cleaning was an important and vital activity that contributed to the successful accomplishment of the objectives of this study. In this project, the main goal of data cleaning was to standardize the attribute names to "sentences," "aspect," and "sentiment" similar to Sem-2014 datasets attributes. This was because the datasets had different attribute names, but all three datasets used the same attributes, so standardizing the attribute names made it easier to use in the future.

3.5.1 MEMD (https://github.com/NUSTM/MEMD-ABSA, 4/1/2024)

The MEMD dataset needed to perform a rename attributes function using the rename function in Python Pandas. The attribute name "raw_words" was renamed to "sentence," "quadruples.aspect.term" was renamed to "aspect," and "quadruples.sentiment" was renamed to "sentiment." Figure 3.21 showed the Python code that was used to rename the attributes' names and displayed the new attributes' names.

```
column_mapping= {'raw_words':'sentence',
                 'quadruples.sentiment' : 'sentiment',
                 'quadruples.aspect.term': 'aspect'}

memd_name = [memd_book_dev,memd_book_test,memd_book_train,memd_clothing_dev,memd_clothing_test,memd_clothing_train,
           memd_hotel_dev,memd_hotel_test,memd_hotel_train,memd_laptop_dev,memd_laptop_test,memd_laptop_train,
           memd_restaurant_dev,memd_restaurant_test,memd_restaurant_train]

memd_name_string = ['memd_book_dev','memd_book_test','memd_book_train','memd_clothing_dev',
                   'memd_clothing_test','memd_clothing_train','memd_hotel_dev','memd_hotel_test',
                   'memd_hotel_train','memd_laptop_dev','memd_laptop_test','memd_laptop_train',
                   'memd_restaurant_dev','memd_restaurant_test','memd_restaurant_train']

print('Before Rename')
for i in range(len(memd_name)):
    print(f'{memd_name_string[i]} ---> {memd_name[i].columns}')
    memd_name[i] = memd_name[i].rename(columns=column_mapping)

print('--------------------------------------------------------------------------------------------')
print('After Rename')
# check the the column name is correct or not.
for i in range(len(memd_name)):
    print(f'{memd_name_string[i]} ---> {memd_name[i].columns}')
```

```
Before Rename
memd_book_dev ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_book_test ---> Index(['raw_words', 'quadruples.aspect.term', 'quadruples.sentiment'], dtype='object')
memd_book_train ---> Index(['raw_words', 'quadruples.aspect.term', 'quadruples.sentiment'], dtype='object')
memd_clothing_dev ---> Index(['raw_words', 'quadruples.aspect.term', 'quadruples.sentiment'], dtype='object')
memd_clothing_test ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_clothing_train ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_hotel_dev ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_hotel_test ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_hotel_train ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_laptop_dev ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_laptop_test ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_laptop_train ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_restaurant_dev ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_restaurant_test ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
memd_restaurant_train ---> Index(['raw_words', 'quadruples.sentiment', 'quadruples.aspect.term'], dtype='object')
--------------------------------------------------------------------------------------------
After Rename
memd_book_dev ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_book_test ---> Index(['sentence', 'aspect', 'sentiment'], dtype='object')
memd_book_train ---> Index(['sentence', 'aspect', 'sentiment'], dtype='object')
memd_clothing_dev ---> Index(['sentence', 'aspect', 'sentiment'], dtype='object')
memd_clothing_test ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_clothing_train ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_hotel_dev ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_hotel_test ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_hotel_train ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_laptop_dev ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_laptop_test ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_laptop_train ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_restaurant_dev ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_restaurant_test ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
memd_restaurant_train ---> Index(['sentence', 'sentiment', 'aspect'], dtype='object')
```

Figure 3. 23. Attributes name in MEMD dataset (Before and After)

3.6 Data Labelling

In order to compare the performance of aspect-based sentiment analysis when combined with subjective classification, a new attribute needed to be manually labelled, and the new attribute name was "subjectivity."

In the subjectivity attributes, there were two categories: subjectivity and objectivity. In the sentiment attributes, the positive and negative categories were defined as subjective, while the neutral category was defined as objective. Figure 3.22 showed the flowchart of defining subjectivity and objectivity classes, and Figure 3.23 showed the Python code used to define subjectivity and objectivity classes.

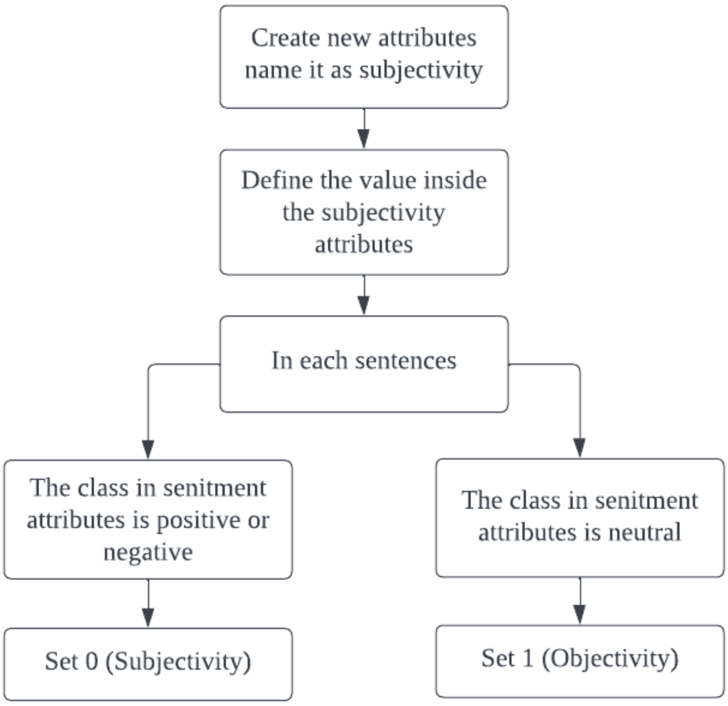


Figure 3. 24. Flowchart of define subjectivity attributes (Sentiment Labelling Approach).

```python
# create a new class for subjective and objective
def map_polarity_to_sub_obj(polarity):
    if polarity in ['positive', 'negative', 'Positive', 'Negative', 'POS', 'NEG', 'pos', 'neg']:
        return 0
    elif polarity in ['neutral', 'Neutral', 'NEU', 'neu']:
        return 1

attributes ='sentiment'

csv_name = ['memd_book_Dev.csv','memd_book_Test.csv', 'memd_book_Train.csv','memd_clothing_Dev.csv',
            'memd_clothing_Test.csv','memd_clothing_Train.csv','memd_hotel_Dev.csv', 'memd_hotel_Test.csv',
            'memd_hotel_Train.csv', 'memd_laptop_Dev.csv','memd_laptop_Test.csv','memd_laptop_Train.csv',
            'laptop_2014_test.csv','laptop_2014_train.csv', 'restaurant_2014_test.csv','restaurant_2014_train.csv']

# Perform the map_polarity_to_sub_obj function
for i in range(len(df_name1)):
    df_name1[i]['subjectivity'] = df_name1[i][attributes].apply(lambda x: map_polarity_to_sub_obj(x))
```

Figure 3. 25. Define subjectivity and objectivity classes python code.

3.7 Subjectivity Classification

In sentiment analysis, subjectivity referred to the expression of personal opinions, emotions, and judgments embedded within the text, distinguishing it from objective content, which was factual and neutral. Subjective text was pivotal as it conveyed the personal stance and feelings of the author towards a particular topic, product, or service, making it the focal point for sentiment analysis endeavours. This distinction was crucial because understanding and analysing subjective expressions enabled researchers and practitioners to glean insights into public sentiment, market trends, and individual preferences, thereby driving informed decision-making and strategies in various domains. As such, the ability to accurately identify and interpret subjective content was a fundamental aspect of sentiment analysis that enhanced the understanding of human emotions and opinions in textual data. In this project, using sentiment to label the subjective and objective (non-subjective) and name this approach as Sentiment Labelling Approach (Please refer Section 3.6). Table 3.9 showed examples of subjective and objective (non-subjective) statements.

Table 3. 10. Example of subjective and objective statement.

Statement	Subjectivity
i charge it at night and skip taking the cord with me because of the good battery life. i asked for seltzer with lime, no ice. A little too small but they are ok	Non- subjective
great laptop that offers many great features! not only was the food outstanding, but the little perks were g reat. A little stiff , but I expect that will resolve .	Subjective

4.0 Research Methodology

The complete workflow of this research project was covered in depth in this part, including the methodology. This chapter discussed a subjective classification research method using a BERT-based sequence classification. Although sentiment analysis generally assessed polarity (positive, negative, or neutral), our main focus was on identifying textual data subjectivity and objectivity. Determining whether a certain text segment was primarily objective or expressed a subjective perspective.

Figure 4.1 showed the flow diagram of the model. Initially, the data set was divided into various ratios such as 80-20, 70-30, and 60-40. Subsequently, features and targets were defined before tokenization. In this project, features constituted sentence attributes, and targets included subjective attributes and objective attributes. After data tokenization, training data was used to train the model, while test data was used to evaluate performance metrics such as accuracy, precision, recall, and F1 score.

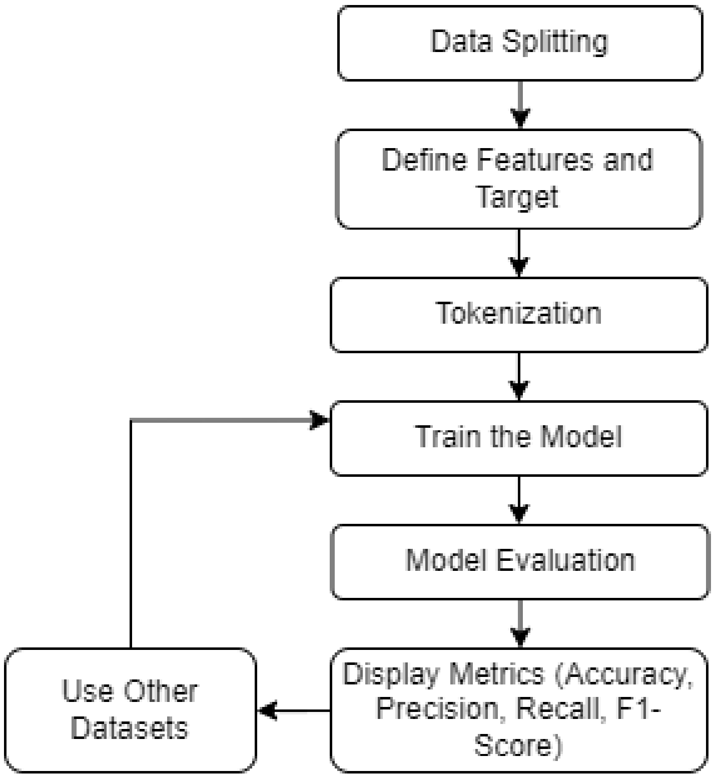


Figure 4. 1. Model Flowchart

4.1 Data Splitting

In this section, the methods used to divide the dataset into different parts for testing and training are described. Properly splitting the dataset was necessary to ensure that the model could be applied generally and to assess its performance on different datasets. The datasets were split into different ratios, with 80% for training and 20% for testing, 70% for training and 30% for testing, and 60% for training and 40% for testing. Table 4.1 and 4.2 showed the number of training and testing data shapes.

In this project, data balancing was not performed. The decision not to perform data balancing was further supported by recent research (Mohsin Iqbal, Asim Karim, Faisal Kamiran., 2019), which highlighted the limitation of accuracy for assessing the performance of sentiment predictions. The statement discussed the prevalence of imbalanced prediction errors in existing sentiment classifiers, both unsupervised and supervised, indicating that imbalanced predictions could occur naturally in sentiment analysis tasks. Therefore, applying data balancing techniques might not have significantly altered the distribution of sentiments expressed in the dataset. Given the intricate relationship between sentiment, subjectivity, and objectivity at the sentence level, attempting to balance data through techniques like SMOTE could have disrupted the genuine sentiment expressed in each sentence, ultimately undermining the accuracy of sentiment analysis. Thus, refraining from data balancing methods aligned with the need to maintain the authenticity of sentiment expressed in each sentence for reliable analysis results.

Table 4. 1. Number of subjectivity and objectivity in different train-test ratio (a)

Datasets	**Train-test split**	**Training Set**	**Testing Set**
		Subjective: Objective	**Subjective: Objective**
Sem Eval 2014 Laptop	80-20	1861:499	461:130
	70-30	1635:430	687:199
	60-40	1390:380	932:249
Sem Eval 2014 Restaurant	80-20	3122:655	771:174
	70-30	2726:579	1167:250
	60-40	2331:502	1562:327

Table 4. 2. Number of subjectivity and objectivity in different train-test ratio (b)

Datasets	**Train-test split**	**Training Set**	**Testing Set**
		Subjective: Objective	**Subjective: Objective**
MEMD Book	80-20	4054:122	1011:33
	70-30	3547:107	1518:48
	60-40	3046:86	2019:69
MEMD Clothing	80-20	2451:55	619:8
	70-30	2145:48	925:15
	60-40	1840:39	1230:24
MEMD Hotel	80-20	4480:17	1122:3
	70-30	3919:16	1683:4
	60-40	3359:14	2243:6
MEMD laptop	80-20	4930:318	1236:76
	70-30	4319:273	1847:121
	60-40	3698:238	2468:156
MEMD All Domian	80-20	18474:496	4632:111
	70-30	16164:435	6942:172
	60-40	13854:373	9252:234

4.2 Tokenization

Before the model was trained, some data preparation needed to be done. The BERT tokenizer was utilized to convert the list of sentences into a tokenized format. Tokenization involved breaking down sentences into individual tokens and converting them into numerical representations. The tokenized sentences were passed to the tokenizer with additional parameters such as padding and truncation. The output of 'encoded' contained the input IDs. After that, a PyTorch 'TensorDataset' was created from the encoded input IDs and the corresponding labels. Then, the 'Dataloader' was used to batch the data with a specified batch size ('64') and an option to shuffle the data ('True' for training the model and 'False' for testing data).

```
def convertDataloader(sentences, labels, shuffle):
    # Tokenize the sentences using the BERT tokenizer
    tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
    encoded = tokenizer(list(sentences), padding=True, truncation=True, return_tensors='pt')

    # Create DataLoader
    dataset = TensorDataset(encoded['input_ids'], torch.tensor(labels))
    dataloader = DataLoader(dataset, batch_size=64, shuffle=shuffle)

    return dataloader
```

Figure 4. 2. Data loader python

4.3 Model Training and Evaluation

4.3.1 Model Architecture

In this project, BERT for sequence classification model is chosen to address the task of subjective-objective classification, as illustrated in Figure 4.2. The selected model already incorporates a classification model, eliminating the necessity for additional classification techniques such as random forest or logistic regression. In this case, there are two classes for one attribute, with subjectivity labelled as 0 and objectivity labelled as 1 in the dataset. Consequently, the number of labels for sequence classification needed to be two.

```
model = BertForSequenceClassification.from_pretrained('bert-base-uncased', num_labels=2)
```

Figure 4. 3. BERT for sequence classification.

4.3.2 Training Procedure

The training model was designed to train a PyTorch model using a specified data loader, number of epochs, and learning rate. The AdamW optimizer, a variant of the Adam optimizer tailored for deep learning models and commonly used in natural language processing tasks was applied. The learning rate, which governs the step size during parameter updates, was set to 2e-5 (0.00005).

The number of epochs for each dataset varied, depending on how quickly the training loss reached its minimum. Table 4.3 displayed the specific number of epochs required for each dataset and Figures 4.4, 4.5, 4.6, 4.7, 4.8, 4.9 and 4.10 show the training loss curve for each dataset.

Table 4. 3. Number of Epochs for Each Dataset

Dataset	Num of Epochs	Training Loss Curve
Sem2014 Laptop	18	Figure 4.4
Sem2014 Restaurant	18	Figure 4.5
MEMD Book	33	Figure 4.6
MEMD Clothing	3	Figure 4.7
MEMD Hotel	3	Figure 4.8
MEMD Laptop	33	Figure 4.9
MEMD Restaurant	33	Figure 4.10

Figure 4.4 displayed the training loss curve for the Sem 2014 laptop datasets. In the curve, the training loss reached its minimum at around 16.5 and 17.5 epochs.

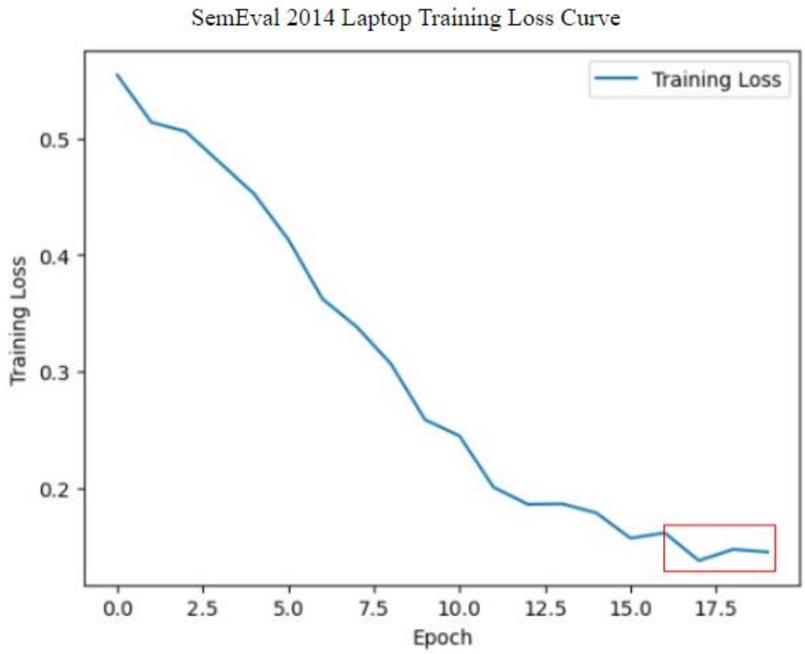


Figure 4. 4. SemEval 2014 Laptop Training loss curve

Figure 4.5 displayed the training loss curve for the Sem 2014 restaurant datasets. In the curve, the training loss reached its minimum at around 16.5 and 18 epochs and had a slight increase when reaching 19 epochs.

