ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 14 (2024) No. 3, pp. 400-417
DOI:10.33168/JSMS.2024.0323

Objectivity-Subjectivity Detection to Boost Sentiment Analysis Accuracy: An Approach using BERT

Eashvaren Chandar [1], Wing Kin Chong[1], Timothy Tzen Vun Yap [2], Hu Ng[1+], Vik Tor Goh[3], Lai Kuan Wong[1], Ian Kim Teck Tan[2], Dong Theng Cher[4]

[1] Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia
[2] School of Mathematical & Computer Sciences, Heriot-Watt University, Putrajaya, Malaysia
[3] Faculty of Engineering, Multimedia University, Cyberjaya, Malaysia
[4] SIRIM Berhad, Shah Alam, Malaysia

Abstract. This study analyzed the impact of preceding objectivity-subjectivity classification on sentiment analysis using BERT embeddings. The English Wikipedia and Shopee Review datasets were utilized to train classification and sentiment analysis models. Multiple machine learning models are implemented, including LinearSVC, Random Forest, AdaBoost, XGBoost, LightGBM, LSTM, and a basic dense layer. A BERT model is used for word embedding's purpose. The research consists of three models: Model 1, Model 2, and Model 3. Model 1 focuses on distinguishing between objective and subjective statements, achieving an accuracy of 99%. Model 2 performs sentiment analysis on the Shopee Review dataset, but its accuracy is observed to be less than 50%. Both models are evaluated using both raw and preprocessed data, with raw data demonstrating better accuracy. In Model 3, the best-performing models from Model 1 and Model 2 are combined in a sequential process. Results showed incorporating objectivity-subjectivity classification improved sentiment analysis accuracy by 7% for the Shopee Review dataset. The findings demonstrate the value of text classification for enhancing performance of sentiment analysis using contextual embeddings.

Keywords: Objectivity and subjectivity classification, Sentiment analysis, BERT, NLP

1. Introduction

The Internet has revolutionized the way people live their lives in the modern world. It has become an indispensable tool, providing easy access to a vast array of resources and information at our fingertips. While the Internet serves various purposes, its fundamental essence lies in the sharing of knowledge. This knowledge sharing can take many forms, including the exchange of opinions. With the advancements in machine learning and deep learning, sophisticated techniques can be applied to analyze such data, leading to solutions for real-world problems like sentiment analysis (Nhan Cach Dang, 2020). While previous studies have explored sentiment analysis with BERT, few works have specifically examined the role of preceding text classification on model performance.

In the realm of sentiment analysis, the classification of objectivity and subjectivity plays a crucial role as it enables the model to understand the context and viewpoint of the text under examination. Sentiment analysis involves determining whether a text reflects a positive, negative, or neutral sentiment. However, the sentiment expressed in language can often be influenced by the author's perspective or bias. For instance, if a news article discusses a natural disaster, it is likely to be objective, presenting factual information. On the other hand, if someone tweets about the same natural disaster, it is likely to be subjective and may convey positive sentiment if they are praising the rescue crew or negative sentiment if they are upset about losing their home. Subjectivity refers to the presence of bias or personal opinion in the text, while objectivity denotes the absence of such elements. To accurately classify the sentiment of a text, the model must be able to distinguish between objective and subjective content. By considering the context and perspective of the text and classifying it as objective or subjective, the model can achieve more precise sentiment analysis.

There is assumption that sentiment classification is more sensitive to subjectivity, as such the accuracy may benefit from preceding objectivity-subjectivity classification. However, this has not been developed and investigated in recent studies. The goal of this project is to develop a model that incorporates Bidirectional Encoder Representations from Transformers (BERT) to accurately classify text as objective or subjective and perform sentiment analysis on the subjective text. The main objectives of the project include:

- Acquiring and preparing datasets that contain objective and subjective statements for training and evaluation purposes.
- Implementing a text classification model using BERT to classify text as objective or subjective.
- Developing a sentiment analysis model using BERT to analyze the sentiment of subjective text.
- Comparing the performance of the sentiment analysis model when coupled with the objective and subjective text classification model versus when it is used alone.

By achieving these objectives, this project aims to enhance the accuracy and effectiveness of sentiment analysis by incorporating objective-subjective classification using BERT.

2. Literature Review

The field of Natural Language Processing (NLP) holds great potential for computers to achieve a level of understanding of text similar to that of humans. NLP encompasses a range of techniques and approaches aimed at enabling machines to process, interpret, and generate human language. It has proven to be beneficial for various tasks, including speech recognition, sentiment analysis, machine translation, question answering, and more. However, teaching computers to comprehend human language is a challenging task that involves overcoming several complexities.

Human language is a complex system with intricate rules and nuances. Words can have different meanings depending on the context and sentence structure. The use of grammar, syntax, and semantics adds further complexity. Additionally, human language often incorporates metaphors, idioms, sarcasm, and cultural references that require a deep understanding of context and cultural knowledge to interpret

correctly. Furthermore, language is dynamic, with new words, phrases, and meanings emerging over time.

Considering these challenges, researchers have been actively exploring ways to improve computers' understanding of human language. They have developed advanced algorithms, models, and techniques based on machine learning, deep learning, and artificial intelligence. These approaches involve training models on vast amounts of text data to learn patterns, relationships, and contextual information.

Since computers can only understand numbers, an algorithm is necessary to convert words into numerical values. NLP experts employ word embeddings to transform words into their numerical equivalents within a sentence. Once converted, these learned representations can be readily processed by NLP algorithms to analyze textual data. Words are represented as real-valued numerical vectors through word embeddings. Each word in a sequence or phrase is tokenized and converted into a linear space. Word embeddings help determine the semantic meaning of each word within the text. They provide numerical representations for words that share similar meanings.

2.1. Traditional Word Embeddings Technique

Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical technique used to evaluate the importance of words in a text. It combines two measures: Term Frequency (TF) and Inverse Document Frequency (IDF). TF measures the frequency of a word in a document, while IDF measures how uncommon the words are across the entire document collection. By multiplying TF and IDF, a final TF-IDF score is obtained, reflecting the significance of a word in a specific document within a corpus. TF-IDF is commonly applied in various NLP tasks, including information retrieval and keyword extraction. It allows for the identification of important keywords based on their frequency in a document and rarity across the entire corpus. However, a limitation of TF-IDF is that it does not effectively capture the semantic meaning of words within a sentence. In a specific research example by Qaiser and Ali (2018) TF-IDF is utilized to assess the relevance of keywords to documents in a corpus. The research focuses on sorting frequently occurring keywords based on website domains, such as '.biz', '.com', and '.edu'. This demonstrates the practical application of TF-IDF in tasks like keyword extraction. The study also highlights the limitation of TF-IDF in distinguishing words with slight changes in tense.

Word2Vec (Mikolov, 2013) is another popular word embedding technique used by researchers. It assigns similar vector representations to words in a sentence based on cosine similarity. There are two variants of Word2Vec: Continuous Bag of Words (CBOW) and skip-gram. CBOW predicts the target word by considering multiple words as input, aiming to capture the context without considering word position. In contrast, skip-gram takes a single word as input and predicts the nearby context words, excluding the input word itself. The skip-gram variant is named as such because it skips predicting the context word for the input word. Researchers such as Mohamed (2021); Kolesnikova, Gelbukh, Pinto, Singh, and Perez (2020); Sharma, Chaurasia, & Srivastava (2020) have utilized Word2Vec in their work, showcasing its effectiveness in various NLP tasks. Word2Vec's ability to generate vector representations of words based on context has proven useful in capturing semantic relationships between words and improving the performance of NLP algorithms.

Global Vectors for Word Representation (GloVe) (Pennington, 2014) is an extension of the Word2Vec technique that considers the contextual information of an entire corpus rather than neighboring words. It creates a large matrix to record word co-occurrence by utilizing two modeling approaches: Local Context and Global Matrix Factorization Window. The Local Context approach effectively utilizes statistical information but lacks the ability to identify patterns beyond word similarity. In contrast, the Global Matrix Factorization Window approach can recognize complex patterns like word analogies but does not effectively utilize statistical information due to training being based on local context windows rather than global co-occurrence counts. GloVe has been extensively researched and applied in various scenarios such as predicting fake news, forecasting disasters, book classification, and more, as evidenced by studies conducted by Kulkarni, Monika, Shruthi, Bharadwaj, and Uday (2022), Ranade, Telge, & Mate (2022), Devi and Sivakumar (2021), Biradar, Raagini, Varier,

and Sudhir (2019). However, GloVe has certain limitations, GloVe does not consider word order or position in a sentence, potentially overlooking important syntactic or structural information Furthermore, Out-of-vocabulary (OOV) words pose a challenge as GloVe may not generate meaningful embeddings for words not present in the training data.

2.2. Word Embeddings Using Transfer Learning

Transfer learning is a method that allows a pre-trained model to be adapted for a specific task, including developing task-specific representations for word embeddings in NLP applications. By fine-tuning a pre-trained model on a smaller dataset while retaining the pre-trained weights, the model can leverage previously learned representations while learning new ones specific to the task. Transfer learning is particularly useful when there is limited labeled data available, as it enables the model to learn from a larger dataset and generalize better. Studies conducted by Alyafeai, AlShaibani, and Ahmad (2020); Park, Hauschild, and Heider (2021) demonstrate the performance benefits of transfer learning in scenarios with limited data. In the domain of word embeddings, popular pre-trained models like ELMO, GPT-3, and BERT can be utilized for transfer learning.

Embeddings from Language Models (ELMO) (Peters, et al., 2018) is a pre-trained deep learning model developed by the Allen Institute for Artificial Intelligence (AI2). It utilizes a bi-directional LSTM architecture, considering both the forward and backward context of words. Unlike uni-directional word embeddings like word2vec or GloVe, ELMO provides contextualized word representations tailored to specific NLP tasks. It can be customized for various NLP jobs and is trained on a large corpus of text data. Numerous studies have demonstrated ELMO's effectiveness in solving NLP problems. For example, research by Madichetty and Sridevi (2020) showcases its superiority over traditional classifiers like SVC in classifying crisis-related data on Twitter. Aligning with Madichetty and Sridevi findings, Ulčar and Robnik-Šikonja's (2022) research demonstrates ELMO's applicability in cross-lingual tasks with limited resources. Other studies, such as those conducted by Maslennikova (2019); Liu, Wen, Gao, Zheng, and Zheng (2020); Gupta and Patel (2020), highlight the efficiency of ELMO as an embedding technique when combined with other models to address NLP challenges effectively.

Generative Pre-trained Transformer 3 (GPT-3) (Brown, et al., 2020), developed by OpenAI, is a state-of-the-art language-generating model. It employs a transformer architecture, introduced by Google researchers (Vaswani et al., 2017), using self-attention mechanisms to effectively handle input sequences of varying lengths and parallelize computations. Trained on a vast corpus of text data, GPT-3 can generate human-like language. The model's capabilities have been demonstrated through the creation of ChatGPT, a chatbot that utilizes the GPT-3 model. Research conducted by Bajaj, Goel, Gupta, and Batra, (2022) showcases GPT-3's effectiveness in extracting software requirements' use cases in multiple languages. Additionally, India and Gupta (2022) highlighted the model's potential in detecting hate speech, emphasizing the role of large language models in addressing such issues. Saravanan and Sudha's (2022) research explores GPT-3's application in content generation and transformation systems, while studies by Dehouche (2021); Zong and Krishnamachari (2022) aim to identify the limitations and challenges of the model. These studies, along with others, contribute to the growing understanding and exploration of GPT-3's capabilities.

Google developed a pre-trained transformer-based neural network model for NLP called BERT (Devlin, Chang, Lee, & Toutanova, 2018). Similar to GPT-3, BERT has a transformer architecture with the addition of bidirectional learning akin to ELMO. Thus, BERT can make use of both the advantage of contextual learning from bidirectional learning and parallel computing using transformer architecture. The contextual nature of BERT's embeddings makes them distinctive in that they vary depending on the context in which a word appears. The word "*bank*" for instance, will be embedded i.e., represented in vectors, differently in the sentences "*I'm going to the bank*" and "*I'm going to the river bank*". BERT is trained on a large corpus of text data using techniques like Masked Language Modelling (MLM) and next-sentence prediction. It generates fixed embeddings for each token in an input sentence, preserving

contextual information. These embeddings are learned during pre-training and not updated during fine-tuning for specific tasks. Semantically related words are mapped to nearby positions in the embedding space. BERT has been successfully applied to various NLP problems. Research done by (Li et al., 2019) incorporates BERT for text-to-dynamic character-level embedding, followed by BiLSTM and CNN to extract local features. In a basic warranty data study done by Xu, Zhang, and Hong (2022), the researcher utilizes BERT for classification and severity modeling, resulting in improved accuracy for warranty claims. BERT also demonstrates effectiveness in languages other than English, as shown in studies involving Turkish and other languages (Özçift, Akarsu, Yumuk, & Söylemez, 2021); (Wadud, Mridha, Shin, Nur, & Saha, 2022). These applications highlight BERT's versatility and its ability to enhance performance across different NLP tasks.

2.3. Objectivity and Subjectivity Text Classification

Text can be categorized as objective or subjective, with objective text conveying facts and information, while subjective text expresses opinions and viewpoints. Objectivity is associated with neutral language, facts, and evidence, commonly found in news reporting and scientific writing. Subjectivity involves personal pronouns, emotions, and value judgments, often seen in opinion pieces and creative writing. Text classification models are trained to recognize and classify content based on these characteristics.

Supervised machine learning approaches, such as using classifiers like Naive Bayes, SVM, Random Forest, and Multi-Layer Perceptron, have been employed for subjectivity classification in social media data, with Random Forest performing the best with 0.676 accuracy in a study by Chiha and Ben Ayed (2020). Subjectivity text characteristics have also been utilized as sub-techniques to address larger NLP problems in studies by Zhao, Zhang, and Hopfgartner (2022); Yu, Ying, Feng, and Min (2022). Additionally, subjectivity classification has been applied in video summarization, as demonstrated by Moraes, Marcacini, and Goularte (2022). These research efforts highlight the various applications of subjectivity classification in different contexts.