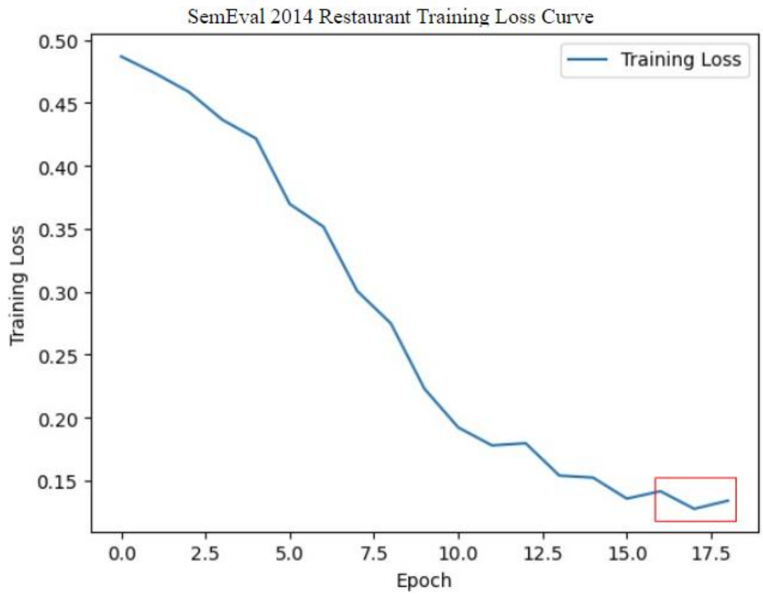


Figure 4. 5. SemEval 2014 Restaurant Training loss curve

Figure 4.6 displayed the training loss curve for the MEMD book datasets. In the curve, the training loss reached its minimum at around 33 epochs and had a slight increase when reaching to 40 epochs.

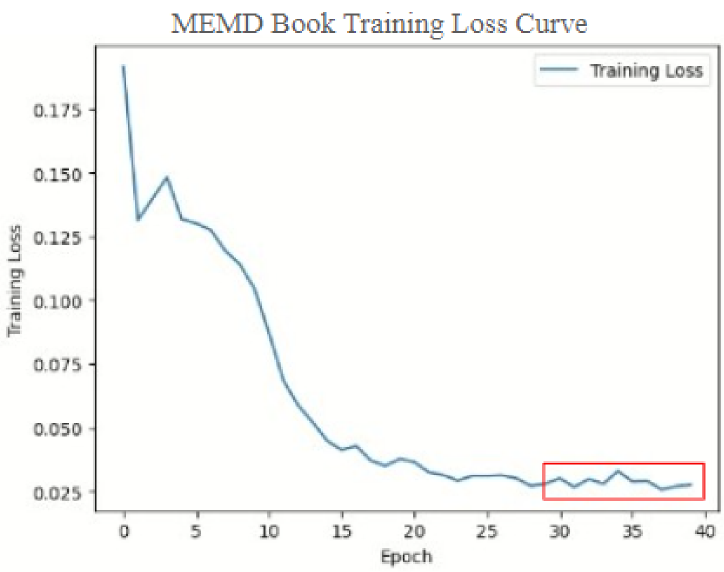


Figure 4. 6. MEMD Book Training loss curve.

Figure 4.7 displayed the training loss curve for the MEMD clothing datasets. In the curve, the training loss reached its minimum at around two epochs and increased during three epochs.

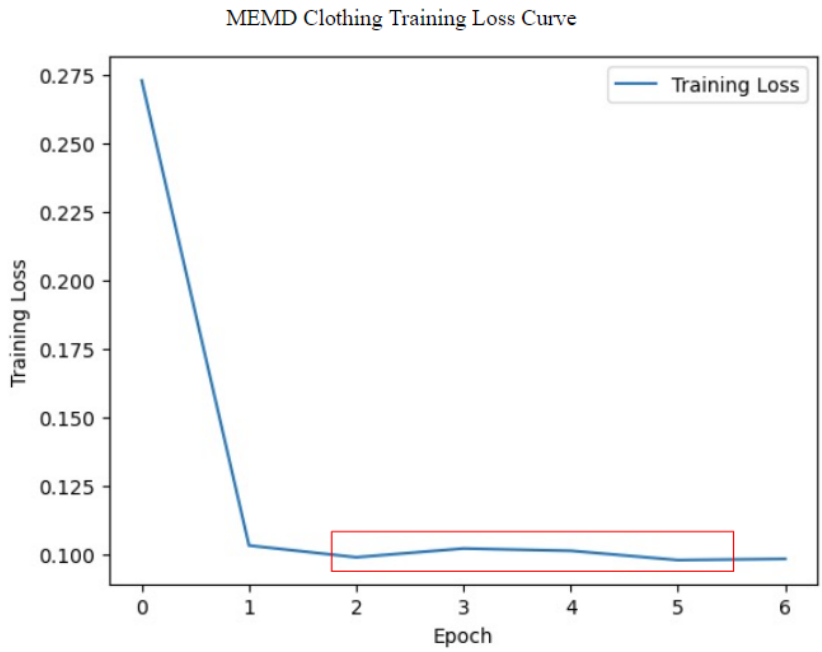


Figure 4. 7. MEMD Clothing Training loss curve

Figure 4.8 displayed the training loss curve for the MEMD hotel datasets. In the curve, the training loss reached its minimum at around three epochs and had slightly increased during four epochs.

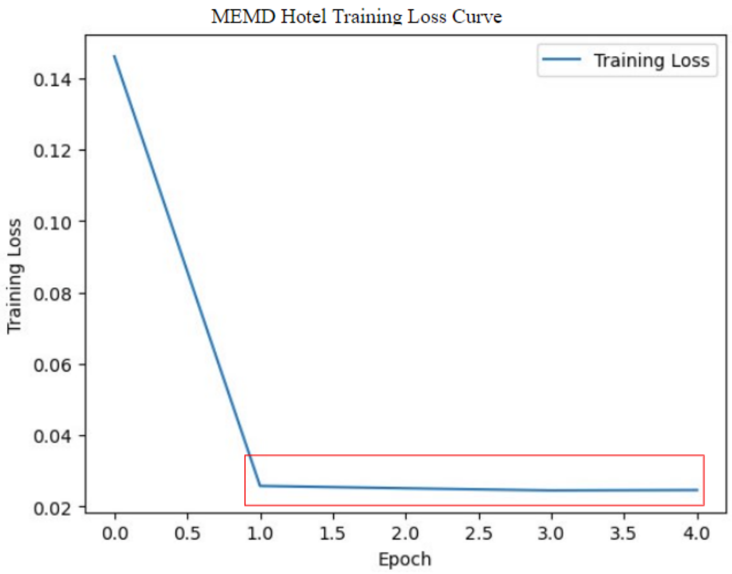


Figure 4. 8. MEMD Hotel Training loss curve

Figure 4.9 displayed the training loss curve for the MEMD hotel datasets. In the curve, the training loss reached its minimum at around 30 epochs.

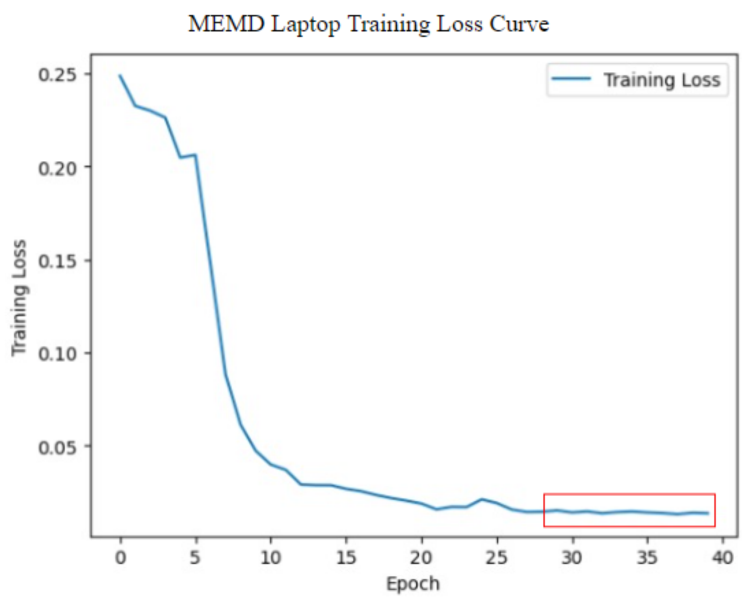


Figure 4. 9. MEMD Laptop Training loss curve

Figure 4.10 displayed the training loss curve for the MEMD restaurant datasets. In the curve, the training loss reached its minimum at around 33 epochs.

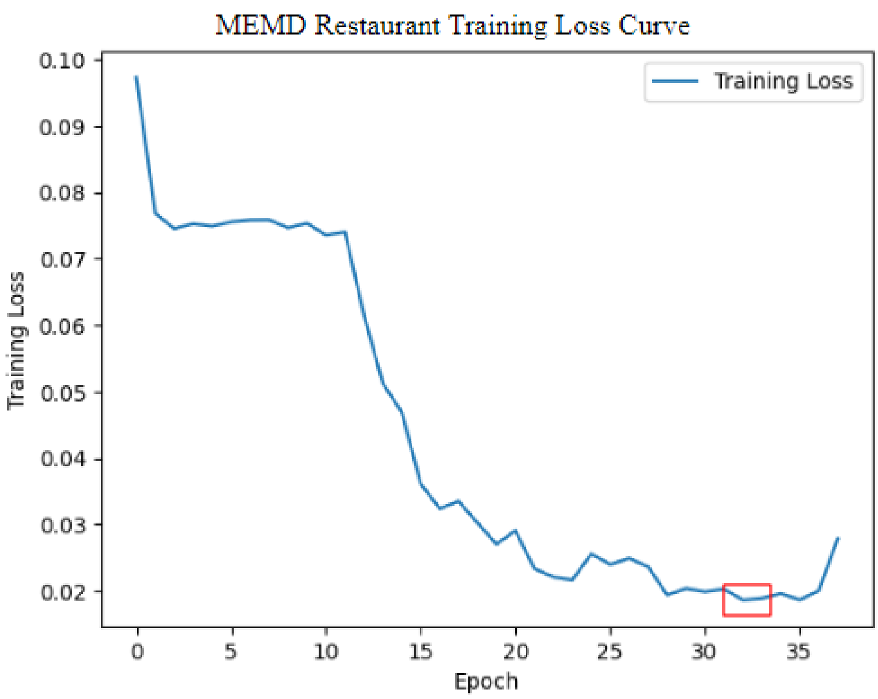


Figure 4. 10. MEMD Restaurant Training loss curve

4.3.3 Code Implementation

Figure 4.11 shows the training model, the model undergoes different epochs which show in Table 4.3, and during each epoch, the function operates in training mode. For every batch in the training data loader, the model's input and labels are transferred to the appropriate device (GPU or CPU), and the optimizer's gradients are reset to zero. The forward pass of the model is executed, leading to the computation of the loss. Subsequently, the backward pass and optimization step occur, updating the model's parameters. The total training loss for the epoch is then accumulated and averaged across all batches.

Throughout the training process, the function keeps track of the average training loss for each epoch, displaying it as a numerical value. Additionally, the functions generate a plot illustrating the training loss over the epochs for visual analysis.

```
def trainModel(model, train_dataloader, num_epochs=number, learning_rate=2e-5):
    optimizer = AdamW(model.parameters(), lr=learning_rate)

    device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
    model.to(device)

    training_loss_values = []

    for epoch in range(num_epochs):
        model.train()

        total_training_loss = 0.0

        for batch in train_dataloader:
            inputs, labels = batch
            inputs, labels = inputs.to(device), labels.to(device)
            optimizer.zero_grad()
            outputs = model(inputs, labels=labels)
            loss = outputs.loss
            loss.backward()
            optimizer.step()
            total_training_loss += loss.item()

        avg_training_loss = total_training_loss / len(train_dataloader)
        training_loss_values.append(avg_training_loss)

        print(f'Epoch {epoch + 1}/{num_epochs}, Avg. Training Loss: {avg_training_loss}')

    plt.plot(training_loss_values, label='Training Loss')
    plt.xlabel('Epoch')
    plt.ylabel('Training Loss')
    plt.legend()
    plt.show()
```

Figure 4. 11. Training model

Figure 4.12 shows the evaluation model was designed to assess the PyTorch model's performance on a specific set of data. In this project, the evaluation focused on determining key metrics, including accuracy, precision, recall, and F1 score.

To initiate the evaluation, the model is set to evaluation mode, ensuring that it operates without updating its parameters. Lists are created within the function to store predictions and losses, facilitating the subsequent analysis. The function dynamically determines the processing unit, selecting either the GPU or CPU, based on availability.

During the evaluation process, the function iterates through the test data in batches. For each batch, inputs and labels are transferred to the selected processor (GPU or CPU). The model is then run to generate predictions, capturing both the predicted values and the corresponding loss for each batch.

After evaluating, the function computes essential metrics like accuracy, F1 score, precision, and recall. These metrics give a clear picture of how well the model is doing. The function will display out these metrics to give us the information about how accurate the model is overall, how balanced it is between precision and recall, and how well it predicts positive instances. The average loss helps us understand how well the model is predicting in general.

```
def evaluateModel(model, dataloader, labels):
    model.eval()

    predictions = []
    losses = []

    device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

    with torch.no_grad():
        for batch in dataloader:
            inputs, batch_labels = batch
            inputs, batch_labels = inputs.to(device), batch_labels.to(device)
            outputs = model(inputs, labels=batch_labels)
            logits = outputs.logits
            predictions_batch = torch.argmax(logits, dim=1)
            predictions.extend(predictions_batch.cpu().numpy())
            losses.append(outputs.loss.item())

    accuracy = accuracy_score(labels, predictions)
    f1 = f1_score(labels, predictions, average='binary')
    precision = precision_score(labels, predictions, average='binary')
    recall = recall_score(labels, predictions, average='binary')
    avg_loss = np.mean(losses)

    conf_matrix = confusion_matrix(labels, predictions)

    print(f'Accuracy: {accuracy}')
    print(f'F1 Score: {f1}')
    print(f'Precision: {precision}')
    print(f'Recall: {recall}')
    print(f'Average Loss: {avg_loss}')
```

Figure 4. 12. Evaluation Model

4.3.5 Evaluation Metrics

The quality of a statistical or machine learning model was assessed using various evaluation metrics. These metrics could be categorized into types such as regression and classification. In this project focused on classification metrics, including accuracy, precision, recall, and F1-score. The formulas for these classification evaluation metrics, as provided by Turabieh (2019) , are listed below.

$$\textbf{Accuracy} = \frac{\textbf{TP+TN}}{\textbf{TP+FP+FN+TN}} \tag{4.1}$$

$$\textbf{Precision} = \frac{\textbf{TP}}{\textbf{TP+FP}} \tag{4.2}$$

$$\textbf{Recall} = \frac{\textbf{TP}}{\textbf{TP+FN}} \tag{4.3}$$

$$\textbf{F-Measure} = \frac{\mathbf{2 \times (Recall \times Precision)}}{\mathbf{(Recall+Precision)}} \tag{4.4}$$

- **True Positive (TP):** Model correctly predicts positive values.
- **True Negative (TN):** Model correctly predicts negative values.
- **False Negative (FN):** Model incorrectly predicts negative when actual is positive.
- **False Positive (FP):** Model incorrectly predicts positive when actual is negative.

4.3.6 Result and Analysis

Table 4.4 presents the evaluation metrics for the SemEval 2014 laptop dataset across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.8393), precision (0.6496), and F1-score (0.6154). The 70-30 train test split has slightly lower accuracy (0.8250), precision (0.5990), recall (0.6080), and F1-score (0.6035). The 60-40 train test split records the highest recall (0.6386) but has the lowest accuracy (0.7993), precision (0.5196), and F1-score (0.5730).

These results suggest that the 80-20 train test split is most effective for achieving high performance across most metrics, likely due to the larger training set enabling the model to learn more comprehensive patterns and features. The 70-30 train test split, while still performing well, shows a slight decline in all metrics compared to the 80-20 train test split. The 60-40 train test split, with the highest recall, indicates that the model identifies more true positive instances at the expense of precision and overall performance, potentially due to the increased diversity in the larger test set.

Table 4. 4. SemEval 2014 laptop Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.8393	0.6496	0.5923	0.6154
70 - 30	0.8250	0.5990	0.6080	0.6035
60 - 40	0.7993	0.5196	0.6386	0.5730

Table 4.5 presents the evaluation metrics for the SemEval 2014 restaurant dataset across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.8603), precision (0.6458), and F1-score (0.5849). The 70-30 train test split has slightly lower accuracy (0.8490), precision (0.5789), recall (0.5280), and F1-score (0.5523). The 60-40 train test split records the highest recall (0.6330) but has the lowest accuracy (0.8311), precision (0.5099), and F1-score (0.5648).

These results suggest that the 80-20 train test split is most effective for achieving high performance across most metrics, likely due to the larger training set enabling the model to learn more comprehensive patterns and features. The 70-30 train test split, while still performing well, shows a slight decline in all metrics compared to the 80-20 train test split. The 60-40 train test split, with the highest recall, indicates that the model identifies more true positive instances at the expense of precision and overall performance, potentially due to the increased diversity in the larger test set.

Table 4. 5. SemEval 2014 Restaurant Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.8603	0.6458	0.5345	0.5849
70 - 30	0.8490	0.5789	0.5280	0.5523
60 - 40	0.8311	0.5099	0.6330	0.5648

Table 4.6 presents the evaluation metrics for the MEMD book dataset across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.9703), precision (0.5417), recall (0.3940), and F1-score (0.4561). In contrast, both the 70-30 and 60-40 train test splits result in zero values for precision, recall, and F1-score, although they have slightly lower accuracy (0.9693 and 0.9670, respectively).

These results suggest that the 80-20 train test split is most effective for achieving high performance across all metrics. The stark difference in performance with the 70-30 and 60-40 train test splits indicates that the model struggles to identify true positives when trained on smaller portions of the dataset, leading to poor performance in terms of precision, recall, and F1-score on the test sets in these splits.

Table 4. 6. MEMD Book Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9703	0.5417	0.3940	0.4561
70 - 30	0.9693	0.0000	0.0000	0.0000
60 - 40	0.9670	0.0000	0.0000	0.0000

Table 4.7 presents the evaluation metrics for the MEMD clothing dataset across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.9872), but yields zero values for precision, recall, and F1-score. Similarly, the 70-30 and 60-40 train test splits also result in zero values for precision, recall, and F1-score, despite slightly lower accuracy (0.9840 and 0.9080, respectively).

These results suggest that while the model achieves high accuracy across all splits, it fails to correctly identify true positive instances, leading to poor performance in terms of precision, recall, and F1-score. This discrepancy highlights a limitation of the model, particularly when trained on smaller portions of the dataset, where it struggles to generalize effectively and identify relevant instances.

Table 4. 7. MEMD Clothing Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9872	0.0000	0.0000	0.0000
70 - 30	0.9840	0.0000	0.0000	0.0000
60 - 40	0.9080	0.0000	0.0000	0.0000

Table 4.8 presents the evaluation metrics for the MEMD hotel dataset across different train-test splits, showing similar results to Table 4.7. The 70-30 train test split achieves the highest accuracy (0.9976), but yields zero values for precision, recall, and F1-score. Similarly, the 80-20 and 60-40 train test splits also result in zero values for precision, recall, and F1-score, despite slightly lower accuracy (0.9973 for both splits).

These results suggest that while the model achieves high accuracy across all splits, it fails to correctly identify true positive instances, leading to poor performance in terms of precision, recall, and F1-score. This consistent pattern indicates a limitation of the model's performance, regardless of the training set size, in effectively identifying relevant instances in the MEMD hotel dataset.

Table 4. 8. MEMD Hotel Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9973	0.0000	0.0000	0.0000
70 - 30	0.9976	0.0000	0.0000	0.0000
60 - 40	0.9973	0.0000	0.0000	0.0000

Table 4.9 presents the evaluation metrics for the MEMD laptop dataset across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.9561), precision (0.7714), and F1-score (0.5684), but has a lower recall (0.4500). The 70-30 train test split achieves the highest recall (0.6446) and a balanced F1-score (0.6240) with slightly lower accuracy (0.9522). The 60-40 train test split has the lowest accuracy (0.9508), precision (0.6452), recall (0.3846), and F1-score (0.4819).

These results indicate that the 80-20 train test split, with its larger training set, allows the model to perform best overall but may miss some true positives. The 70-30 train test split, while having slightly lower accuracy, provides better recall, making it more suitable for applications prioritizing the identification of all relevant instances. The 60-40 train test split shows that a smaller training set reduces overall performance across all metrics.

Table 4. 9. MEMD Laptop Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9561	0.7714	0.4500	0.5684
70 - 30	0.9522	0.6047	0.6446	0.6240
60 - 40	0.9508	0.6452	0.3846	0.4819

Table 4.10 presents the evaluation metrics for the MEMD restaurant dataset across different train-test splits. The 70-30 train test split achieves the highest accuracy (0.9822) and precision (0.5185) but has a lower recall (0.2692) and F1-score (0.3544). The 80-20 train test split, while having slightly lower accuracy (0.9795), achieves a better balance with recall (0.4164) and F1-score (0.4000). The 60-40 train test split has the lowest recall (0.1449) and F1-score (0.2062), with an accuracy of 0.9798 and precision of 0.3571.

These results suggest that the 70-30 train test split is most effective in achieving high accuracy and precision, indicating that it is good at identifying positive instances but may miss many true positives due to its lower recall. The 80-20 train test split, with a better balance between precision and recall, provides a more consistent performance across metrics. The 60-40 train test split, with the lowest recall and F1-score, indicates that a smaller training set reduces the model's ability to generalize and identify true positives effectively.

Table 4. 10. MEMD Restaurant Dataset Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9795	0.3824	0.4164	0.4000
70 - 30	0.9822	0.5185	0.2692	0.3544
60 - 40	0.9798	0.3571	0. 1449	0.2062

Table 4.11 presents the evaluation metrics for the MEMD dataset, combining all domains, across different train-test splits. The 80-20 train test split achieves the highest accuracy (0.9720), precision (0.6686), recall (0.6163), and F1-score (0.6372). The 70-30 train test split has slightly lower accuracy (0.9714), precision (0.6547), recall (0.5828), and F1-score (0.6068). The 60-40 train test split shows the lowest accuracy (0.9707), precision (0.6587), recall (0.5676), and F1-score (0.5931).

These results suggest that the 80-20 train test split is the most effective for achieving high performance across all metrics, likely due to the larger training set providing better learning opportunities for the model. The 70-30 and 60-40 train test splits, while still performing well, show a gradual decline in all metrics, indicating that a smaller training set reduces the model's ability to generalize as effectively. Overall, the 80-20 split offers the best balance and highest performance.

Table 4. 11. MEMD Datasets using Sentiment Labelling Approach Evaluation Metrics

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9720	0.6686	0.6163	0.6372
70 - 30	0.9714	0.6547	0.5828	0.6068
60 - 40	0.9707	0.6587	0.5676	0.5931

4.3.7 Discussion

The performance differences seen across various datasets highlighted how complex subjectivity classification tasks can be and emphasized the importance of considering domain-specific factors.

The datasets contained data imbalance, as indicated in Table 4.1 and Table 4.2. However, in this project, data balancing techniques such as oversampling, undersampling, and random sampling were not performed. This decision was based on the nature of aspect-based sentiment analysis datasets, where altering the sentiment of sentences to achieve balanced data could have disrupted the genuine sentiment expressed in each sentence. Consequently, this would have undermined the accuracy of the subjectivity classification. Section 3.3 detailed the number of sentiments in each dataset.

SemEval 2014 laptop datasets (Please refer to Table 4.5) and SemEval 2014 restaurant datasets (Please refer to Table 4.5) performed well in subjectivity classification. However, in the MEMD datasets, only the laptop domain (Please refer to Table 4.9) and restaurant domain (Please refer to Table 4.10) exhibited satisfactory performance. In the other three domains, namely book, clothing, and hotel domains (Please refer to Table 4.6, Table 4.7, and Table 4.8), the precision, recall, and F1-score metrics were notably poor. This could be attributed to data imbalances, where the majority class was heavily favoured (Wen San Yee, Hu Ng, Timothy Tzen Vun Yap, Vik Tor Goh, Keng Hong Ng, Dong Theng Cher,. 2022). Table 4.2 provides clear information on the number of subjective and objective (non-subjective) instances in each MEMD domain, supporting San Yee, (2022) statement. This discrepancy suggests potential challenges in applying subjectivity classification techniques to different datasets, possibly due to differences in language usage, emotional expressions, or domain specificity.

To address the poor performance metrics of precision, recall, and F1-score in MEMD datasets, the project combined all domains in the MEMD datasets and achieved improved performance metrics (Please refer to Table 4.11). In this project, it was found that the evaluation metrics of precision, recall, and F1-score were zero when evaluated across different domains due to dataset imbalance, where one class was significantly larger than the other. However, when all domains were combined and subjectivity classification again, the results significantly improved compared to single-domain evaluations. It was observed that, although the combined dataset remained imbalanced (Please refer to Table 4.2), the increased number of instances per class was sufficient for the model to train effectively and achieve good precision, recall, and F1-score.

5.0 Datasets Labelling Approaches and Performance Evaluation

Due to encountering zero precision, recall, and F1-scores in the book, clothing, hotel and restaurant domains, the focus of the project shifted towards MEMD datasets. Drawing from the novel labelling methodologies introduced by (Samuel Pecar & Marian Simko, 2021), the subjectivity and non-subjectivity (objective) labels were re-evaluated. Figure 5.1 shows the flowchart of new labelling method.

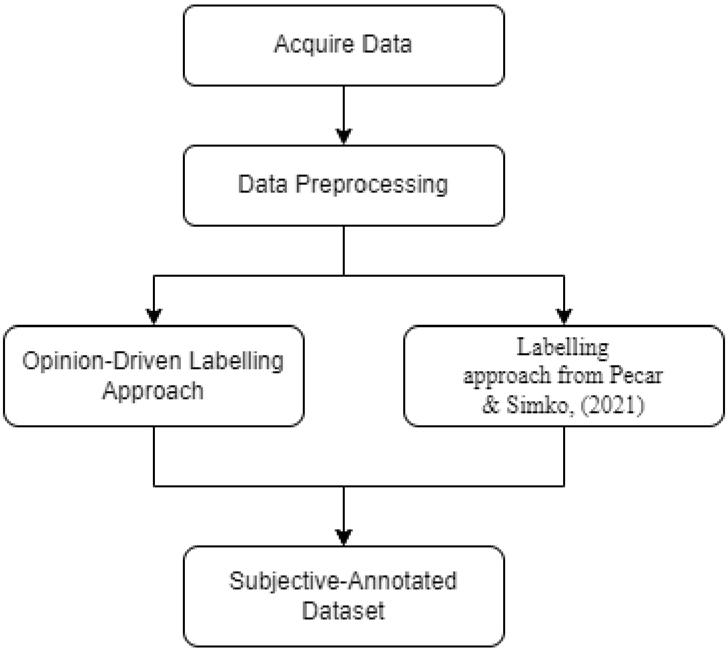


Figure 5. 1. Flowchart of new labelling method.

Before the datasets were labelled, they were reviewed again in their original JSON format. Afterward, specific attributes like aspect, opinion, category, and sentiment were extracted from the quadruples attributes. To ensure clarity and consistency, sentences containing only one aspect and one opinion were expanded. Moreover, all brackets within the datasets were eliminated to simplify the data preprocessing process. Figure 5.2 shows the process in the step of data preprocessing.

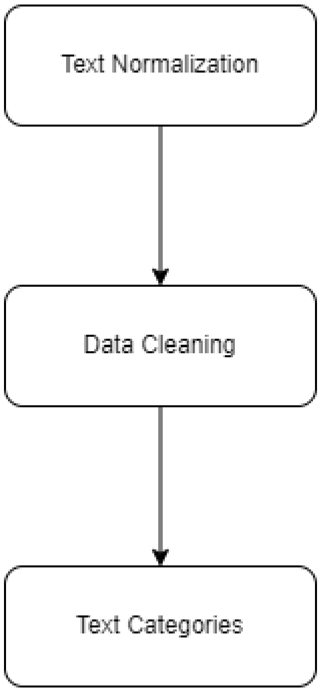


Figure 5. 2. Flowchart of Data Preparation.

5.1 Data preprocessing

In this step was used to extract aspect, opinion, category and sentiment out from quadruples attributes. Figure 5.3 shows the example of quadruples attributes in original datasets (JSON format). The aspect, opinion, category and sentiment are stored in quadruples attributes in list format.

```
root [] 292 items
  0
    raw_words "Unfortunately , it was n't up to par with her past works ."
    task "ACOS"
    quadruples [] 1 item
      0
        aspect
          from -1
          to -1
          term [] 1 item
            0 "NULL"
        category "Book#General"
        opinion
          from 0
          to 1
          term [] 1 item
        sentiment "NEG"
```


Figure 5. 3. Data Information in JSON File.

To extract out the aspect and opinion, the ‘extract_attributes’ function iterated over each quadruple in the list, accessing the 'aspect' and 'opinion' keys within each quadruple using the .get() method. If any of these keys were missing or if their corresponding values were empty dictionaries, it assigned None to the respective attribute. The extracted attributes were then appended to separate lists for aspects and opinions. Finally, the function returned these lists containing the extracted attributes. Figure 5.4 shows the extract attributes function. Figure 5.5, Figure 5.6, Figure 5.7, Figure 5.8, and Figure 5.9 display the outcome in each domain after extracting the attributes.

```python
def extract_attributes(quadruples):
    aspects = []
    opinions = []

    for quad in quadruples:
        aspect = quad.get('aspect', {}).get('term', None)
        opinion = quad.get('opinion', {}).get('term', None)

        aspects.append(aspect)
        opinions.append(opinion)

    return aspects, opinions
```

Figure 5. 4. Extract Attributes Function.

	sentence	aspect	opinion
3	This book was worth the wait , and I will keep...	[[book], [next, novel]]	[[worth], [NULL]]
4	An excellent page turner .	[[page, turner]]	[[excellent]]
5	Jodie 's tantrums range from cursing and kicki...	[[NULL], [Cathy], [book]]	[[good], [impressive], [hard]]
6	(this was really at odds with her personality...	[[NULL], [NULL]]	[[just, plain, awful], [Confusing]]
7	love these two characters , and have been foll...	[[characters], [characters], [NULL]]	[[NULL], [love], [NULL]]
...	...	...	...
287	Everything seems to be falling apart in Midkem...	[[NULL]]	[[falling, apart]]
288	boring characters , above mentioned awful edit...	[[NULL], [editing], [characters], [NULL]]	[[afraid.It], [awful], [boring], [the]]
289	First half excellent second half repetitive Co...	[[First, half], [second, half]]	[[excellent], [repetitive]]
290	I ca n't wait to read Magician 's end ! ! !	[[Magician, 's, end]]	[[NULL]]
291	There is a lot to admire , but if I were compl...	[[NULL]]	[[NULL]]

Figure 5. 5. Example of Book Domain After Perform Extract Attributes Function.

	sentence	aspect	opinion
0	I probably could have ordered a C but I wanted...	[[NULL]]	[[NULL]]
1	I expected the waistband to be soft as well !	[[waistband]]	[[soft]]
2	Nope , returned for damaged 13D and still tigh...	[[NULL]]	[[still, tight]]
3	I ordered the Kelly green & Steel polo shirts ...	[[condition]]	[[good]]
4	This was a gift for my 13 year old granddaught...	[[quality], [men, 's, medium]]	[[nice], [perfectly]]
...	...	...	...
228	I definitely recommend them .	[[them]]	[[definitely, recommend]]
229	must be seconds quality because they are way s...	[[quality]]	[[weird]]
230	Beats pantyhose hands down , and they 're DURA...	[[they]]	[[DURABLE]]
231	Did not hurt their feet !	[[NULL]]	[[NULL]]
232	I throw in the washing marchine and still have...	[[NULL]]	[[NULL]]

Figure 5. 6. Example of Clothing Domain After Perform Extract Attributes Function.

	sentence	aspect	opinion
3	Cozy bed , and good shower !	[[shower], [bed]]	[[good], [Cozy]]
4	Excellent host .	[[host]]	[[Excellent]]
5	Awesome hosts , very comfortable and clean room .	[[room], [hosts], [room]]	[[very, comfortable], [Awesome], [clean]]
6	The unit was as described and the location ca ...	[[location]]	[[NULL]]
7	She was very helpful and the flat is well prep...	[[NULL], [flat]]	[[very, helpful], [well]]
...	...	...	...
344	We have recommended this location to over 100 ...	[[location], [amenities]]	[[recommended], [wonderful]]
345	We were so comfortable for the weekend !	[[NULL]]	[[so, comfortable]]
346	Shockingly great location , right in the heart...	[[location], [location]]	[[Shockingly, great], [NULL]]
347	The room is quite and cozy .	[[room], [room]]	[[cozy], [quite]]
348	Our stay at Tiffany 's apartment was a good one .	[[Tiffany, 's, apartment]]	[[good]]

Figure 5. 7. Example of Hotel Domain After Perform Extract Attributes Function.

	sentence	aspect	opinion
3	- though the case is plastic , the keyboard ar...	[[keyboard, area], [case]]	[[NULL], [NULL]]
4	very poor battery life .	[[battery, life]]	[[poor]]
5	don ' t buy it .	[[NULL]]	[[NULL]]
6	i opened the box after ordering on black frida...	[[machine]]	[[NULL]]
7	biggest gripe , no backlights on the keyboard .	[[keyboard]]	[[NULL]]
...	...	...	...
400	really happy with this laptop !	[[laptop]]	[[happy]]
401	the unit is sturdy and the screen is excellent .	[[unit], [screen]]	[[sturdy], [excellent]]
402	battery life is great if you use half of the b...	[[battery, life]]	[[great]]
403	greatest thing i ' ve bought myself in a long ...	[[NULL]]	[[greatest]]
404	- pretty loud speakers .	[[speakers]]	[[loud]]

Figure 5. 8. Example of Laptop Domain After Perform Extract Attributes Function.

	sentence	aspect	opinion
0	I had a fantastic experience at Belly Wine Bar .	[[Belly, Wine, Bar]]	[[fantastic]]
1	Olive is a pretty cute little place , offering...	[[Olive]]	[[pretty, cute]]
2	FINALLY got around to trying this place since ...	[[place]]	[[well, worth, it]]
3	The hostess looked at us confused , because ou...	[[hostess]]	[[NULL]]
4	Finally we order , and then wait , and wait , ...	[[NULL]]	[[NULL]]
...	...	...	...
505	Will return enjoyed a order of 20 wings 911 fl...	[[wings]]	[[enjoyed]]
506	Definitely not worth it and will likely never ...	[[place], [place]]	[[NULL], [Definitely, not, worth]]
507	After initial orders for beverages , our waitr...	[[waitress]]	[[NULL]]
508	We had a pretty big table and everyone seemed ...	[[food], [table]]	[[love], [pretty, big]]
509	I ordered a side of dipper fries as well , but...	[[dipper, fries]]	[[gross]]

Figure 5. 9. Example of Restaurant Domain After Perform Extract Attributes Function.