2.4. Sentiment Analysis

Sentiment analysis, also known as opinion mining, is a technique that uses computational linguistics and NLP to extract subjective information from source materials. It involves identifying the sentiment expressed in the text, such as positive, negative, or neutral. Sentiment analysis finds applications in various industries, including marketing, customer service, and public relations, to understand client opinions and feedback. Training the sentiment analysis model requires high-quality and diverse labelled training data. The model's performance improves with larger and more representative datasets. Scraping data from company websites is a common approach to obtain suitable training data.

Sentiment analysis is often integrated with subjectivity classification, as demonstrated in research by Satapathy, Pardeshi, and Cambria (2022), which shows the interrelatedness of subtasks in sentiment analysis, such as subjectivity classification. They achieve an accuracy of 95.1 and 94.6 on subjectivity classification and sentiment classification respectively. CHIHAB et al. (2022) propose a sentiment analysis model combined with subjectivity classification for the Twitter Airlines Sentiment dataset. Furthermore, (Cobos, Jurado, & Blázquez-Herranz, 2019) develops NLP tools that support sentiment analysis and subjectivity classification to enhance online course teaching materials. These studies highlight the importance of subjectivity classification in sentiment analysis tasks.

3. Research Methodology

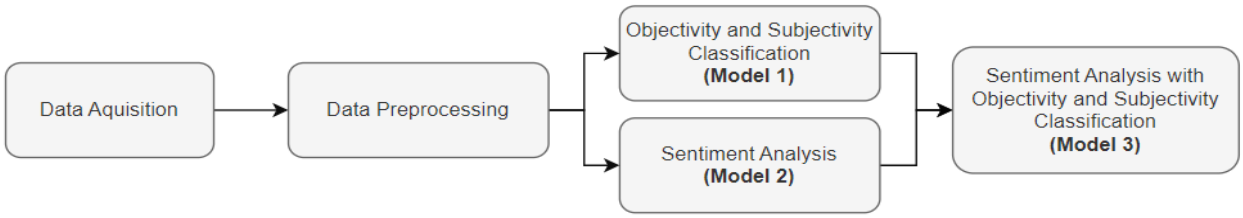

Fig.1: Methodology overview

Figure 1 provides an overview of the methodology process used in the study. The process begins with data acquisition, where the necessary data is collected from Wikipedia and Shopee**.** Once the data is acquired, it goes through a preprocessing stage. Data preprocessing involves cleaning the text, removing unnecessary characters or symbols, converting the text to lowercase, and handling other tasks to standardize the data. This step ensures that the text is in a consistent format and ready for further analysis. After preprocessing, the data is utilized to train the objectivity and subjectivity classification models (Model 1). These models are designed to distinguish between Wikipedia and Shopee text. The data is also used to train the sentiment analysis model (Model 2). Sentiment analysis aims to determine the sentiment of a text in between 1 star rating to 5 star rating. Finally, the outputs of the objectivity and subjectivity classification models, along with the sentiment analysis model, are combined to implement sentiment analysis with subjectivity and objectivity classification (Model 3). This integration allows for a more comprehensive analysis of the text, considering both the sentiment expressed and the objective or subjective nature of the content.

3.1. Data Acquisition

To train the objectivity and subjectivity text classification models, two distinct datasets are required: objective text and subjective text with polarity labels for sentiment analysis. The English Wikipedia dataset (Foundation, n.d.) is utilized for obtaining objective text data. Wikipedia follows a neutral point of view (NPOV) policy, ensuring that its content is presented in a fair and unbiased manner.

On the other hand, subjective text with polarity labels is essential for training the sentiment analysis model. For this purpose, the Shopee Review dataset (Shopee, 2022) is employed. The Shopee Review dataset consists of customer opinions and ratings on various products. This dataset serves a dual purpose as it provides subjective text for subjectivity classification and also includes polarity labels necessary for sentiment analysis. Notably, the dataset was made available during the *'Shopee Code League 2022 Data Science*' competition held in 2022, organized by Shopee.

3.2. Data Preprocessing

Both the Wikipedia and Shopee Review datasets have undergone standard preprocessing techniques to ensure the data is in a suitable format for analysis. These preprocessing steps involve applying commonly used methods to enhance the quality and consistency of the text. Firstly, the texts in both datasets are expanded to resolve contradictions or ambiguities that may arise due to abbreviations or acronyms. This expansion process helps to provide a more comprehensive understanding of the content. Next, the texts are transformed to lowercase letters to achieve case normalization. This step is crucial for treating words with different capitalizations as the same, avoiding duplication and inconsistencies in the data.

Repetitive words, such as consecutive occurrences of the same word, are removed to eliminate redundancy and optimize the data for analysis. This step helps to improve the efficiency and accuracy of subsequent processing steps. Stopwords, which are commonly used words with little semantic value (e.g., "the," "is," "and"), are removed from the texts. This helps to reduce noise in the data and focus on

the more meaningful content. Lemmatization is performed to convert words into their base or dictionary form. By reducing words to their base form, variations of the same word (e.g., "run," "running," "ran") are unified, aiding in the identification of common patterns and improving analysis results.

In the case of the Shopee Review dataset, additional steps is undertaken, which involves filtering the text to include only English language content. This ensures that the analysis focuses on texts in the desired language and enhances the accuracy of subsequent models. Moreover, emoji and emoticon conversion are performed in the Shopee Review dataset. Emojis and emoticons are converted to their textual representations to standardize their representation and avoid potential discrepancies in sentiment analysis due to these graphical elements. By following these standard preprocessing techniques, both the Wikipedia and Shopee Review datasets are prepared for subsequent steps.

3.3. Design of Experiments

In this section, we will provide a detailed explanation of the steps involved in the experiment. This includes the training, testing, and validation of the dataset, the machine learning algorithms used, the BERT model employed for generating embeddings, and the deep learning models employed for performance comparison for each Model 1, Model 2, and Model 3.

3.3.1. Dataset

Model 1 utilizes a combination of the English Wikipedia dataset, which contains objective statements, and the Shopee Review dataset, which consists of subjective statements. The dataset comprises a total of 300,000 entries, with an equal split of 150,000 for objective and 150,000 for subjective statements. Both datasets are carefully balanced to ensure an equal representation of different classes. In Model 2, the same Shopee Review dataset is employed, but this time it focuses on classifying reviews based on a 1 to 5 star rating system, ensuring class balance among the different star ratings. The dataset is then split into training and testing data, following the common ratio of 70:30. During the training process, the training data is further divided into train and validation data, with varying percentages (50%, 60%, 70%, 80%, and 90%) to assess the model's performance at different training sizes. This process is repeated twice, using the raw dataset and the preprocess dataset.