5.12 Text Normalization

The text normalization step was used to expand the sentence mentioned several aspects and corresponding opinions. In response the function ‘expand_sentences’ was designed to handle situations where a single sentence contained one or more aspects and opinions. By going through each row of the original dataset, it extracted the sentences, aspects, and opinions. Then, it created a new DataFrame for each row, repeating the sentences for each aspect-opinion pair identified in the original row. This process guaranteed that each aspect-opinion pair had its own row, which aided in further analysis or processing. Figure 5.7 display the outcome after expanding the sentences. Figure 5.6 shows the expand sentences function. Figure 5.11, Figure 5.12, Figure 5.13, Figure 5.14, and Figure 5.15 display the outcome after expanding the sentences. Table 5.1 shows the number of sentences after expanding.

```python
def expand_sentences(datasets):
    new_datasets = []

    # Iterate through each row of the original DataFrame
    for index, row in df.iterrows():
        raw_words = row['sentence']
        aspects = row['aspect']
        opinions = row['opinion']

        # Create a DataFrame for the current row's aspect-opinion pairs
        row_df = pd.DataFrame({'raw_words': [raw_words] * len(aspects),
                               'aspect': aspects,
                               'opinion': opinions})

        # Append the DataFrame to the list
        new_datasets.append(row_df)
```

Figure 5. 10. Expand Sentences Function.

	sentence	aspect	opinion
2	A very good read from the author who saw it al...	[read]	[very, good]
3	This book was worth the wait , and I will keep...	[book]	[worth]
4	This book was worth the wait , and I will keep...	[next, novel]	[NULL]
5	An excellent page turner .	[page, turner]	[excellent]
6	Jodie 's tantrums range from cursing and kicki...	[NULL]	[good]
...	...	...	...
419	boring characters , above mentioned awful edit...	[NULL]	[the]
420	First half excellent second half repetitive Co...	[First, half]	[excellent]
421	First half excellent second half repetitive Co...	[second, half]	[repetitive]
422	I ca n't wait to read Magician 's end ! ! !	[Magician, 's, end]	[NULL]
423	There is a lot to admire , but if I were compl...	[NULL]	[NULL]

Figure 5. 11. Example of Book Domain After Perform Expand Sentences Function.

	sentence	aspect	opinion
0	I probably could have ordered a C but I wanted...	[NULL]	[NULL]
1	I expected the waistband to be soft as well !	[waistband]	[soft]
2	Nope , returned for damaged 13D and still tigh...	[NULL]	[still, tight]
3	I ordered the Kelly green & Steel polo shirts ...	[condition]	[good]
4	This was a gift for my 13 year old granddaught...	[quality]	[nice]
...	...	...	...
330	I definitely recommend them .	[them]	[definitely, recommend]
331	must be seconds quality because they are way s...	[quality]	[weird]
332	Beats pantyhose hands down , and they 're DURA...	[they]	[DURABLE]
333	Did not hurt their feet !	[NULL]	[NULL]
334	I throw in the washing marchine and still have...	[NULL]	[NULL]

Figure 5. 12. Example of Clothing Domain After Perform Expand Sentences Function.

	sentence	aspect	opinion
2	A wonderful location with great amenities and ...	[location]	[wonderful]
3	You probably would n't find a better location ...	[location]	[better]
4	Thank you !	[NULL]	[Thank]
5	Cozy bed , and good shower !	[shower]	[good]
6	Cozy bed , and good shower !	[bed]	[Cozy]
...	...	...	...
597	Shockingly great location , right in the heart...	[location]	[Shockingly, great]
598	Shockingly great location , right in the heart...	[location]	[NULL]
599	The room is quite and cozy .	[room]	[cozy]
600	The room is quite and cozy .	[room]	[quite]
601	Our stay at Tiffany 's apartment was a good one .	[Tiffany, 's, apartment]	[good]

Figure 5. 13. Example of Hotel Domain After Perform Expand Sentences Function.

	sentence	aspect	opinion
3	- though the case is plastic , the keyboard ar...	[keyboard, area]	[NULL]
4	- though the case is plastic , the keyboard ar...	[case]	[NULL]
5	very poor battery life .	[battery, life]	[poor]
6	don ' t buy it .	[NULL]	[NULL]
7	i opened the box after ordering on black frida...	[machine]	[NULL]
...	...	...	...
555	the unit is sturdy and the screen is excellent .	[unit]	[sturdy]
556	the unit is sturdy and the screen is excellent .	[screen]	[excellent]
557	battery life is great if you use half of the b...	[battery, life]	[great]
558	greatest thing i ' ve bought myself in a long ...	[NULL]	[greatest]
559	- pretty loud speakers .	[speakers]	[loud]

Figure 5. 14. Example of Laptop Domain After Perform Expand Sentences Function.

	raw_words	aspect	opinion
0	The apps are merely shortcuts on the desktop .	[apps]	[NULL]
1	It is extremely light weight .	[NULL]	[extremely, light]
2	Thanks , MSI	[MSI]	[NULL]
3	even though I do n't have a lot of programs in...	[NULL]	[NULL]
4	Overall the laptop is good , although I added ...	[laptop]	[good]
...	...	...	...
1070	I ordered this straight from Apple , since I s...	[NULL]	[NULL]
1071	This is a very slow laptop .	[laptop]	[very, slow]
1072	It 's been a year since I bought them , and I ...	[NULL]	[NULL]
1073	The only downside is the fans are a bit loud .	[fans]	[a, bit, loud]
1074	Its slow .	[NULL]	[slow]

Figure 5. 15. Example of Restaurant Domain After Perform Expand Sentences Function.

Table 5. 1. Number of Sentences Before and After Expanding.

Datasets	Sentence before expanding	Sentence after expanding
MEMD Book	2967	4507
MEMD Clothing	2373	3416
MEMD Hotel	3526	6017
MEMD Laptop	4086	5690
MEMD Restaurant	5152	8496

5.1.3 Data Cleaning

This step was to remove the brackets from the aspect and opinion attributes and replace commas with spaces. In the function 'remove_Bracket', it operated on the dataset provided as input. Specifically, for each row in the dataset, it removed the brackets surrounding the aspect and opinion attributes using a lambda function with the 'apply' method. Then, it replaced commas with spaces. This ensured that the aspect and opinion attributes were formatted suitably for additional analysis or presentation. Figure 5.8 shows the remove bracket function. Figure 5.17, Figure 5.18, Figure 5.19 Figure 5.20, and Figure 5.21 displays the outcome after remove bracket and comma in the aspect and opinion attributes.

```
def remove_Bracket(dataset):
    dataset['aspect'] = dataset['aspect'].apply(lambda x: ' '.join(map(str, x))).str.strip('[]')
    dataset['opinion'] = dataset['opinion'].apply(lambda x: ' '.join(map(str, x))).str.strip('[]')
```

Figure 5. 16. Remove Bracket Function.

	sentence	aspect	opinion
2	A very good read from the author who saw it al...	read	very good
3	This book was worth the wait , and I will keep...	book	worth
4	This book was worth the wait , and I will keep...	next novel	NULL
5	An excellent page turner .	page turner	excellent
6	Jodie 's tantrums range from cursing and kicki...	NULL	good
...	...	...	...
419	boring characters , above mentioned awful edit...	NULL	the
420	First half excellent second half repetitive Co...	First half	excellent
421	First half excellent second half repetitive Co...	second half	repetitive
422	I ca n't wait to read Magician 's end ! ! !	Magician 's end	NULL
423	There is a lot to admire , but if I were compl...	NULL	NULL

Figure 5. 17. Example of Book Domain After Perform Remove Bracket Function.

	sentence	aspect	opinion
2	Nope , returned for damaged 13D and still tigh...	NULL	still tight
3	I ordered the Kelly green & Steel polo shirts ...	condition	good
4	This was a gift for my 13 year old granddaught...	quality	nice
5	This was a gift for my 13 year old granddaught...	men 's medium	perfectly
6	I very fond of the style but I 'm so sad they ...	they	so sad
...	...	...	...
330	I definitely recommend them .	them	definitely recommend
331	must be seconds quality because they are way s...	quality	weird
332	Beats pantyhose hands down , and they 're DURA...	they	DURABLE
333	Did not hurt their feet !	NULL	NULL
334	I throw in the washing marchine and still have...	NULL	NULL

Figure 5. 18. Example of Clothing Domain After Perform Remove Bracket Function.

	sentence	aspect	opinion
2	A wonderful location with great amenities and ...	location	wonderful
3	You probably would n't find a better location ...	location	better
4	Thank you !	NULL	Thank
5	Cozy bed , and good shower !	shower	good
6	Cozy bed , and good shower !	bed	Cozy
...	...	...	...
597	Shockingly great location , right in the heart...	location	Shockingly great
598	Shockingly great location , right in the heart...	location	NULL
599	The room is quite and cozy .	room	cozy
600	The room is quite and cozy .	room	quite
601	Our stay at Tiffany 's apartment was a good one .	Tiffany 's apartment	good

Figure 5. 19. Example of Hotel Domain After Perform Remove Bracket Function.

	sentence	aspect	opinion
3	- though the case is plastic , the keyboard ar...	keyboard area	NULL
4	- though the case is plastic , the keyboard ar...	case	NULL
5	very poor battery life .	battery life	poor
6	don ' t buy it .	NULL	NULL
7	i opened the box after ordering on black frida...	machine	NULL
...	...	...	...
555	the unit is sturdy and the screen is excellent .	unit	sturdy
556	the unit is sturdy and the screen is excellent .	screen	excellent
557	battery life is great if you use half of the b...	battery life	great
558	greatest thing i ' ve bought myself in a long ...	NULL	greatest
559	- pretty loud speakers .	speakers	loud

Figure 5. 20. Example of Laptop Domain After Perform Remove Bracket Function.

	sentence	aspect	opinion
2	FINALLY got around to trying this place since ...	place	well worth it
3	The hostess looked at us confused , because ou...	hostess	NULL
4	Finally we order , and then wait , and wait , ...	NULL	NULL
5	They brought us out some complimentary hummus ...	complimentary hummus	NULL
6	I love their baked beans - they 're rich and f...	baked beans	rich
...	...	...	...
800	Definitely not worth it and will likely never ...	place	Definitely not worth
801	After initial orders for beverages , our waitr...	waitress	NULL
802	We had a pretty big table and everyone seemed ...	food	love
803	We had a pretty big table and everyone seemed ...	table	pretty big
804	I ordered a side of dipper fries as well , but...	dipper fries	gross

Figure 5. 21. Example of Restaurant Domain After Perform Remove Bracket Function.

5.1.4 Text Categories

This step was to group the sentences that has the same aspect. The Python function 'group_same_aspect' was used to group sentences with the same aspect. First, it checked for any missing values in the 'opinion' and 'aspect' columns, replacing them with 'NULL'. Next, it grouped the dataset by the sentence and 'aspect' columns, combining the related 'opinion' values. This resulted in a new DataFrame where sentences were grouped by aspect, making it suitable for additional analysis. Figure 5.10 shows the text categories function. Figure 5.23, Figure 5.24, Figure 5.25, Figure 5.26, and Figure 5.27 displays the result after group the sentences with the same aspect.

```python
def text_categories(datasets):
    datasets['opinion'].fillna('NULL', inplace=True)
    datasets['aspect'].fillna('NULL', inplace=True)

    grouped_df = datasets.groupby(['sentence', 'aspect'])['opinion'].apply(', '.join).reset_index()

    return grouped_df
```

Figure 5. 22. Text Categories Function.

	sentence	aspect	opinion
11	) As one might expect of a book that covers so...	book	not thin
12	) As one might expect of a book that covers so...	book	NaN

	sentence	aspect	opinion
10	) As one might expect of a book that covers so...	book	not thin, NULL

Figure 5. 23. Example of Book Domain After Perform Text Categories Function.

	sentence	aspect	opinion
18	A good light sweat shirt , washes and wears we...	sweat shirt	good
19	A good light sweat shirt , washes and wears we...	sweat shirt	light
20	A good light sweat shirt , washes and wears we...	sweat shirt	well

	sentence	aspect	opinion
16	A good light sweat shirt , washes and wears we...	sweat shirt	good, light, well

Figure 5. 24. Example of Clothing Domain After Perform Text Categories Function.

	sentence	aspect	opinion
0	A wonderful location with great amenities and ...	amenities	great
1	A wonderful location with great amenities and ...	location	very clean
2	A wonderful location with great amenities and ...	location	wonderful

	sentence	aspect	opinion
58	A wonderful location with great amenities and ...	amenities	great
59	A wonderful location with great amenities and ...	location	very clean, wonderful

Figure 5. 25. Example of Hotel Domain After Perform Text Categories Function.

	sentence	aspect	opinion
8	**Update to this review- still wearing this wa...	watch	great
9	**Update to this review- still wearing this wa...	watch	NaN
10	**Update to this review- still wearing this wa...	watch	great

	sentence	aspect	opinion
6	**Update to this review- still wearing this wa...	watch	great, NULL, great

Figure 5. 26. Example of Laptop Domain After Perform Text Categories Function.

	sentence	aspect	opinion
14	! the tandoori chicken was bland bland bland a...	tandoori chicken	bland
15	! the tandoori chicken was bland bland bland a...	tandoori chicken	bland
16	! the tandoori chicken was bland bland bland a...	naan	NaN
17	! the tandoori chicken was bland bland bland a...	tandoori chicken	bland

	sentence	aspect	opinion
9	! the tandoori chicken was bland bland bland a...	naan	NULL
10	! the tandoori chicken was bland bland bland a...	tandoori chicken	bland, bland, bland

Figure 5. 27. Example of Restaurant Domain After Perform Text Categories Function.

5.2 Data Labelling for Additional Datasets

In this step, the datasets were relabelled using the approach introduced by (Pecar & Simko, 2021) which is opinion-driven labelling approach and the aspect and opinion-based labelling approach.

Opinion-driven labelling methods using manual labelling involve manually checking each sentence for the presence of a clear opinion. Sentences containing explicitly stated opinions were marked as subjective, while sentences lacking such opinions were classified as non-subjective (Objective). Table 5.2 shows the example of Opinion-driven labelling methods.

Table 5. 2. Example of Opinion-driven labelling methods.

Sentence	Aspect	Opinion	Subjectivity
A book that keeps you interested and thirsty for more.	book	Null	Subjective
Although these are n't perfect , they are VERY comfortable and I 'll be purchasing them again when my current pair is torn up .	them	Null	Subjective
Absolutely love the color of this hoodie !	color	Absolutely love	Subjective
After taking a look at the size chart on the packaging , I didn't even bother to try these on and instead returned them immediately .	them	Null	Non-Subjective
And the color is exactly as pictured .	color	Null	Non-Subjective

Another method is taken from Pecar & Simko (2021). They identify different category in opinion such as miscellaneous, opinionated sentence, suggestion to customers, suggestion to management, expectation, and unrelated opinion. In this project follow the method where all types of opinions, except for miscellaneous ones, are considered subjective. Only sentences categorized as miscellaneous are regarded as non-subjective. The miscellaneous represents sentence containing only factual information, with no expressed opinions included.

Table 5. 3 Labelling Approach From Pecar & Simko (2021).

Sentence	Aspect	Opinion	Subjectivity
A book that keeps you interested and thirsty for more.	book	Null	Non-Subjective
Although these are n't perfect , they are VERY comfortable and I 'll be purchasing them again when my current pair is torn up .	them	Null	Non-Subjective
Absolutely love the color of this hoodie !	color	Absolutely love	Subjective
After taking a look at the size chart on the packaging , I didn't even bother to try these on and instead returned them immediately .	them	Null	Non-Subjective
And the color is exactly as pictured .	color	Null	Non-Subjective

5.3 EDA

In the opinion-driven labelling method within the book domain, the subjectivity attributes consisted of 2203 instances of subjective, and 496 instances of non-subjective (objectivity). For the Labelling method from (Pecar & Simko, 2021), subjectivity attributes consisted of 3759 instances of subjectivity and 1461 instances of non-subjective (objectivity). Figures 5.28 shows the book domain subjectivity attribute distribution using opinion-driven labelling method. 5.29 shows book domain subjectivity attribute distribution using aspect and opinion-based labelling methods.

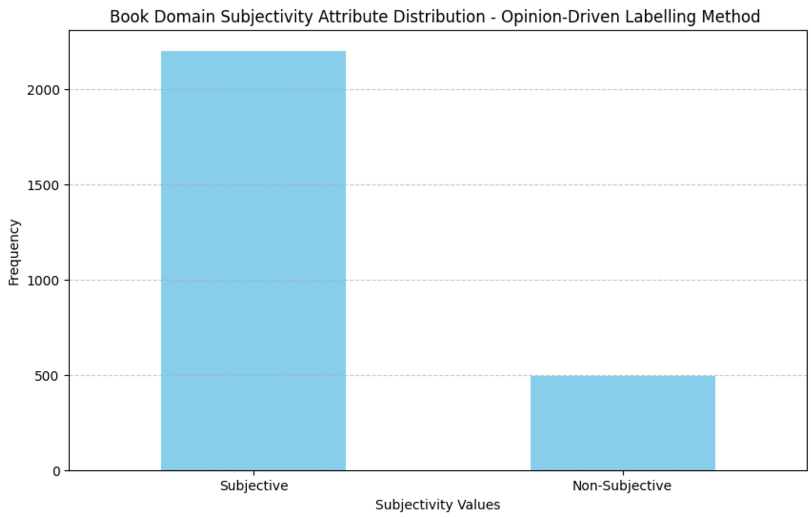


Figure 5. 28. Book Domain Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

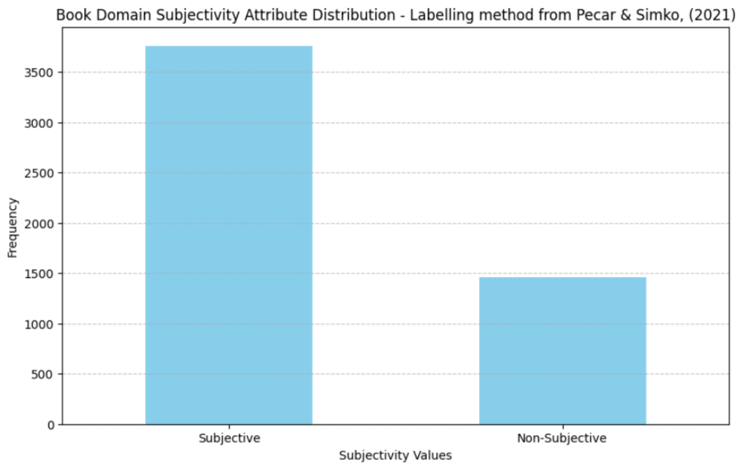


Figure 5. 29. Book Domain Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

In the opinion-driven labelling method within the clothing domain, the subjectivity attributes consisted of 1651 instances of subjective, and 285 instances of non-subjective (objectivity). For the aspect and opinion-based labelling, subjectivity attributes consisted of 2477 instances of subjectivity and 659 instances of non-subjective (objectivity). Figures 5.30 shows the clothing domain subjectivity attribute distribution using opinion-driven labelling method. 5.31 shows clothing domain subjectivity attribute distribution using aspect and opinion-based labelling methods.

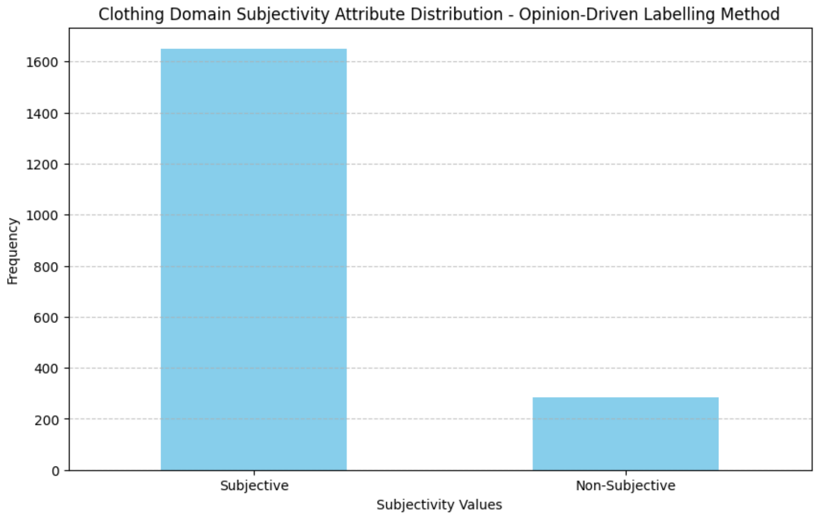


Figure 5. 30. Clothing Domain Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

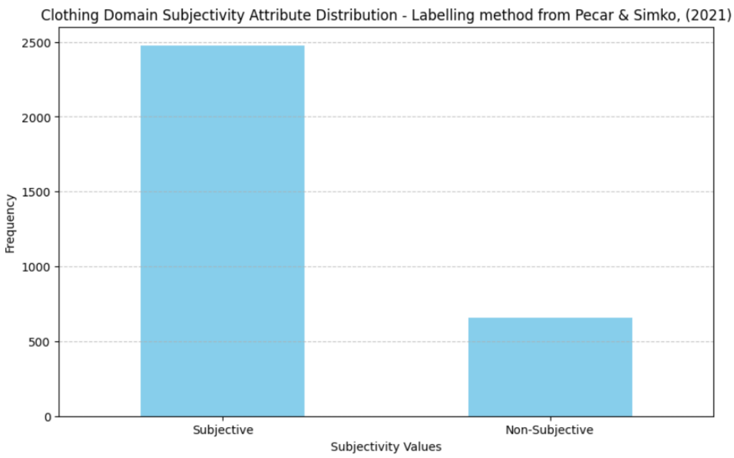


Figure 5. 31. Clothing Domain Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

In the opinion-driven labelling method within the hotel domain, the subjectivity attributes consisted of 3439 instances of subjective, and 196 instances of non-subjective (objectivity). For the aspect and opinion-based labelling, subjectivity attributes consisted of 5003 instances of subjectivity and 619 instances of non-subjective (objectivity). Figures 5.32 shows the hotel domain subjectivity attribute distribution using opinion-driven labelling method. 5.33 shows hotel domain subjectivity attribute distribution using aspect and opinion-based labelling methods.

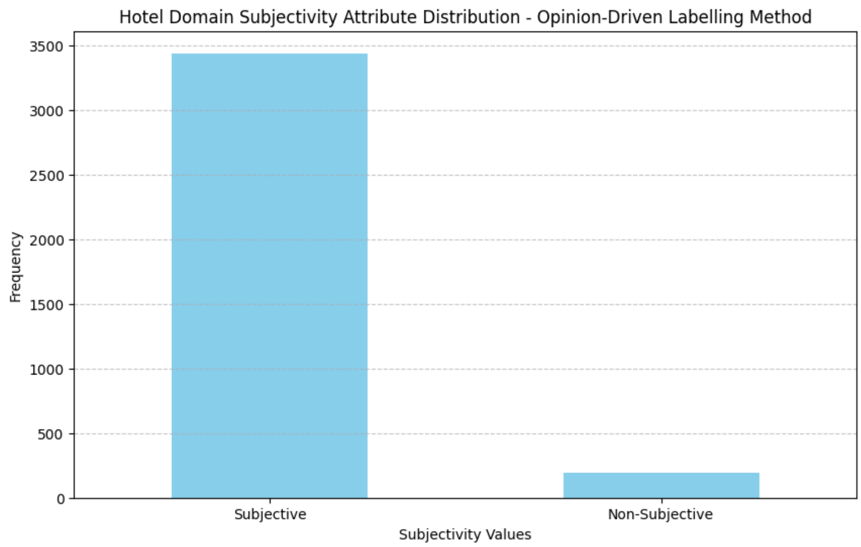


Figure 5. 32. Hotel Domain Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

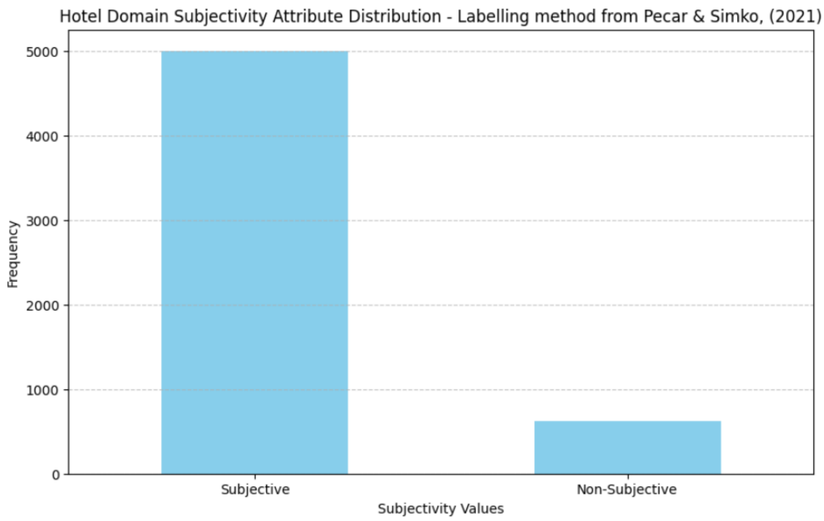


Figure 5. 33. Hotel Domain Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

In the opinion-driven labelling method within the laptop domain, the subjectivity attributes consisted of 2799 instances of subjective, and 1013 instances of non-subjective (objectivity). For the aspect and opinion-based labelling, subjectivity attributes consisted of 4445 instances of subjectivity and 2109 instances of non-subjective (objectivity). Figures 5.34 shows the laptop domain subjectivity attribute distribution using opinion-driven labelling method. 5.35 shows laptop domain subjectivity attribute distribution using aspect and opinion-based labelling methods.

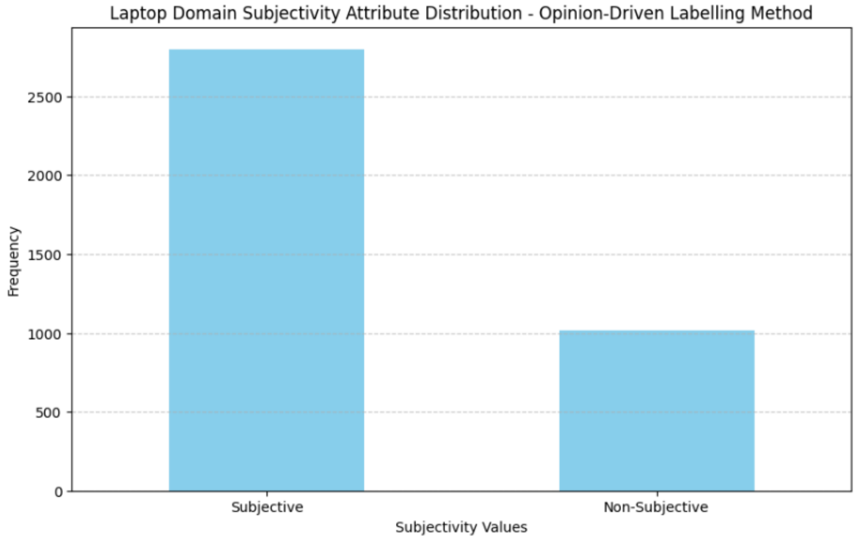


Figure 5. 34 Laptop Domain Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

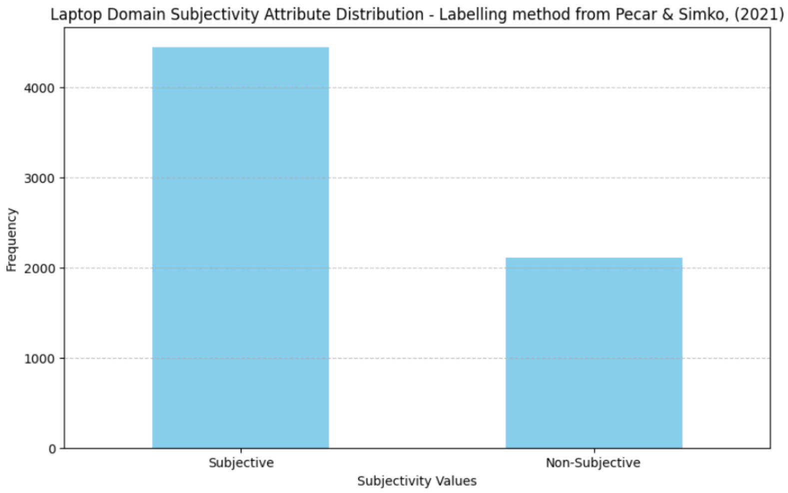


Figure 5. 35. Laptop Domain Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

In the opinion-driven labelling method within the restaurant domain, the subjectivity attributes consisted of 5085 instances of subjective, and 399 instances of non-subjective (objectivity). For the aspect and opinion-based labelling, subjectivity attributes consisted of 8166 instances of subjectivity and 1374 instances of non-subjective (objectivity). Figures 5.36 shows the restaurant domain subjectivity attribute distribution using opinion-driven labelling method. 5.37 shows restaurant domain subjectivity attribute distribution using aspect and opinion-based labelling methods.

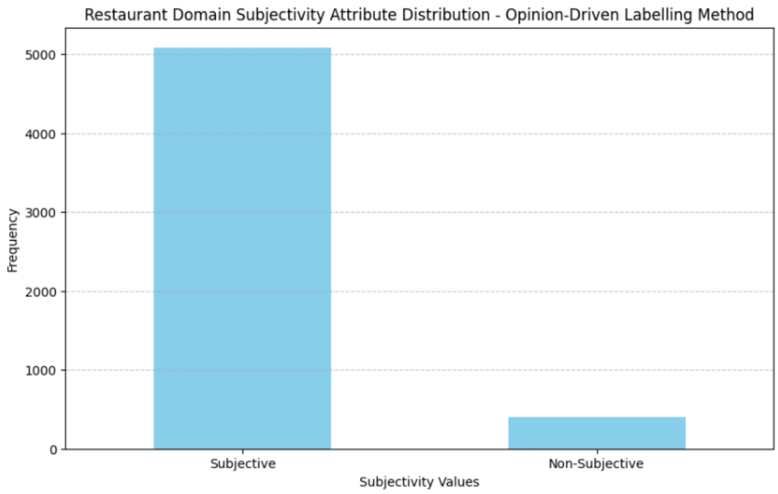


Figure 5. 36. Restaurant Domain Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

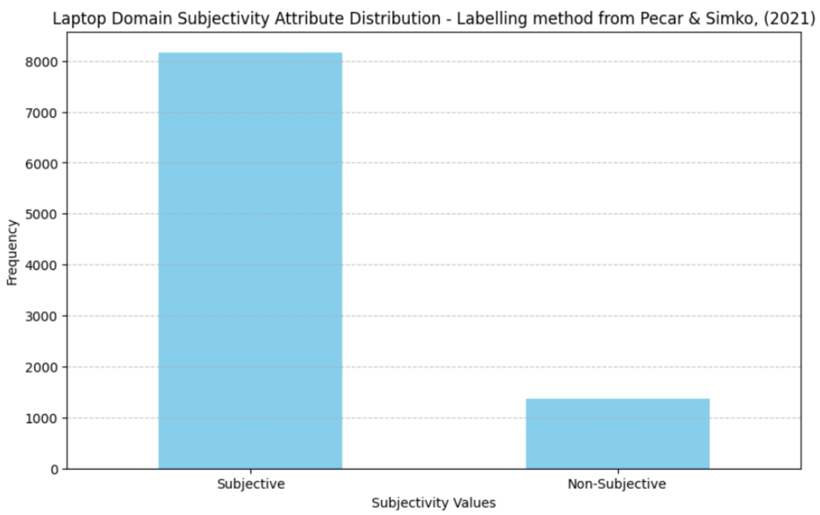


Figure 5. 37. Restaurant Domain Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

In the opinion-driven labelling method across all the domain, the subjectivity attributes consisted of 15177 instances of subjective, and 2389 instances of non-subjective (objectivity). For the labelling method from (Pecar & Simko, 2021), subjectivity attributes consisted of 18798 instances of subjectivity and 4915 instances of non-subjective (objectivity). For the sentiment labelling, subjectivity attributes consisted of 23106 instances of subjectivity and 607 instances of non-subjective (objectivity). The sentiment labelling was set positive and negative to subjective and set neutral to non-subjective (objective). Figures 5.38 shows MEMD datasets subjectivity attribute distribution using opinion-driven labelling method. 5.39 shows MEMD datasets subjectivity attribute distribution using for the labelling method from (Pecar & Simko, 2021). 5.40 shows MEMD datasets subjectivity attribute distribution using sentiment labelling methods.

As mentioned in section 4.1, the data balancing was not performed because the data balancing method like SMOTE could have disrupted the genuine sentiment expressed in each sentence, ultimately undermining the accuracy.

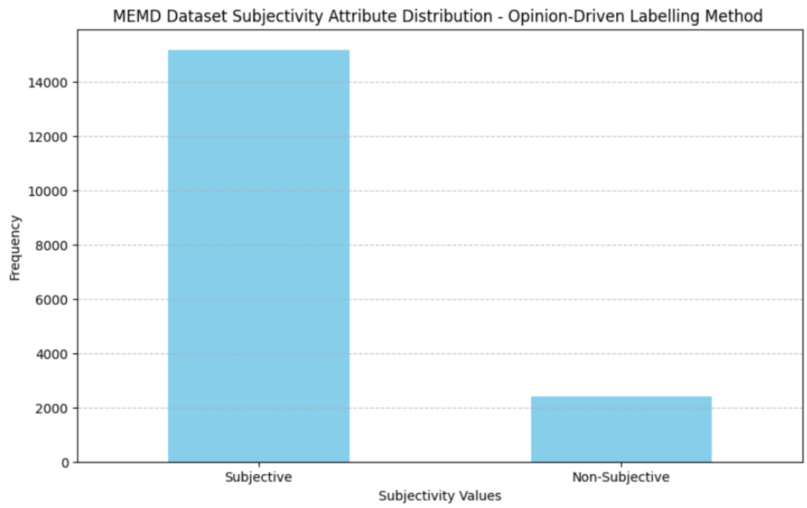


Figure 5. 38. MEMD Dataset Subjectivity Attribute Distribution - Opinion-Driven Labelling Method.

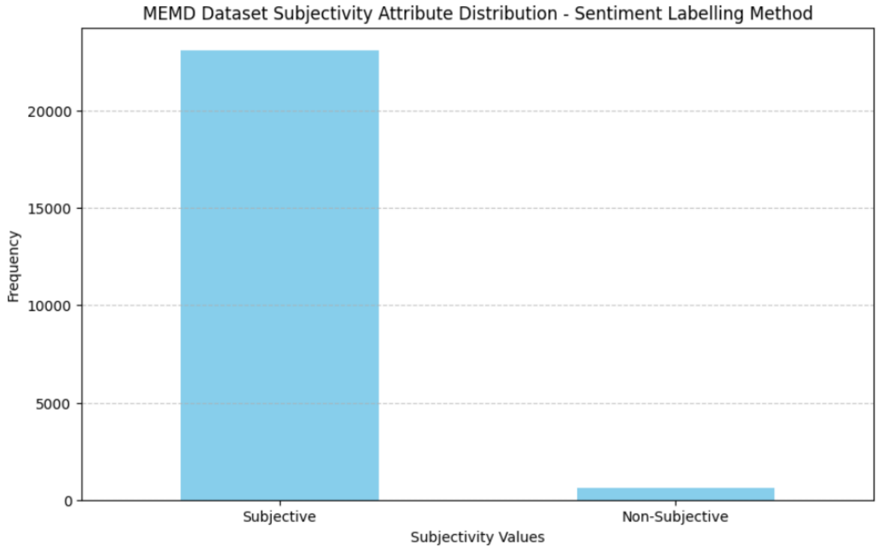


Figure 5. 39. MEMD Dataset Subjectivity Attribute Distribution - Sentiment Labelling Method.

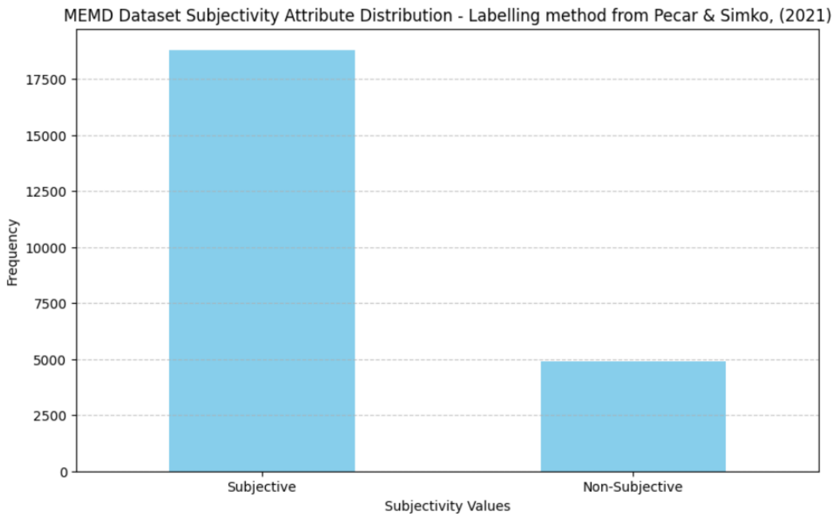


Figure 5. 40. MEMD Datasets Subjectivity Attribute Distribution - Labelling method from Pecar & Simko (2021).