3.3.2. Machine Learning Algorithms

Three distinct categories of machine learning algorithms are employed in the experiment: Support Vector Machine (SVM), ensemble methods, and boosting algorithms. Each category encompasses specific algorithms that are utilized for performance comparison in both Model 1 and Model 2. For SVM, the Linear Support Vector Classifier (linearSVC) is selected as the algorithm of choice. SVM aims to find an optimal hyperplane that separates different classes in the feature space. The linearSVC variant is utilized, which employs a linear kernel for classification since it tends to be faster to converge the larger the number of samples.

The ensemble methods employed in the experiment include the Random Forest (RF) classifier and AdaBoost classifier. RF classifier constructs multiple decision trees and combines their predictions to generate the final classification. AdaBoost classifier, on the other hand, is an ensemble method that iteratively adjusts the weights of training instances to focus on misclassified samples. In this experiment, it uses RandomForestClassifier as the base estimator.

The boosting algorithms used are XGBoost (XGB) and LightGBM (LGBM) classifier. XGB is a powerful gradient boosting algorithm that creates an ensemble of weak learners, sequentially correcting the mistakes made by previous learners. LGBM, another gradient boosting algorithm, focuses on speed and efficiency by employing techniques such as leaf-wise tree growth.

The performance comparison between the algorithms is advantageous for both Model 1 and Model 2. The algorithms will be trained using BERT's embeddings of the input text data. By utilizing BERT's embeddings, the algorithms can leverage the powerful contextual representations of words generated by the BERT model. This allows for a more accurate and meaningful analysis of the text data, enhancing

the performance of the algorithms in their respective tasks.

3.3.3. BERT

The BERT model utilized in this experiment is specifically the "bert_en_uncased_L-12_H-768_A-12_4" variant. This particular model is obtained from TensorHub, a popular platform for sharing and accessing pre-trained models. The chosen BERT model, being the "English uncased" version, does not differentiate between uppercase and lowercase letters in the text. The "L-12" indicates that the BERT model has 12 transformer layers, while the "H-768" refers to the hidden size of 768, indicating the dimensionality of the model's internal representations. The "A-12" signifies the presence of 12 self-attention heads in the model, enabling it to capture different aspects of context and relationships within the text. Finally, the “4” signifies this is the 4th version of the model. The selected BERT model, being the "most basic" variant, serves as a strong starting point for the experiment.

3.3.4. Objectivity and Subjectivity Classification (Model 1)

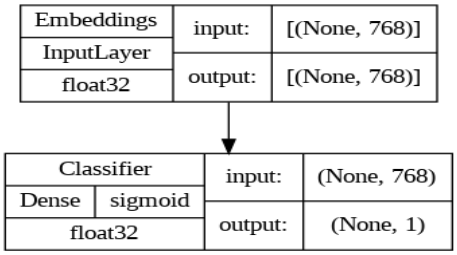

Fig.2: Bert Separate (Model 1)

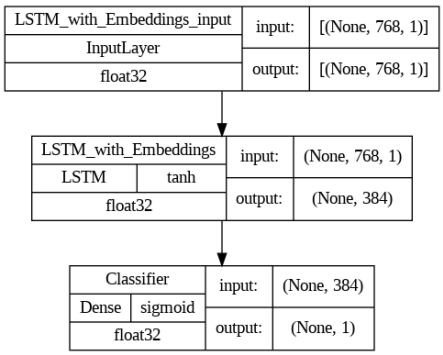

Fig.3: LSTM (Model 1)

For Model 1, the output layer of each deep learning approach consists of a dense layer with 1 unit that uses the sigmoid activation function. Figure 2 illustrates that the dense layer is connected to the separately predicted BERT embeddings, which serves as the input. Similarly, in Figure 3, the dense layer is connected to an LSTM layer with a tanh activation function which is a useful choice for hidden layers before being connected to the BERT embeddings, following a similar structure as Figure 2. In Figure 4, the approach remains the same as Figure 2, but in this case, the dense layer is directly connected to the BERT model, which takes the raw text as input. All of these approaches in Model 1 are trained for a maximum of 10 epochs, with early stopping implemented after 3 epochs based on the validation accuracy.

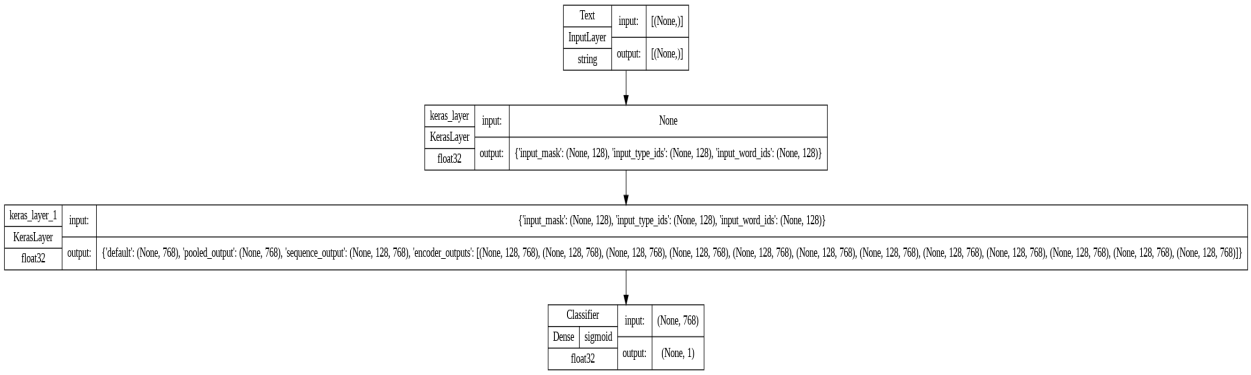

Fig.4: Bert Combine (Model 1)

3.3.5. Sentiment Analysis (Model 2)

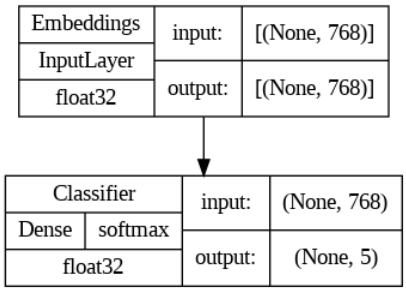

Fig.5: Bert Separate (Model 2)

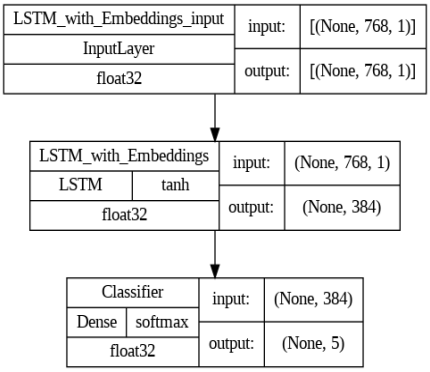

Fig.6: LSTM (Model 2)