5.2 Data Splitting

The MEMD datasets were split into different ratios, with 80% for training and 20% for testing, 70% for training and 30% for testing, and 60% for training and 40% for testing. Table 5.4 showed the number of training and testing data shapes.

In this project, data balancing was not performed. The decision not to perform data balancing was further supported by recent research (Iqbal et al., 2019). Given the intricate relationship between sentiment, subjectivity, and objectivity at the sentence level, attempting to balance data through techniques like SMOTE could have disrupted the genuine sentiment expressed in each sentence, ultimately undermining the accuracy of sentiment analysis. Thus, refraining from data balancing methods aligned with the need to maintain the authenticity of expressed in each sentence for reliable analysis results.

Table 5. 4. Number of subjectivity and objectivity in MEMD datasets in different train-test ratio

Approach	**Train-test split**	**Training Set**	**Testing Set**
		Subjective: Objective	**Subjective: Objective**
Opinion-Driven labelling approach	80-20	12148:1904	3029:1904
	70-30	10649:1647	4528:742
	60-40	9138:1401	6039:988
Labelling Approach From Pecar & Simko (2021).	80-20	15025:3945	3773:970
	70-30	13135:3464	5663:1451
	60-40	11248:2979	7550:1936

5.3 Training Procedure

The training model was designed to train a PyTorch model using a specified data loader, number of epochs, and learning rate. The AdamW optimizer, a variant of the Adam optimizer tailored for deep learning models and commonly used in natural language processing tasks was applied. The learning rate, which governs the step size during parameter updates, was set to 2e-5 (0.00005). Table 5.5 shows the epoch for the two approaches. Figure 5.41, Figure 5.42, Figure 5.43 shows the training loss curve in different ratio for Opinion-Driven Labelling approach. All the training loss curve show the minimum training loss at around 15 epoch. Figure 5.44, Figure 5.45, and Figure 5.46 shows the training loss curve in different ratio for Labelling Approach From Pecar & Simko (2021). In the three training loss curve show that the training loss reach minimum in 20 epochs, after 20 epochs the training loss become flat.

Table 5. 5. Number of Epochs for Each Approach

Approach	Num of Epochs	Training Loss Curve
Opinion-Driven labelling approach	15	Figure 5.41, Figure 5.42, Figure 5.43
Labelling Approach From Pecar & Simko, (2021).	20	Figure 5.44, Figure 5.45, Figure 5.46

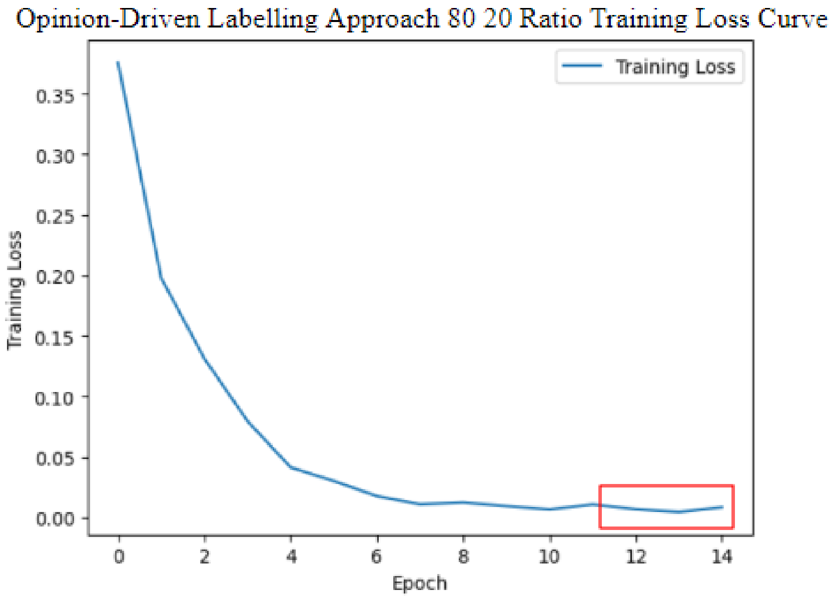


Figure 5. 41. Opinion-Driven Labelling Approach 80 20 Ratio Training Loss Curve

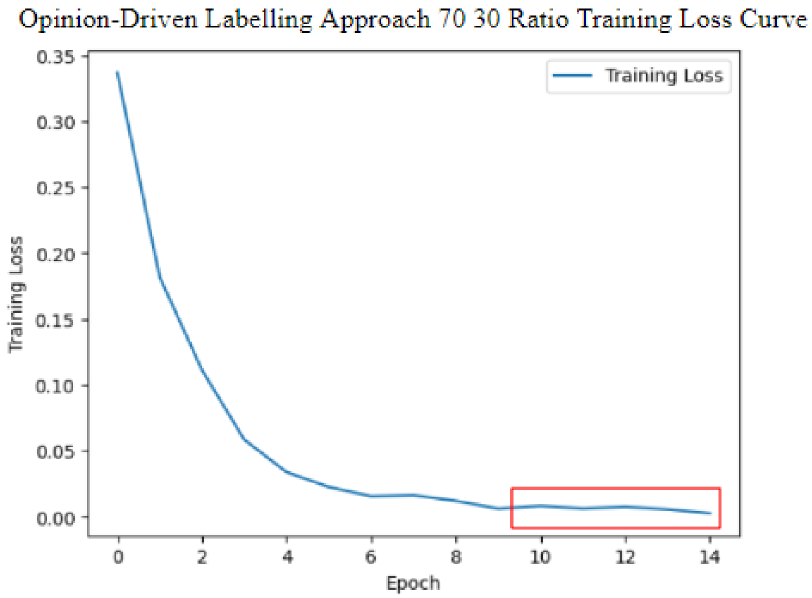


Figure 5. 42. Opinion-Driven Labelling Approach 70 30 Ratio Training Loss Curve

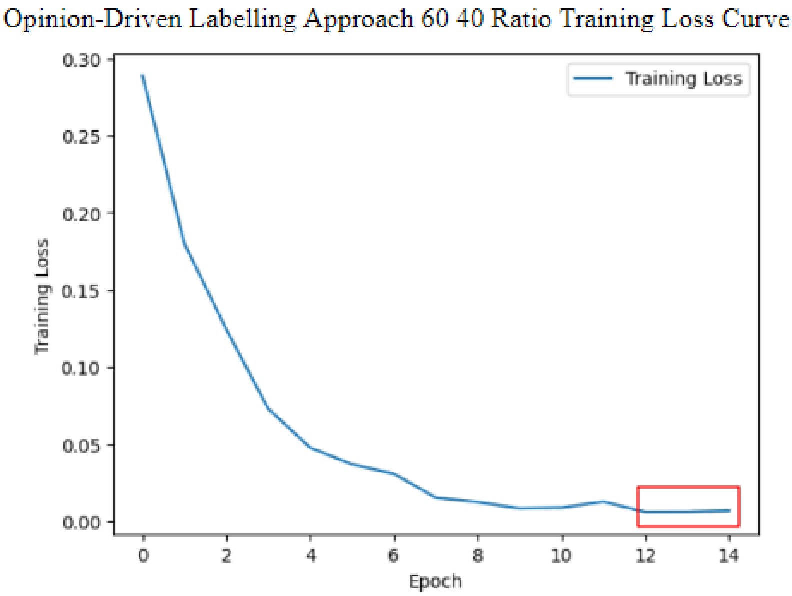


Figure 5. 43. Opinion-Driven Labelling Approach 60 40 Ratio Training Loss Curve

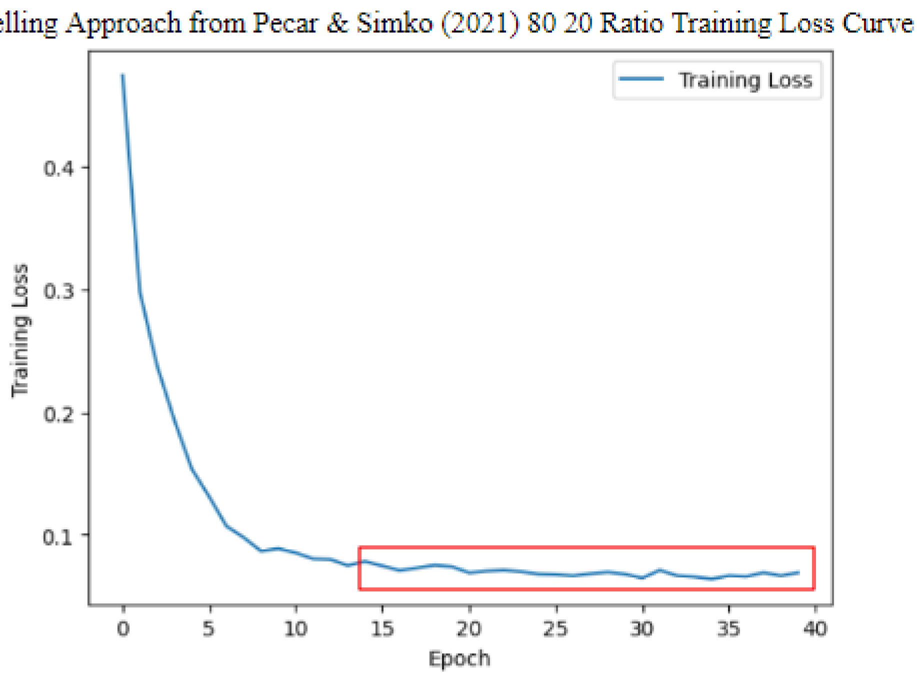


Figure 5. 44. Labelling Approach from Pecar & Simko (2021) 80 20 Ratio Training Loss Curve

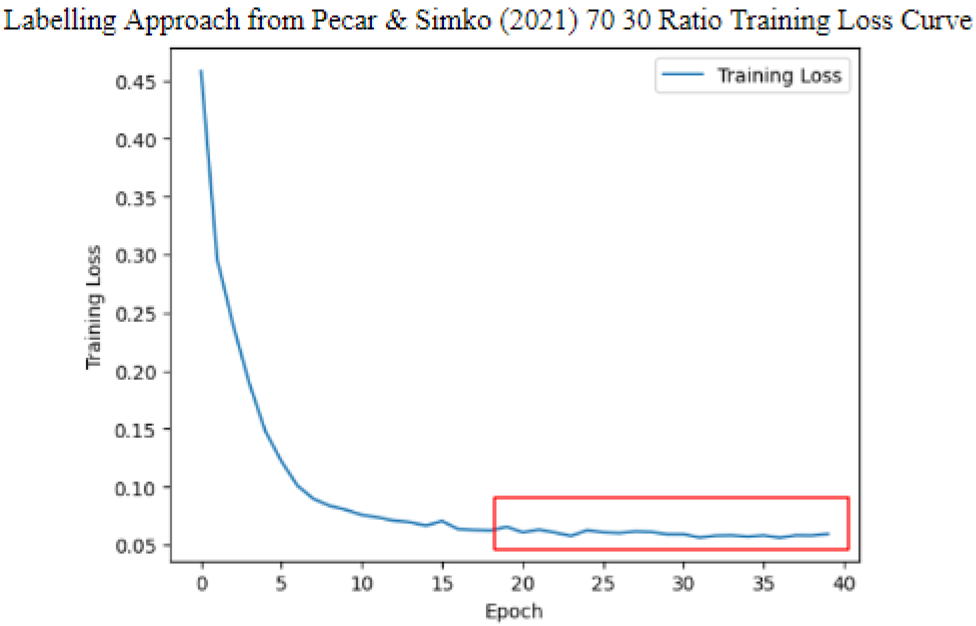


Figure 5. 45. Labelling Approach from Pecar & Simko (2021) 70 30 Ratio Training Loss Curve

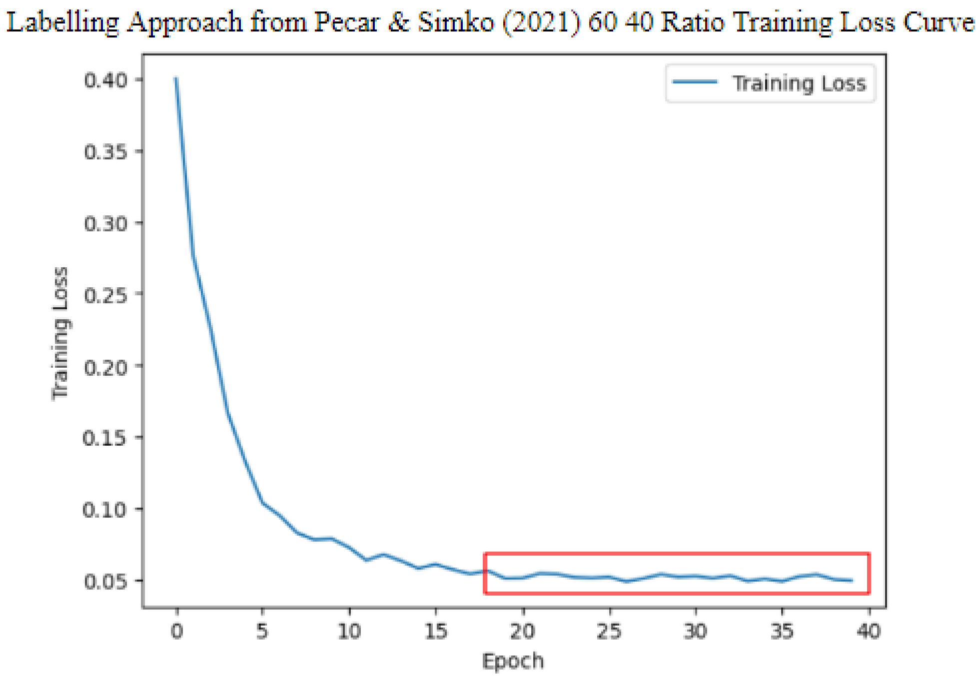


Figure 5. 46. Labelling Approach from Pecar & Simko (2021) 60 40 Ratio Training Loss Curve

5.4 Performance Evaluation

Table 5.6 presents the evaluation metrics for the MEMD dataset using the Opinion Driven Labelling Approach across different train-test splits. The 80-20 split achieves an accuracy of 0.9451, with precision, recall, and F1-score values of 0.8510, 0.7299, and 0.7858, respectively. The 70-30 split achieves slightly lower accuracy (0.9427) but higher precision (0.8931), recall (0.8624), and F1-score (0.8768). The 60-40 split has the lowest accuracy (0.9372) but shows similar precision (0.8821), recall (0.8488), and F1-score (0.8643) to the 70-30 split.

These results suggest that the Opinion Driven Labelling Approach yields consistent performance across different train-test splits. The 70-30 split, in particular, demonstrates higher precision, recall, and F1-score compared to the other splits, indicating its effectiveness in identifying relevant instances in the MEMD dataset. Overall, the results suggest that the choice of train-test split can impact the model's performance, with slight variations in accuracy, precision, recall, and F1-score observed across different splits.

Table 5. 6. MEMD Dataset using Opinion Driven Labelling Approach Evaluation Metrics.

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.9451	0.8510	0.7299	0.7858
70 - 30	0.9427	0.8931	0.8624	0.8768
60 - 40	0.9372	0.8821	0.8488	0.8643

Table 5.7 presents the evaluation metrics for the MEMD dataset using the Pecar & Simko (2021) Approach across different train-test splits. The 80-20 split achieves an accuracy of 0.8640, with precision, recall, and F1-score values of 0.7912, 0.7901, and 0.7906, respectively. The 70-30 split achieves slightly lower accuracy (0.8635) but higher precision (0.7982), recall (0.7579), and F1-score (0.7751). The 60-40 split has the lowest accuracy (0.8592) but shows similar precision (0.7830), recall (0.7852), and F1-score (0.7841) to the other splits.

These results suggest that the Pecar & Simko (2021) Approach yields consistent performance across different train-test splits. The 80-20 split demonstrates higher accuracy, recall, and F1-score compared to the other splits, indicating its effectiveness in identifying relevant instances in the MEMD dataset. Overall, the results suggest that the choice of train-test split can impact the model's performance, with slight variations in accuracy, precision, recall, and F1-score observed across different splits.

Table 5. 7. MEMD Dataset using Pecar & Simko, (2021) Labelling Approach Evaluation Metrics.

	Evaluation metrics			
Train-test split	Accuracy	Precision	Recall	F1-score
80 - 20	0.8640	0.7912	0.7901	0.7906
70 - 30	0.8635	0.7982	0.7579	0.7751
60 - 40	0.8592	0.7830	0.7852	0.7841

5.5 Discussion

In the project, the datasets were trained and tested using three different approaches: the Sentiment Labelling approach (Please refer to Section 3.6), the Opinion Driven Labelling approach (Please refer to Section 5.2), and the Pecar & Simko, (2021) Labelling approach (Please refer to Section 5.2). It was found that the Sentiment Labelling approach achieved the highest accuracy compared to the other two approaches.

The higher accuracy obtained by the Sentiment Labelling approach may be attributed to the smaller number of objective (non-subjective) class instances. Table 4.3 displays the counts of subjective and objective instances, revealing that the training set contains 18,474 subjective instances and 496 objective instances, whereas the other two approaches have more than 1,000 rows for the objective class. Consequently, the model may exhibit less bias when trained on the objective class with fewer instances. However, as the number of objective class instances increases, the model's bias towards it may escalate. While the model may predict the small class with higher accuracy, the overall accuracy may decrease. (Why Will the Accuracy of a Highly Unbalanced Dataset Reduce After Oversampling?, n.d.)

Upon comparing the results of the three approaches, it was observed that the Opinion Driven Labelling approach (Please refer to Table 5.6) exhibited overall stable results in accuracy, precision, recall, and F1-score.

6.0 Conclusion

In phase 1, this project collected datasets containing sentences, aspects, and sentiment attributes, termed aspect-based sentiment analysis datasets (ABSA). The sentiment attributes comprised three classes: positive, negative, and neutral, it uses Sentiment Labelling Approach (Refer to Section 3.6) to define subjective and objective (non-subjective) which were used for subjectivity classification. The subjectivity classification model, utilizing BERT for sequence classification, was tested on various ABSA datasets. The model performed well on datasets such as SemEval 2014 and the MEMD laptop dataset, which had balanced classes of subjectivity and objectivity. However, it exhibited poor performance metrics on the MEMD book, clothing, and hotel domains. Therefore, to address the poor performance metrics, this project combined all the MEMD domains and conducted subjectivity classification again.