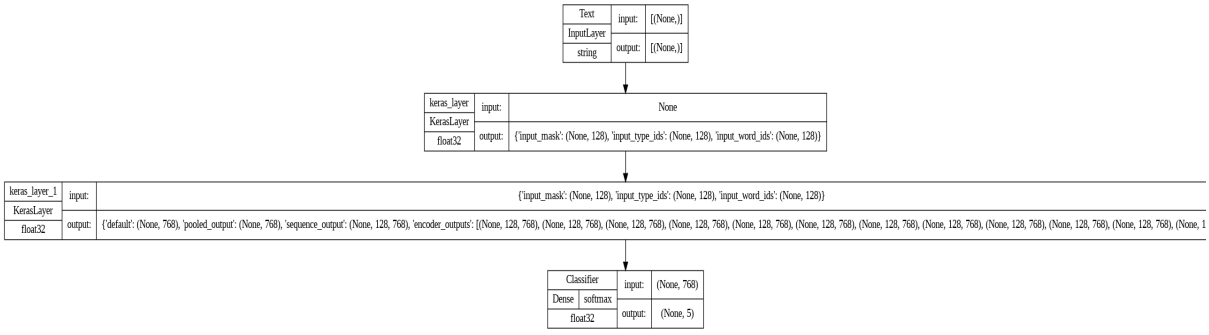

Fig.7: Bert Combine (Model 2)

For Model 2, the output layer of each deep learning approach consists of a dense layer with 5 unit that uses softmax activation function. Figure 5 illustrates that the dense layer is connected to the separately predicted BERT embeddings, which serves as the input. Similarly, in Figure 6, the dense layer is connected to an LSTM layer with a tanh activation function before being connected to the BERT embeddings, following a similar structure as Figure 5. In Figure 7, the approach remains the same as Figure 5, but in this case, the dense layer is directly connected to the BERT model, which takes the raw text as input. All the approaches in Model 2 are trained for a maximum of 30 epochs, with early stopping implemented after 5 epochs based on the validation accuracy.

3.3.6. Sentiment Analysis with Objectivity and Subjectivity Classification (Model 3)

As illustrated in Figure 8, Model 3 consist of the best performing model from both Model 1 and Model 2 combined. The purpose of Model 3 is to evaluate its performance compared to a sentiment analysis model that does not include the objectivity and subjectivity classification step. The detailed architecture and performance of Model 3 will be further explained in the Result and Discussion section.

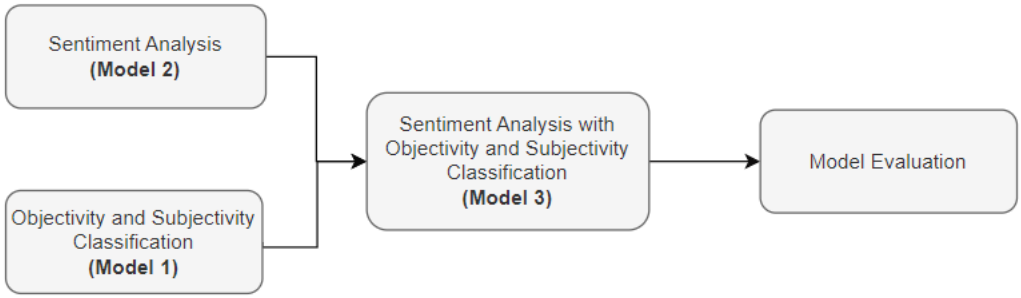

Fig.8: Process Overview (Model 3)

3.4. Model Evaluation

A confusion matrix is a vital tool for evaluating a model's performance. It presents a visual comparison between the predicted class labels and actual class labels to provide a comprehensive analysis of a classification model's performance. The confusion matrix breaks down the effectiveness of the model into various components, such as the number of true positives, true negatives, false positives, and false negatives. By using a confusion matrix, one can observe the model's performance for each class and identify any patterns of error it may make. The confusion matrix also enables the calculation of various performance metrics, including accuracy, precision, recall, and F1 score. However, since the class balance datasets is used, the focus will be on accuracy as the main metric.

4. Result and Discussion

In this section, we present the performance comparison results for each Model 1, Model 2, and Model 3. Additionally, we analyze the impact of using raw data versus preprocessed data when utilizing BERT as embeddings.

4.1. Objectivity and Subjectivity Classification (Model 1)

Table 1 shows the accuracy for raw data where all the models perform similar with an accuracy of 0.99. In table 2, the accuracy for preprocessed data where BERT Combine and Bert Separate achieve a consistent accuracy of 0.99 among 70% to 90% split train ration while Random Forest achieve the lowest accuracy among all split train ratio which is 0.97.

Table 1: Model's accuracy for raw data based on train ratio (Model 1)

Model / Train Ratio	50%	60%	70%	80%	90%
RF	0.99	0.99	0.99	0.99	0.99
AdaBoost	0.99	0.99	0.99	0.99	0.99
XGBoost	0.99	0.99	0.99	0.99	0.99
LightGBM	0.99	0.99	0.99	0.99	0.99
Linear-SVC	0.99	0.99	0.99	0.99	0.99
LSTM	0.99	0.99	0.99	0.99	0.99
Bert Separate	0.99	0.99	0.99	0.99	0.99
Bert Combine	0.99	0.99	0.99	0.99	0.99

Table 2: Model's accuracy for preprocess data based on train ratio (Model 1)

Model / Train Ratio	50%	60%	70%	80%	90%
RF	0.97	0.97	0.97	0.97	0.97
AdaBoost	0.97	0.97	0.98	0.97	0.98
XGBoost	0.99	0.99	0.99	0.99	0.99
LightGBM	0.98	0.98	0.98	0.98	0.98
Linear-SVC	0.99	0.99	0.99	0.99	0.99
LSTM	0.97	0.98	0.98	0.99	0.99
Bert Separate	0.98	0.99	0.99	0.99	0.99
Bert Combine	0.99	0.98	0.99	0.99	0.99

The results presented in Table 1 and Table 2 indicate that the performance of all models, across different train ratios, demonstrates a consistently high level of accuracy. The accuracy scores for objectivity and subjectivity classification (Model 1) of the English Wikipedia and Shopee Review dataset are very similar among all the models, varying only by around 2%. This suggests that distinguishing between objectivity and subjectivity using these datasets is relatively straightforward for the models employed in this experiment. Furthermore, the comparison between raw data (Table 1) and preprocessed data (Table 2) reveals that there is not a significant difference in accuracy. Despite the raw data achieving a remarkable 99% accuracy for all models, the performance with preprocessed data is only slightly lower. This implies that the data preprocessing step, when using BERT embeddings, does not have a substantial impact on the overall accuracy of the models in this particular case. Overall, these findings indicate that the models exhibit consistent and high accuracy in distinguishing between objective and subjective statements. Additionally, the minimal difference observed between raw and preprocessed data suggests that the performance of the models is not heavily influenced by the specific data preprocessing techniques used in this experiment.

4.2. Sentiment Analysis (Model 2)

Table 3 shows the accuracy for raw data with BERT Combine achieve the highest accuracy which is 0.46 on 80% split ratio and Random Forest achieve the lowest accuracy which is 0.4. In table 4, surprisingly LSTM achieve the lowest accuracy which is 0.21 and the highest accuracy is achieved by Linear-SVC with 0.41 on 50% split ratio.

Table 3: Model's accuracy for raw data based on train ratio (Model 2)

Model / Train Ratio	50%	60%	70%	80%	90%
RF	0.40	0.40	0.40	0.40	0.40
AdaBoost	0.41	0.42	0.42	0.42	0.42
XGBoost	0.43	0.43	0.43	0.43	0.44
LightGBM	0.44	0.43	0.44	0.44	0.44
Linear-SVC	0.43	0.46	0.46	0.45	0.45
LSTM	0.41	0.42	0.44	0.43	0.44
Bert Separate	0.41	0.45	0.43	0.46	0.45
Bert Combine	0.44	0.44	0.45	0.46	0.44

Table 4: Model's accuracy for preprocess data based on train ratio (Model 2)

Model / Train Ratio	50%	60%	70%	80%	90%
RF	0.34	0.35	0.35	0.35	0.35
AdaBoost	0.36	0.36	0.36	0.37	0.37
XGBoost	0.37	0.37	0.38	0.38	0.38
LightGBM	0.37	0.38	0.38	0.38	0.38
Linear-SVC	0.41	0.37	0.40	0.40	0.37
LSTM	0.21	0.25	0.22	0.23	0.24
Bert Separate	0.37	0.38	0.39	0.40	0.40
Bert Combine	0.36	0.38	0.39	0.37	0.40

Upon initial examination of Table 3 and Table 4, it becomes apparent that the models show an overall increase in accuracy of approximately 6% when using raw data compared to preprocessed data for sentiment analysis (Model 2). This finding indicates that text preprocessing has a noticeable impact on the accuracy of sentiment analysis models. The discrepancy in accuracy between raw and preprocessed data can be attributed to the significance of word variations in sentiment analysis. For instance, variations in words such as "good" and "gooood" can have distinct sentimental values. Preprocessing these variations may not effectively capture the nuances required for sentiment extraction when using BERT embeddings.

Moreover, when considering the LSTM model, the results indicate that preprocessing the data is highly inefficient. The accuracy of the LSTM model drops by half when using preprocessed data compared to using raw data. This suggests that the preprocessing step, in this case, hampers the LSTM model's ability to effectively capture sentiment-related patterns and nuances. In summary, the observed increase in accuracy when using raw data over preprocessed data for sentiment analysis suggests that text preprocessing can have a significant impact on sentiment classification models. The ineffective handling of word variations and the adverse effect on LSTM accuracy further emphasize the importance of considering the appropriate preprocessing techniques for sentiment analysis tasks.

4.3. Sentiment Analysis with Objectivity and Subjectivity Classification (Model 3)

4.3.1. Model Design

As outlined in the methodology section, Model 3 involves a sequential process that incorporates both Model 1 and Model 2. Figure 9 provides a detailed representation of this process. Specifically, for this case, the Bert Separate model from Figure 2 is utilized as Model 1, and the Bert Separate model from Figure 5 is used as Model 2. It is important to note that these models were trained using the raw data since it yielded higher accuracy. The rationale behind this choice is to select the model that utilizes

BERT embeddings most effectively. Among the available models, Bert Separate and Bert Combine align closest to the desired criteria of using BERT embeddings in their purest form.

However, when examining Figure 9, it becomes apparent that the average time taken for each epoch in Bert Combine is significantly longer compared to Bert Separate. This discrepancy arises because Bert Combine needs to generate embeddings for each epoch, while Bert Separate can reuse the already created embeddings. Considering these factors, Bert Separate is determined to be the optimal choice as the best model for both Model 1 and Model 2 in Model 3. This selection is based on its ability to effectively utilize BERT embeddings and its more efficient runtime, making it a suitable candidate for the sequential process of Model 3.

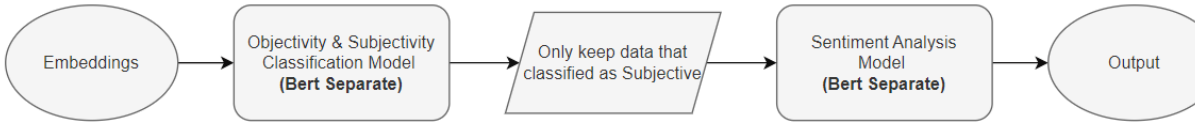

Fig.9: Sentiment analysis with objectivity and subjectivity classification (Model 3)

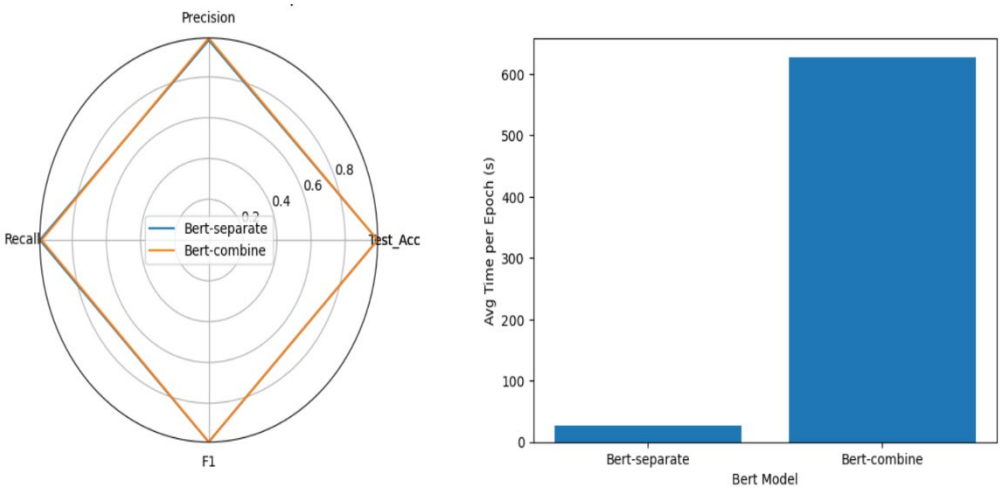

Fig.9: Time comparison between BERT Separate and BERT Combine

4.3.2. Comparison

The test is conducted using a combined dataset comprising sentiment analysis data and objectivity data. This combined dataset is designed to have ratings ranging from 0 to 5, where a rating of 0 represents the objectivity data.

Based on the results presented in Table 5, there is an approximate 7% increase in accuracy for sentiment analysis when incorporating objectivity and subjectivity classification. This finding suggests that objectivity and subjectivity classification play a significant role in improving the accuracy of sentiment analysis, particularly when exposed to objective statements. To support this evidence, the confusion matrices in Figure 10 and Figure 11 illustrate the distribution of predicted classes. Specifically, focusing on the 0 class label representing objectivity statements, it is evident that in the case of sentiment analysis without objectivity and subjectivity classification (Figure 10), almost all objectivity statements are predicted incorrectly. However, in Figure 11, which incorporates objectivity and subjectivity classification, most of the objective statements have been successfully filtered out, resulting in a considerable reduction in wrongly predicted objectivity data.