In phase 2, this project also employed two other labelling approaches: the Opinion Driven Labelling approach and the Pecar & Simko (2021) Labelling Approach to label the subjectivity attributes in MEMD domains. Learning from the lessons of phase 1, the datasets were trained by combining all the MEMD domains, resulting in improved outcomes. Overall, the Sentiment approach provide the highest accuracy, the Opinion Driven Labelling approach provide a stable performance metrics.

References

Amarouche, K., Benbrahim, H., & Kassou, I. (2015). Product Opinion Mining for Competitive Intelligence. *Procedia Computer Science*, *73*, 358–365. https://doi.org/10.1016/j.procs.2015.12.004

Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 632–642. https://doi.org/10.18653/v1/D15-1075

Cai, H., Song, N., Wang, Z., Xie, Q., Zhao, Q., Li, K., Wu, S., Liu, S., Yu, J., & Xia, R. (2023). *MEMD-ABSA: A Multi-Element Multi-Domain Dataset for Aspect-Based Sentiment Analysis*.

Cho, K., Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. https://doi.org/10.3115/v1/D14-1179

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014a). Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 49–54. https://doi.org/10.3115/v1/P14-2009

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014b). Adaptive recursive neural network for target-dependent twitter sentiment classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 49–54.

Hoang, M., Bihorac, O. A., & Rouces, J. (2019). Aspect-Based Sentiment Analysis using BERT. In M. Hartmann & B. Plank (Eds.), *Proceedings of the 22nd Nordic Conference on Computational Linguistics* (pp. 187–196). Linköping University Electronic Press. https://aclanthology.org/W19-6120

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Iqbal, M., Karim, A., & Kamiran, F. (2019). Balancing Prediction Errors for Robust Sentiment Classification. *ACM Transactions on Knowledge Discovery from Data*, *13*(3), 1–21. https://doi.org/10.1145/3328795

Karimi, A., Rossi, L., & Prati, A. (2020). *Adversarial Training for Aspect-Based Sentiment Analysis with BERT*.

Keshavarz, H., & Abadeh, M. S. (2017). Accurate frequency-based lexicon generation for opinion mining. *Journal of Intelligent & Fuzzy Systems*, *33*(4), 2223–2234. https://doi.org/10.3233/JIFS-16562

Li, R., Chen, H., Feng, F., Ma, Z., Wang, X., & Hovy, E. (2021). Dual Graph Convolutional Networks for Aspect-based Sentiment Analysis. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6319–6329. https://doi.org/10.18653/v1/2021.acl-long.494

Li, X., Bing, L., Zhang, W., & Lam, W. (2019). Exploiting BERT for End-to-End Aspect-based Sentiment Analysis. *Proceedings of the 5th Workshop on Noisy User-Generated Text (W-NUT 2019)*, 34–41. https://doi.org/10.18653/v1/D19-5505

Liang, B., Yin, R., Du, J., Gui, L., He, Y., Yang, M., & Xu, R. (2023). Embedding Refinement Framework for Targeted Aspect-Based Sentiment Analysis. *IEEE Transactions on Affective Computing*, *14*(1), 279–293. https://doi.org/10.1109/TAFFC.2021.3071388

Liu, B. (2022). *Sentiment analysis and opinion mining*. Springer Nature.

Liu, H., Chatterjee, I., Zhou, M., Lu, X. S., & Abusorrah, A. (2020). Aspect-Based Sentiment Analysis: A Survey of Deep Learning Methods. *IEEE Transactions on Computational Social Systems*, *7*(6), 1358–1375. https://doi.org/10.1109/TCSS.2020.3033302

Mowlaei, M. E., Abadeh, M. S., & Keshavarz, H. (2018). A lexicon generation method for aspect-based opinion mining. *2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES)*, 107–112.

Mowlaei, M. E., Saniee Abadeh, M., & Keshavarz, H. (2020). Aspect-based sentiment analysis using adaptive aspect-based lexicons. *Expert Systems with Applications*, *148*, 113234. https://doi.org/https://doi.org/10.1016/j.eswa.2020.113234

Nguyen, T. H., & Shirai, K. (2015). PhraseRNN: Phrase Recursive Neural Network for Aspect-based Sentiment Analysis. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2509–2514. https://doi.org/10.18653/v1/D15-1298

Pasupa, K., & Seneewong Na Ayutthaya, T. (2019). Thai sentiment analysis with deep learning techniques: A comparative study based on word embedding, POS-tag, and sentic features. *Sustainable Cities and Society*, *50*, 101615. https://doi.org/10.1016/j.scs.2019.101615

Patel, A. A., & Arasanipalai, A. U. (2021). *Applied Natural Language Processing in the Enterprise*. “ O’Reilly Media, Inc.”

Pecar, S., & Simko, M. (2021). *Exploiting Subjectivity Knowledge Transfer for End-to-End Aspect-Based Sentiment Analysis* (pp. 269–280). https://doi.org/10.1007/978-3-030-83527-9_23

Phan, M. H., & Ogunbona, P. O. (2020). Modelling Context and Syntactical Features for Aspect-based Sentiment Analysis. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 3211–3220. https://doi.org/10.18653/v1/2020.acl-main.293

Plank, B., Søgaard, A., & Goldberg, Y. (2016). *Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss*.

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S. M., & Eryiğit, G. (2016). SemEval-2016 Task 5: Aspect Based Sentiment Analysis. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 19–30. https://doi.org/10.18653/v1/S16-1002

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., & Androutsopoulos, I. (2015). SemEval-2015 Task 12: Aspect Based Sentiment Analysis. *Proceedings of the 9th*

International Workshop on Semantic Evaluation (SemEval 2015), 486–495. https://doi.org/10.18653/v1/S15-2082
Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). SemEval-2014 Task 4: Aspect Based Sentiment Analysis. *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 27–35. https://doi.org/10.3115/v1/S14-2004
Rothman, D. (2021). *Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more*. Packt Publishing Ltd.
Saeidi, M., Bouchard, G., Ai, B., Liakata, M., & Riedel, S. (n.d.). *SentiHood: Targeted Aspect Based Sentiment Analysis Dataset for Urban Neighbourhoods*. http://www.geonames.org/
San Yee, W., Ng, H., Yap, T. T. V., Goh, V. T., Ng, K. H., & Cher, D. T. (2022). An evaluation study on the predictive models of breast cancer risk factor classification. *J. Logist. Inform. Serv. Sci*, *9*, 129–145.
Satapathy, R., Pardeshi, S., & Cambria, E. (2022). *Polarity and Subjectivity Detection with Multitask Learning and BERT Embedding*.
Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093
Song, X., Salcianu, A., Song, Y., Dopson, D., & Zhou, D. (2020). *Fast WordPiece Tokenization*.
Song, Y., Wang, J., Liang, Z., Liu, Z., & Jiang, T. (2020). *Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference*.
Turabieh, H. (2019). Hybrid Machine Learning Classifiers to Predict Student Performance. *2019 2nd International Conference on New Trends in Computing Sciences (ICTCS)*, 1–6. https://api.semanticscholar.org/CorpusID:208880861
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*.
Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2017). *Graph Attention Networks*.
Wang, X., Liu, Y., SUN, C., Wang, B., & Wang, X. (2015). Predicting Polarities of Tweets by Composing Word Embeddings with Long Short-Term Memory. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1343–1353. https://doi.org/10.3115/v1/P15-1130
Why will the accuracy of a highly unbalanced dataset reduce after oversampling? (n.d.). Data Science Stack Exchange. https://datascience.stackexchange.com/questions/28227/why-will-the-accuracy-of-a-highly-unbalanced-dataset-reduce-after-oversampling
Zeng, J., Ma, X., & Zhou, K. (2019). Enhancing Attention-Based LSTM With Position Context for Aspect-Level Sentiment Classification. *IEEE Access*, *7*, 20462–20471. https://doi.org/10.1109/ACCESS.2019.2893806
Zhang, B., Xu, D., Zhang, H., & Li, M. (2019). STCS Lexicon: Spectral-Clustering-Based Topic-Specific Chinese Sentiment Lexicon Construction for Social Networks. *IEEE*

Transactions on Computational Social Systems, *6*(6), 1180–1189. https://doi.org/10.1109/TCSS.2019.2941344

Zhang, W., Li, X., Deng, Y., Bing, L., & Lam, W. (2021). Towards Generative Aspect-Based Sentiment Analysis. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 504–510. https://doi.org/10.18653/v1/2021.acl-short.64

Zhang, W., Li, X., Deng, Y., Bing, L., & Lam, W. (2023). A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges. *IEEE Transactions on Knowledge and Data Engineering*, *35*(11), 11019–11038. https://doi.org/10.1109/TKDE.2022.3230975

Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., & Xu, B. (2016). Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. *ArXiv Preprint ArXiv:1611.06639*.

Zhu, L., Xu, M., Bao, Y., Xu, Y., & Kong, X. (2022). Deep learning for aspect-based sentiment analysis: a review. *PeerJ Computer Science*, *8*, e1044. https://doi.org/10.7717/peerj-cs.1044

Appendices

Appendix A: Conference Paper

Aspect-Based Subjectivity Analysis Using a BERT-based Approach

Hu Ng
Faculty of Computing and Informatics
Multimedia University
Cyberjaya, Malaysia
nghu@mmu.edu.my

Wing Kin Chong
Faculty of Computing and Informatics
Multimedia University
Cyberjaya, Malaysia
1191302246@student.mmu.edu.my

Yu Zhang Chia
Faculty of Computing and Informatics
Multimedia University
Cyberjaya, Malaysia
1201101003@mmu.edu.my

Timothy Tzen Vun Yap
School of Mathematical & Computer Sciences
Heriot-Watt University
Putrajaya, Malaysia
timothy.yap@hw.ac.uk

Vik Tor Goh
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
vtgoh@mmu.edu.my

Lai Kuan Wong
Faculty of Computing and Informatics
Multimedia University
Cyberjaya, Malaysia
lkwong@mmu.edu.my

Ian Kim Teck Tan
School of Mathematical & Computer Sciences
Heriot-Watt Univeristy
Putrajaya, Malaysia
i.tan@hw.ac.uk

Dong Theng Cher
SIRIM Berhad
Shah Alam, Malaysia
dtcher@sirim.my

Abstract— **Aspect-based subjectivity analysis stands as an important task in natural language processing, seeking to identify the subjectivity of various aspects or features within a text. A new method for aspect-based subjectivity analysis using BERT is introduced in this paper. BERT has demonstrated impressive performance across various NLP tasks, and its capabilities are utilized to accurately ascertain the subjectivity of specific aspects within a given text. The approach involves fine-tuning BERT on a sizable dataset annotated with aspect-level subjectivity labels, enabling the model to grasp the subtleties of aspect-based subjectivity analysis. Extensive experiments on benchmark datasets are conducted to showcase the effectiveness of this approach and compare it with existing methods. The results reveal that this proposed approach surpasses state-of-the-art techniques in aspect-based subjectivity analysis, underscoring the potential of leveraging BERT for such purposes.**

Keywords—Subjectivity Analysis, NLP, Aspect-based

I. Introduction

Subjectivity analysis, specifically on the aspect-level plays a significant role in natural language processing tasks by identifying the subjectivity of different aspects or features within a given text. This analysis is particularly useful for understanding the sentiment and opinions expressed in text data, which can be valuable for applications such as customer feedback analysis, product or service reviews, and market research[1].

By breaking down the subjectivity analysis into specific aspects or attributes, we can gain deeper insights into the nuances of opinions and sentiments expressed by individuals. This allows for a more granular understanding of the overall sentiment towards different aspects of a topic, which can be crucial for making informed decisions in various domains.

Previous research in aspect-based subjectivity analysis has focused on various techniques for identifying and analyzing subjectivity towards specific aspects or attributes of a given topic. Many studies have demonstrated the importance of this type of analysis in understanding sentiment and opinions in text data. Researchers have used approaches such as aspect-based sentiment analysis, opinion mining, and natural language processing to delve into the nuances of opinions and sentiments expressed by individuals [2], [3].

Nevertheless, with advancements in machine learning and natural language processing, there have been notable developments in aspect-based subjectivity analysis. Techniques such as deep learning, neural networks, and transfer learning have shown promise in addressing some of the challenges and improving the accuracy of sentiment analysis at the aspect level.

This study explores aspect-based subjectivity analysis with the use of Bidirectional Encoder Representation from Transformer (BERT). One of the key advantages of using BERT for aspect-based subjectivity analysis is its ability to capture contextual information and understand the relationships between words in a sentence or document. Additionally, subjectivity information labelling is performed on the selected dataset for conducting subjectivity analysis.

II. Literature Review

There are some studies that have highlighted the applications of aspect-based subjectivity analysis in fields such as customer feedback analysis, product or service reviews, and market research. The ability to analyze sentiments towards different aspects of a topic has proven to be valuable for making informed decisions and understanding the varying perspectives of individuals.

Challenges and considerations in aspect-based subjectivity analysis have also been extensively discussed in the literature. Researchers have identified issues related to the accuracy of aspect identification, handling of negation and intensification, and domain-specific sentiment analysis.

Additionally, the lack of labelled data for aspect-based sentiment analysis has been a recurring challenge.

A. Aspect-based Natural Language Processing

Sentiment analysis and opinion mining are important aspects of Natural Language Processing (NLP) that play a crucial role in understanding and analyzing textual data across various fields [4], [5]. These methods have diverse applications, such as market research, assessing customer feedback, monitoring social media activity, political analysis, and more. By examining the emotions expressed in the text, valuable insights into public perception, customer satisfaction, and overall sentiment on specific topics or products can be gained. For example, companies can use sentiment analysis to understand customer attitudes towards their products and make informed business decisions.

Aspect-based natural language processing (NLP) is a specialized field that focuses on analyzing text data at the aspect level [6]. It involves analyzing text data at a granular level, specifically focusing on different aspects or attributes of a given topic. Zhang et al. [7] made significant contributions to the field by conducting a survey on sentiment analysis and opinion mining. They not only identified the existence of different entities and aspects in a single-opinion mining problem but also provided a hierarchical-based explanation to illustrate this concept. In recent years, this insight has greatly influenced the development of aspect-based sentiment analysis (ABSA).

B. Subjectivity Analysis

Identifying a sentence as non-subjective (objective) may imply that it lacks any aspect or only contains aspects with neutral polarity [8]. While representing non-subjective sentences is relatively simple and should include only factual statements, the subjectivity of reviewers can be expressed in various ways. Knowing that a sentence is subjective (opinionated) may not be enough for additional analysis. Further exploration of subjectivity can provide valuable insights into the emotions, beliefs, and perspectives of individuals.

While much work has been done on ABSA, there is limited work on aspect-based subjectivity analysis, where the goal is to identify the subjectivity or objectivity of opinions expressed towards specific aspects. This aspect-level subjectivity analysis is crucial in understanding the overall sentiment and opinions expressed by individuals towards different aspects of a product, service, or topic [4].

Interest has been expressed in subjectivity analysis for aspect-based sentiment and opinion mining [1]. Subjectivity analysis is performed on product reviews extracted from YouTube comments, along with aspect extraction and sentiment analysis to understand subjective opinions toward different aspects of the products mentioned. However, the subjectivity analysis is not currently conducted on the aspect level, which leaves a gap in understanding nuanced subjectivity towards individual aspects. To address this gap, this research focuses on developing techniques for aspect-level subjectivity analysis.

C. BERT

The BERT model, introduced by [9], utilize the transformer architecture, has set a new standard in large language models. Its ability to incorporate bidirectional conditioning on both left and right context at all layers using only the encoder from the transformer has made it an influential language representation model in NLP. This framework allows for easy fine-tuning for different downstream tasks through transfer learning by simply adding an extra output layer. Additionally, BERT demonstrates strong performance across various tasks due to its comprehensive ability to learn word embedding context, making it suitable as a word embedding technique for capturing semantic meaning within sentences. Fig. 1 shows the architecture of the BERT model [10]. It consists of stacks of encoder from transformer where the BERT-base model contains 12 stacks and BERT-large model contains 24 stacks.

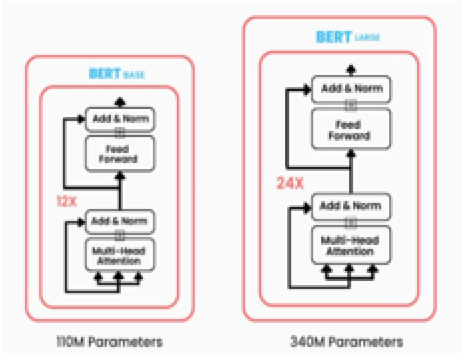


Fig. 1. Architecture for BERT model, left: BERT-base, right: BERT-large [10]

III. RESEARCH METHODOLOGY

In this study, a comprehensive approach to aspect-based subjectivity analysis using BERT will be employed. This section outlines the methodology that comprises several key steps to ensure a robust and systematic investigation.

A. Dataset

To conduct the study on aspect-based subjectivity analysis, an ABSA dataset comprising text data from diverse domains is collected. This research utilized the Multi-Element Multi-Domain (MEMD) dataset presented by [11], which is specially designed for ABSA tasks. While most existing ABSA datasets only contain part of the elements such as aspects and sentiments, the MEMD dataset includes multiple elements such as aspects, opinions, categories, and sentiments.

Furthermore, this dataset contains multiple domains and is one of the large-scale datasets for ABSA tasks. The multi-domain characteristic of the dataset allows for testing the generalizability of the aspect-based subjectivity analysis models across different domains.

MEMD contains five different domains, including books, clothing, hotel, restaurant, and laptop. All domains are labelled with the four elements: aspect, category, opinion word and sentiment, making it suitable for various ABSA tasks. Table I presents the statistics of the MEMD dataset.

TABLE I. STATISTIC OF MEMD DATASET

Domain	Books	Cloting	Hotel	Restaurant	Laptop
Sentences	1983	1538	2584	3722	2848

B. Data Labelling

A dataset with subjective and non-subjective labelling is required to conduct subjectivity analysis. Since the MEMD dataset does not contain information about the subjectivity of each aspect in each sentence, manual labelling is done to assign subjectivity labels to the aspects in the dataset. The labelling process followed a similar approach by (cite) to label the subjectivity of aspects in the MEMD dataset. The authors determine the subjectivity of an aspect in each sentence according to the opinion type. According to the paper, non-subjective labels are used for samples containing only factual information without any expressed opinion.