Table 5: Accuracy comparison for sentiment analysis with and without objectivity and subjectivity classification

Model	Accuracy
Sentiment Analysis	0.38
Sentiment Analysis with Objectivity and Subjectivity Classification	0.45

The result of subjectivity classification is higher than that in (Chiha & Ben Ayed, 2020) is expected since we used BERT as word embedding method which could capture the semantic of the words in a sentence. In terms of sentiment analysis, there is improvement with preceding subjectivity classification, however if compared with other research works the accuracy may not be as high. Further analysis and experiments would have to be taken into account as the datasets and models are not the same.

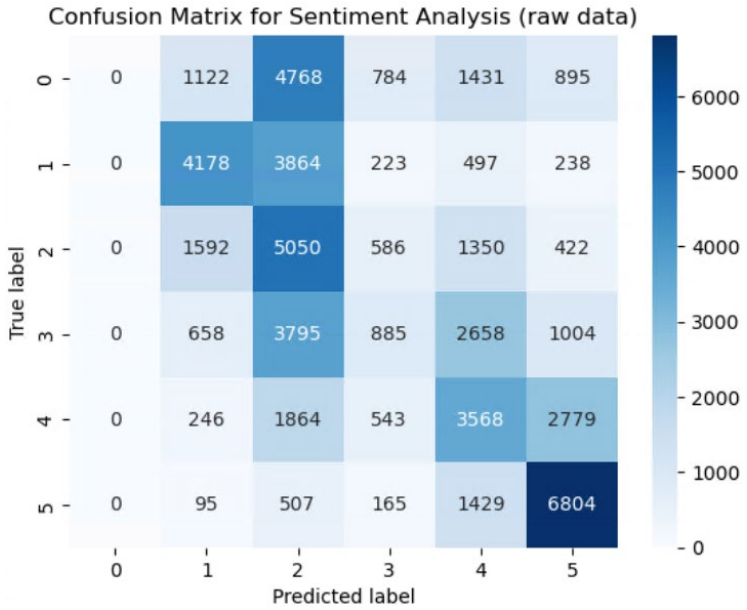

Fig.10: Confusion matrix for sentiment analysis

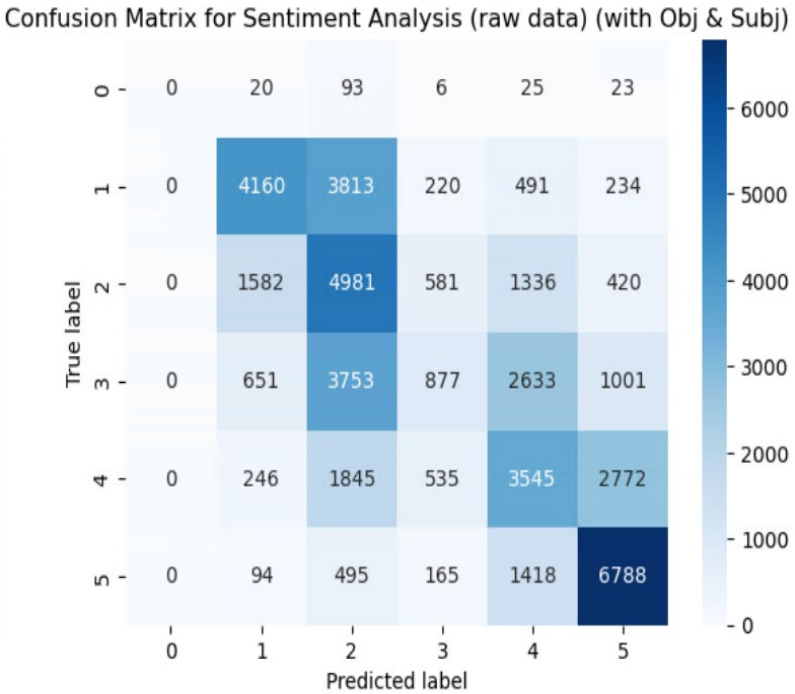

Fig.11: Confusion matrix for sentiment analysis with objectivity and subjectivity classification

These findings underscore the importance of incorporating objectivity and subjectivity classification for more accurate sentiment analysis, especially in scenarios involving objective statements. The inclusion of this classification step helps improve the overall performance of the sentiment analysis model by effectively identifying and handling objective statements separately from subjective ones.

5. Conclusions

In conclusion, the experiment showed that distinguishing between the English Wikipedia dataset and the Shopee Review dataset was relatively easy, as indicated by consistently high accuracy in Model 1. However, sentiment analysis on the Shopee Review dataset proved to be more challenging, with an accuracy of less than 50%. The experiment demonstrated that different machine learning algorithms and deep learning approaches, when using BERT embeddings, yielded similar accuracy levels. Additionally, the use of raw data was found to positively impact the accuracy of sentiment analysis when employing BERT embeddings. Importantly, the inclusion of objectivity and subjectivity classification improved the performance of sentiment analysis. This step helped differentiate objective and subjective statements, enhancing the accuracy of the sentiment analysis model.

It is important to acknowledge some limitations of the experiment. The default BERT model was used, thus not utilizing its full capacity. Future research could explore alternative transfer learning approaches for word embeddings to achieve similar results. Additionally, the subjectivity data was derived only from the Shopee Review dataset, highlighting the need for further investigation using diverse review data for generalizability. Additional real-world data from diverse sources would boost generalizability. Overall, the experiment provided valuable insights into different aspects of the models and highlighted the significance of objectivity and subjectivity classification in sentiment analysis. These findings pave the way for future advancements in utilizing BERT and exploring other word embedding techniques in NLP tasks. Furthermore, testing other contextual embeddings like ELMo for this approach is also a future direction.

Acknowledgements

This research is supported by Fundamental Research Grant Scheme (FRGS), FRGS/1/2022/ICT06/MMU/03/3, and TM Research & Development Grant (TM R&D), MMUE/220028.

References

Agarwal, N. (2022). *The Ultimate Guide To Different Word Embedding Techniques In NLP*. Retrieved from The Ultimate Guide To Different Word Embedding Techniques In NLP: https://www.kdnuggets.com/2021/11/guide-word-embedding-techniques-nlp.html

Alyafeai, Z., AlShaibani, M. S., & Ahmad, I. (2020). A Survey on Transfer Learning in Natural Language Processing. *A Survey on Transfer Learning in Natural Language Processing*. arXiv. doi:10.48550/ARXIV.2007.04239

Bajaj, D., Goel, A., Gupta, S., & Batra, H. (2022, February). MUCE: a multilingual use case model extractor using GPT-3. *International Journal of Information Technology, 14*, 1-12. doi:10.1007/s41870-022-00884-2

Biradar, G. R., Raagini, J. M., Varier, A., & Sudhir, M. (2019). Classification of Book Genres using Book Cover and Title. *2019 IEEE International Conference on Intelligent Systems and Green Technology (ICISGT)*, (pp. 72-723). doi:10.1109/ICISGT44072.2019.00031

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. *Language Models are Few-Shot Learners*. arXiv. doi:10.48550/ARXIV.2005.14165

Chiha, R., & Ben Ayed, M. (2020). Supervised Machine Learning Approach for Subjectivity/Objectivity Classification of Social Data. In M. Themistocleous, & M. Papadaki (Ed.), *Information Systems* (pp. 193–205). Cham: Springer International Publishing.

CHIHAB, M., CHINY, M., Mabrouk, N., BOUSSATTA, H., CHIHAB, Y., & Moulay Youssef, H. A. (2022). BiLSTM and Multiple Linear Regression based Sentiment Analysis Model using Polarity and Subjectivity of a Text.

Cobos, R., Jurado, F., & Blázquez-Herranz, A. (2019). A Content Analysis System That Supports Sentiment Analysis for Subjectivity and Polarity Detection in Online Courses. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, 14*, 177-187. doi:10.1109/RITA.2019.2952298

Dehouche, N. (2021). Plagiarism in the age of massive Generative Pre-trained Transformers (GPT-3). *Ethics in Science and Environmental Politics, 21*, 17–23.

Devi, S., & Sivakumar, S. (2021, September). An efficient contextual glove feature extraction model on large textual databases. *International Journal of Speech Technology, 25*, 1-10. doi:10.1007/s10772-021-09884-2

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv. doi:10.48550/ARXIV.1810.04805

Foundation, W. (n.d.). *Wikimedia Downloads*. Retrieved from https://dumps.wikimedia.org

Gupta, H., & Patel, M. (2020, October). Study of Extractive Text Summarizer Using The Elmo Embedding., (pp. 829-834). doi:10.1109/I-SMAC49090.2020.9243610

India, & Gupta, S. (2022, May). Hate Speech Detection using OpenAI and GPT-3. *12*. doi:10.46338/ijetae0522_15

Kasnesis, P. a. (2021). Combating Fake News with Transformers: A Comparative Analysis of Stance Detection and Subjectivity Analysis. *Information*, 409.

Kolesnikova, O., Gelbukh, A., Pinto, D., Singh, V., & Perez, F. (2020, January). A Study of Lexical Function Detection with Word2vec and Supervised Machine Learning. *J. Intell. Fuzzy Syst., 39*, 1993–2001. doi:10.3233/JIFS-179866

Kulkarni, C., Monika, P., Shruthi, S., Bharadwaj, D., & Uday, D. (2022). COVID-19 Fake News Detection Using GloVe and Bi-LSTM. *Proceedings of Second International Conference on Sustainable Expert Systems*, (pp. 43–56).

Li, W., Gao, S., Zhou, H., Huang, Z., Zhang, K., & Li, W. (2019). The automatic text classification method based on bert and feature union. *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*, (pp. 774–777).

Liu, W., Wen, B., Gao, S., Zheng, J., & Zheng, Y. (2020). A multi-label text classification model based on ELMo and attention. *MATEC Web of Conferences*, *309*, p. 03015.

Madichetty, S., & Sridevi, M. (2020). Improved classification of crisis-related data on Twitter using contextual representations. *Procedia Computer Science, 167*, 962–968.

Maslennikova, E. (2019). ELMo Word Representations For News Protection. *CLEF (Working Notes).*

Mikolov, T. C. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Mohamed, E. (2021, January). An Ensemble Multi-label Themes-Based Classification for Holy Qur'an Verses Using Word2Vec Embedding. *ARABIAN JOURNAL FOR SCIENCE AND ENGINEERING, 46*. doi:10.1007/s13369-020-05184-0

Moraes, L., Marcacini, R. M., & Goularte, R. (2022). Video Summarization Using Text Subjectivity Classification. *Proceedings of the Brazilian Symposium on Multimedia and the Web* (pp. 133–141). New York, NY, USA: Association for Computing Machinery. doi:10.1145/3539637.3556998

Nhan Cach Dang, M. N.-G. (2020). Sentiment Analysis Based on Deep Learning: A Comparative Study. *Electronics, 9*, 483.

Özçift, A., Akarsu, K., Yumuk, F., & Söylemez, C. (2021). Advancing natural language processing (NLP) applications of morphologically rich languages with bidirectional encoder representations from transformers (BERT): an empirical case study for Turkish. *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije, 62*, 226–238.

Park, Y., Hauschild, A.-C., & Heider, D. (2021). Transfer learning compensates limited data, batch effects and technological heterogeneity in single-cell sequencing. *NAR genomics and bioinformatics, 3*, lqab104.

Pennington, J. S. (2014). Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* , pp. 1532-1543.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *Deep contextualized word representations*. arXiv. doi:10.48550/ARXIV.1802.05365

Qaiser, S., & Ali, R. (2018, July). Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. *International Journal of Computer Applications, 181*. doi:10.5120/ijca2018917395

Ranade, A., Telge, S., & Mate, Y. (2022, January). Predicting Disasters from Tweets Using GloVe Embeddings and BERT Layer Classification. doi:10.1007/978-3-030-95502-1_37

Saravanan, S., & Sudha, K. (2022). GPT-3 Powered System for Content Generation and Transformation. *2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT)*, (pp. 514-519). doi:10.1109/CCiCT56684.2022.00096

Satapathy, R., Pardeshi, S. R., & Cambria, E. (2022). Polarity and subjectivity detection with multitask learning and bert embedding. *Future Internet, 14*, 191.

Sharma, A. K., Chaurasia, S., & Srivastava, D. K. (2020). Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec. *Procedia Computer Science, 167*, 1139-1147. doi:https://doi.org/10.1016/j.procs.2020.03.416

Shopee. (2022). *Shopee Code League 2022*. Retrieved from https://careers.shopee.sg/codeleague/

Ulčar, M., & Robnik-Šikonja, M. (2022). Cross-lingual alignments of ELMo contextual embeddings. *Neural Computing and Applications*, 1–19.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention Is All You Need. *Attention Is All You Need*. arXiv. doi:10.48550/ARXIV.1706.03762

Wadud, M. A., Mridha, M., Shin, J., Nur, K., & Saha, A. K. (2022). Deep-BERT: Transfer Learning for Classifying Multilingual Offensive Texts on Social Media. *Comput. Syst. Sci. Eng, 44*, 1775–1791.

Xu, S., Zhang, C., & Hong, D. (2022). BERT-based NLP techniques for classification and severity modeling in basic warranty data study. *Insurance: Mathematics and Economics, 107*, 57–67.

Yap, T., & Chandar, E. (2023). My Paper. *My Journal, 1*(1), 20-30.

Yu, C. X., Ying, S., Feng, G., & Min, Z. X. (2022). Research on the Classification Modeling for the Natural Language Texts with Subjectivity Characteristic.

Zhang T, G. X. (2022). BMT-Net: Broad Multitask Transformer Network for Sentiment Analysis. *IEEE Trans Cybern*, 6232-6243.

Zhao, Z., Zhang, Z., & Hopfgartner, F. (2022). Utilizing subjectivity level to mitigate identity term bias in toxic comments classification. *Online Social Networks and Media, 29*, 100205.

Zong, M., & Krishnamachari, B. (2022). a survey on GPT-3. *arXiv preprint arXiv:2212.00857*.