Following the same approach, the subjectivity labels for the aspects in the MEMD dataset are assigned based on the presence or absence of expressed opinions. Table II presents the number of subjective and non-subjective sentences of the MEMD dataset based on the aspect of each sentence after labelling.

TABLE II. NUMBER OF SENTENCES AFTER SUBJECTIVITY LABELLING ON MEMD DATASET.

Domain	Book	Clothing	Hotel	Laptop	Restaurant
Subjective	2203	1651	3439	2799	5085
Non-subjective	496	285	196	1013	399

C. Design of Experiment

This research aims to conduct subjectivity classification on an ABSA dataset. The experiment is conducted on the MEMD dataset with the annotations of subjectivity explained in the previous section. All domains are combined into one dataset instead of separated since the domain information does not affect whether a sentence is subjective. The MEMD dataset undergoes several preprocessing steps. These steps involve tokenizing which converts each sentence into word tokens and adding special tokens such as [CLS] and [SEP] tokens for BERT model usage.

In this experiment, the BERT model is utilized as the embedding model for the subjectivity classification task. Each preprocessed sentence is fed into the BERT model, which generates embeddings for each word token in the sentence. The generated embeddings are then used to classify the subjectivity of each aspect in the sentences. A linear layer is added at the end of the BERT model and acts as a classifier to return the subjectivity label of each aspect in a sentence. The overall methodology is presented in Fig. 2.

For evaluation, the accuracy and F1 measure are selected. These metrics are commonly used in subjectivity classification tasks [1] and provide a comprehensive evaluation of the model's performance.

IV. RESUTLS AND DISCUSSIONS

A. Results

The results of the aspect-based subjectivity analysis are presented in this section. Since no previous work has been done on subjectivity classification specifically for aspect-based sentiment analysis datasets, a direct comparison with other models is not possible. However, the proposed BERT-based approach achieved promising results in classifying the subjectivity of aspects in the MEMD dataset.

Table III presents the results of aspect-based subjectivity analysis and the result from [6]. The BERT-based approach is adopted in both experiments but the dataset used is different. They used SemEval 15 restaurant dataset instead of MEMD dataset. They achieved an accuracy of 87.29 and F1 score of 70.63. While the MEMD dataset achieved an accuracy of 94.05 and F1 score of 87.49 in classifying the subjectivity of aspects in the MEMD dataset. These results indicate that the BERT model is effective in accurately classifying the subjectivity of aspects in the MEMD dataset, achieving high levels of accuracy. Additionally, the confusion matrix for the MEMD dataset is shown in Fig. 3. It shows the distribution of predicted subjectivity labels for the aspect-based subjectivity classification task.

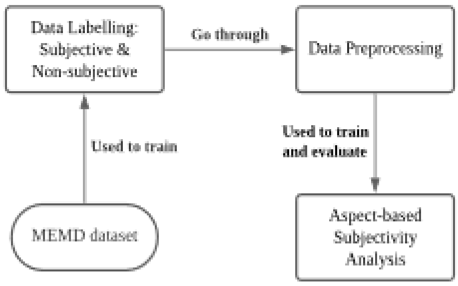


Fig. 2. Flowchart of the overall methodology.

TABLE III. RESULTS OF ASPECT-BASED SUBJECTIVITY ANALYSIS USING BERT

Dataset	Results (%)	
	Accuracy	F1
MEMD	94.06	87.49
SemEval 2015 restaurant [6]	87.29	70.63

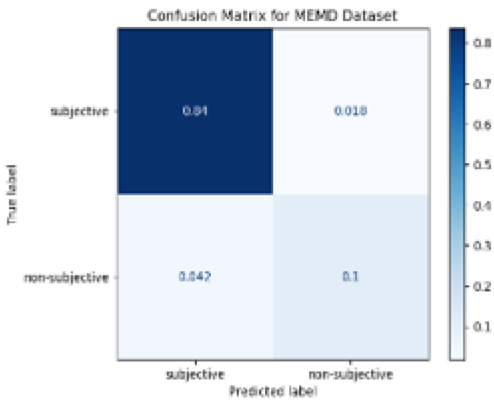


Fig. 3. Confusion matrix of subjectivity analysis on MEMD dataset.

B. Discussions

The findings of the subjectivity classification demonstrate that the models employed in this study are capable of effectively differentiating between objectivity and subjectivity according to the aspect. The results of the aspect-based subjectivity analysis using the BERT model demonstrate its effectiveness in accurately classifying the subjectivity of aspects in the MEMD dataset. The results from [6] are compared with the work in this paper. It shows that the results of this surpassed the previous work. This may be caused by more data and domains are available in the MEMD dataset than the SemEval 15 dataset.

The high accuracy and F1 score indicate the robustness of the BERT-based approach in capturing nuanced subjectivity towards individual aspects across diverse domains. This ability has the potential to assist in minimizing errors that may affect other tasks, such as ABSA, given the experiment is conducted in a pipeline method.

Furthermore, the confusion matrix suggests that the BERT model's predictions for subjectivity classification aligned well with the ground truth labels in the MEMD dataset. It also shows that there are more Type II errors (the model classifies the aspect as subjective, but it is actually objective) than Type I errors. By looking at the confusion matrix, the accuracy per non-subjective instances is worse than for subjective instances, being equal to 70% of total non-subjective sentences. These shows that the imbalance in the dataset may have effect on the dataset where there are more subjective aspects than objective ones and caused some bias in the model. However, the same problem is also encountered in the dataset used by previous work.

V. Conclusions

In conclusion, the BERT model has proved to be highly effective in classifying the subjectivity of aspects in the MEMD dataset, achieving an accuracy of 94.05% and an F1 score of 87.49% which surpass the baseline of 87.29 for accuracy and 70.63 F1 score in the work by [6]. The ability of the BERT-based approach to accurately capture nuanced subjectivity towards individual aspects across diverse domains has been demonstrated, indicating its potential to minimize errors in other tasks such as aspect-based sentiment analysis.

The findings also suggest that the BERT model's predictions for subjectivity classification align well with the ground truth labels in the MEMD dataset, as indicated by the confusion matrix. Although there are more Type II errors than Type I errors, this is potentially due to the imbalance in the dataset, with more objective aspects than subjective ones. Overall, the results of this study highlight the robustness and effectiveness of the BERT model in aspect-based subjectivity analysis, providing valuable insights for future research and applications in natural language processing and sentiment analysis.

Acknowledgment

This research was funded by the Malaysia Ministry of Higher Education (MOHE), under grant No. FRGS/1/2022/ICT06/MMU/03/3.

References

[1] M. Tetteh and M. G. Thushara, "Subjectivity Analysis for Aspect-based Sentiment and Opinion Detection of YouTube Product Reviews," 2023.

[2] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An Interactive Multi-Task Learning Network for End-to-End Aspect-Based Sentiment Analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 504–515. doi: 10.18653/v1/P19-1048.

[3] H. Yan, J. Dai, T. Ji, X. Qiu, and Z. Zhang, "A Unified Generative Framework for Aspect-based Sentiment Analysis," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 2416–2429. doi: 10.18653/v1/2021.acl-long.188.

[4] P. Sampathirao Suneetha and S. V. Row, "Aspect-Based Sentiment Analysis: A Comprehensive Survey Of Techniques And Applications," *Journal of Data Acquisition and Processing*, vol. 38, no. 3, p. 177, 2023.

[5] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, and L. Si, "Knowing What, How and Why: A Near Complete Solution for Aspect-Based Sentiment Analysis," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8600–8607, Apr. 2020, doi: 10.1609/aaai.v34i05.6383.

[6] S. Pecar and M. Simko, "Exploiting Subjectivity Knowledge Transfer for End-to-End Aspect-Based Sentiment Analysis," 2021, pp. 269–280. doi: 10.1007/978-3-030-83527-9_23.

[7] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, "A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges," *IEEE Trans Knowl Data Eng*, pp. 1–20, 2022, doi: 10.1109/TKDE.2022.3230975.

[8] I. Chaturvedi, E. Cambria, R. E. Welsch, and F. Herrera, "Distinguishing between facts and opinions for sentiment analysis: Survey and challenges," *Information Fusion*, vol. 44, pp. 65–77, Nov. 2018, doi: 10.1016/j.inffus.2017.12.006.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." 2019.

[10] A. Kumar, "BERT vs GPT Models: Differences, Examples." Accessed: Apr. 10, 2024. [Online]. Available: https://vitalflux.com/bert-vs-gpt-differences-real-life-examples/#google_vignette

[11] H. Cai *et al.*, "MEMD-ABSA: A Multi-Element Multi-Domain Dataset for Aspect-Based Sentiment Analysis," Jun. 2023.

[12] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," Aug. 2019.

[13] Z. Zhao, Z. Zhang, and F. Hopfgartner, "Utilizing subjectivity level to mitigate identity term bias in toxic comments classification," *Online Soc Netw Media*, vol. 29, p. 100205, May 2022, doi: 10.1016/j.osnem.2022.100205.

[14] T. Schopf, E. Gerber, M. Ostendorff, and F. Matthes, "AspectCSE: Sentence Embeddings for Aspect-

based Semantic Textual Similarity using Contrastive Learning and Structured Knowledge," Jul. 2023.
[15] S. U. S. Chebolu, F. Dernoncourt, N. Lipka, and T. Solorio, "Survey of Aspect-based Sentiment Analysis Datasets," Apr. 2022.
[16] H. Yang and K. Li, "PyABSA: A Modularized Framework for Reproducible Aspect-based Sentiment Analysis." 2023.
[17] M. E. Mowlaei, M. Saniee Abadeh, and H. Keshavarz, "Aspect-based sentiment analysis using adaptive aspect-based lexicons," *Expert Syst Appl*, vol. 148, p. 113234, 2020, doi: https://doi.org/10.1016/j.eswa.2020.113234.
[18] Y. Song, J. Wang, Z. Liang, Z. Liu, and T. Jiang, "Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference," Feb. 2020.
[19] S. Fan, J. Yao, Y. Sun, and Y. Zhan, "A Summary of Aspect-based Sentiment Analysis," *J Phys Conf Ser*, vol. 1624, no. 2, p. 22051, Oct. 2020, doi: 10.1088/1742-6596/1624/2/022051.
[20] R. Chiha and M. Ben Ayed, "Supervised Machine Learning Approach for Subjectivity/Objectivity Classification of Social Data," 2020, pp. 193–205. doi: 10.1007/978-3-030-44322-1_15.

Appendix B: Turnitin Similarity Index Page

Turnitin *Originality Report*

- Processed on: 19-Jun-2024 22:07 +08
- ID: 2405323761
- Word Count: 18122
- Submitted: 1

chiafyp2 *By Ng HU*

Similarity Index
17%
Similarity by Source

Internet Sources:
12%
Publications:
14%
Student Papers:
N/A

include quoted | include bibliography | excluding matches < 5 words | mode: qui

1% match (Internet from 26-Jan-2024)

https://www.mdpi.com/2073-431X/13/1/25

1% match (Internet from 07-Nov-2023)

https://ips.mmu.edu.my/wp-content/uploads/2023/10/IPS_THESIS-TEMPLAT

<1% match (Internet from 06-Mar-2024)

https://www.mdpi.com/2227-7390/12/3/456

<1% match (Internet from 23-Apr-2024)

https://WWW.MDPI.COM/1660-4601/17/11/4147

Appendix C: FYP2 Meeting Logs

Meeting Log 1:

MULTIMEDIA UNIVERSITY | FACULTY OF COMPUTING & INFORMATICS

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 10/4/2024	**Meeting No.:** 1
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE
[Please write the details of the work done, after the last meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Using the approach found in the paper named Exploiting Subjectivity Knowledge Transfer for End-to-End Aspect-Based Sentiment Analysis to manually label the datasets.
2. Open a new chapter in the report to record the work done in datasets.

2. WORK TO BE DONE
[Please write the details of the work to be done, before the next meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Use the new datasets to run the subjectivity classification model.
2. Save the training loss curve and evaluation metrics like accuracy, precision, recall and F1-score.

3. PROBLEMS ENCOUNTERED AND SOLUTIONS
[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]

4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU
....
Supervisor's Signature

CHIAYUZHANG
....
Student's Signature

....
Co-Supervisor's Signature
(if applicable)

....
Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.

2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.

3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.

4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Meeting Log 2:

FACULTY OF COMPUTING & INFORMATICS

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 24/4/2024	**Meeting No.:** 2
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE
[Please write the details of the work done, after the last meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Split the new datasets into 80/20,70/30, and 60/40 ratio and run the subjectivity classification model.
2. Save the training loss curve and evaluation metrics like accuracy, precision, recall and F1-score.

2. WORK TO BE DONE
[Please write the details of the work to be done, before the next meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Complete the part in the report that records the result of new datasets.

3. PROBLEMS ENCOUNTERED AND SOLUTIONS
[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]

1. Do some research on what make the F1-score is lower than accuracy.

4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU

....

Supervisor's Signature

CHIAYUZHANG

....

Student's Signature

....

Co-Supervisor's Signature
(if applicable)

....

Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.

2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.

3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.

4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Meeting Log 3:

FACULTY OF COMPUTING & INFORMATICS

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 8/5/2024	**Meeting No.:** 3
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE
[Please write the details of the work done, after the last meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Split the new datasets into 80/20,70/30, and 60/40 ratio and run the subjectivity classification model.
2. Save the training loss curve and evaluation metrics like accuracy, precision, recall and F1-score.
3. Complete the chapter 5

2. WORK TO BE DONE
[Please write the details of the work to be done, before the next meeting]

Tasks: Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion

(Please strike out the tasks, which are not applicable)

Details (in point form):

1. Add more sentences in result and analysis part.

3. PROBLEMS ENCOUNTERED AND SOLUTIONS
[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]

4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU
………………………………
Supervisor's Signature

CHIAYUZHANG
………………………………
Student's Signature

………………………………
Co-Supervisor's Signature
(if applicable)

………………………………
Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.
2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.
3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.
4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Meeting Log 4:

FACULTY OF COMPUTING & INFORMATICS

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 22/5/2024	**Meeting No.:** 4
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE *[Please write the details of the work done, after the last meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):** 1. Split the new datasets into 80/20,70/30, and 60/40 ratio and run the subjectivity classification model. 2. Save the training loss curve and evaluation metrics like accuracy, precision, recall and F1-score. 3. Complete chapter 5.
2. WORK TO BE DONE *[Please write the details of the work to be done, before the next meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):** 1. Rewrite the introduction part.
3. PROBLEMS ENCOUNTERED AND SOLUTIONS *[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]*
4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU

....

Supervisor's Signature

CHIAYUZHANG

....

Student's Signature

....

Co-Supervisor's Signature
(if applicable)

....

Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.
2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.
3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.
4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Meeting Log 5:

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 5/6/2024	**Meeting No.:** 5
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE *[Please write the details of the work done, after the last meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):** 1. Complete chapter 5.
2. WORK TO BE DONE *[Please write the details of the work to be done, before the next meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):** 1. Do Poster. 2. Do Video Presentation.
3. PROBLEMS ENCOUNTERED AND SOLUTIONS *[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]*
4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU
....
Supervisor's Signature

CHIAYUZHANG
....
Student's Signature

....
Co-Supervisor's Signature
(if applicable)

....
Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.
2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.
3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.
4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Meeting Log 6:

TPT3101 Project (FYP2) Meeting Log
Trimester: March 2024 (Trimester ID:2410)

Meeting Date: 19/6/2024	**Meeting No.:** 6
Meeting Mode: Online **(In-person / Online)**	
Project ID: 2545	**Project Type:** **Research-based / Application-based**
Project Title : Aspect Based Sentiment Analysis Using Bert and Deep Learning	
Student ID : 1201101003	**Student Name:** ChiaYuZhang
Student Programme and Specialisation: Bachelor of Computer Science Major in Data Science	
Supervisor Name: Ng Hu	**Co-Supervisor Name:** **(if applicable)**
Collaborating Company: **(if applicable)**	**Company Supervisor Name:** **(if applicable)**

1. WORK DONE *[Please write the details of the work done, after the last meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):** 1. Complete chapter 5. 2. Poster 3. Video Presentation
2. WORK TO BE DONE *[Please write the details of the work to be done, before the next meeting]* **Tasks:** Implementation / Testing (Application-based projects) or Evaluation of Findings and Research Contribution (Research-based projects) / Commercialisation Proposal (Application-based projects) or Research Paper (Research-based Projects) / Draft Final Report Completion ***(Please strike out the tasks, which are not applicable)*** **Details (in point form):**
3. PROBLEMS ENCOUNTERED AND SOLUTIONS *[Please write the details of the problems encountered, after the last meeting and provide the solutions / plan for the solutions]*
4. COMMENTS (Supervisor / Co-Supervisor / Company Supervisor)

NGHU
....
Supervisor's Signature

CHIAYUZHANG
....
Student's Signature

....
Co-Supervisor's Signature
(if applicable)

....
Company Supervisor's Signature
(if applicable)

IMPORTANT NOTES TO STUDENTS:

1. Items 1 – 3 are to be completed by the students prior to the meeting. Item 4 is to be completed by the supervisor / co-supervisor / company supervisor.

2. Student must upload the soft copies of the meeting logs and attach them along with final (FYP2) report.
   Minimum requirement is SIX Meeting Logs (Period: Week 4 to Week 14). Students can have fortnightly meetings with the supervisor.

3. Log sheets provide the basis for evaluating the General Effort (Project Management, Attitude, and Technical Competency) of the student, by the supervisor and for checking the attendance requirement of the student, by the FYP Committee.

   This also provides the student with feedback from the supervisor / co-supervisor / company supervisor on the tasks done and provides the plan for the upcoming tasks. This can provide the motivation for the student to give consistent and efficient effort throughout the period of FYP.

4. Student who fails to meet the minimum requirement (six nos.) of log sheets will not be allowed to submit FYP report.

Appendix D: Source Code

GitHub link: https://github.com/zychia2020/Aspect-based-sentiment-analysis-using-BERT-and-deep-learning.git