FMDB ***Transactions on Sustainable Computing Systems***

Deciphering Product Review Sentiments Using BERT and TensorFlow

D. Suraj[1, *], S. Dinesh[2], R. Balaji[3], P. Deepika[4], Freddy Ajila[5]

[1,2,3,4]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[5]Department of Information Technology, Faculty of Computer Science and Electronics, Higher Polytechnic School of Chimborazo (ESPOCH), Orellana Headquarters, El Coca, Ecuador.
dr0604@srmist.edu.in[1], ds3511@srmist.edu.in[2], br1371@srmist.edu.in[3], deepikap2@srmist.edu.in[4], freddy.ajila@espoch.edu.ec[5]

Abstract: Using BERT and TensorFlow in analyzing product reviews is important in understanding customer sentiment. After careful fine-tuning with a large dataset of reviews, BERT becomes adept at categorizing sentiments as positive or negative. This breakthrough benefits businesses by improving their marketing strategy and product development and empowers consumers to make better decisions. The sentiment analysis pipeline integrates seamlessly with a refined BERT model. The BERT model is the key to predicting sentiment polarity. Based on its sensitive insights, BERT classifies reviews by providing a binary difference between positive and negative sentiments. In order to rigorously evaluate the model's performance, we use several key metrics. These metrics comprehensively evaluate the model's effectiveness, ensuring reliable results. This research project cannot be overemphasized. It changes the way businesses gauge customer opinions. It serves as a compass to fine-tune marketing strategies and improve product quality. This improves customer satisfaction, strengthening the relationship between businesses and consumers. This research bridges the gap between NLP and sentiment analysis, demonstrating the huge potential for improving sentiment analysis performance and ushering in a new age of cutting-edge applications in NLP. The outcome of this project will be revolutionary. It will revolutionize how businesses understand customer feedback, translate it into more informed decisions, build stronger customer relationships, and deliver better products. In short, it will revolutionize the customer-centric landscape and take the business-consumer relationship to the next level.

Keywords: Sentiment Analysis; Machine Learning; Deep Learning; Artificial Intelligence; Product Reviews; Long Short-Term Memory; Neural Networks; Natural Language Processing (NLP); Recurrent Neural Networks.

Received on: 16/12/2022, **Revised on:** 21/02/2023, **Accepted on:** 26/03/2023, **Published on:** 17/04/2023

Cited by: D. Suraj, S. Dinesh, R. Balaji, P. Deepika, and F. Ajila, "Deciphering Product Review Sentiments Using BERT and TensorFlow," *FMDB Transactions on Sustainable Computing Systems*., vol. 1, no. 2, pp. 77–88, 2023.

1. Introduction

Customer feedback is an essential resource for any business. It offers a direct line of communication between a business and its customers, allowing them to gain insight into their customers' needs, preferences and pain points. Companies actively seeking and listening to customer feedback can make more informed decisions to enhance their offerings, services and customer experience [11]. Not only does customer feedback highlight areas that need improvement, but it also reinforces what the business is doing well. Positive customer feedback can improve employee morale and confidence, whereas constructive criticism can provide a roadmap for expansion and innovation [12]. Customer feedback can also foster a sense of engagement and loyalty among customers, as they feel heard and appreciated when their opinion is considered [13]. Customer feedback can have a positive impact on customer satisfaction and brand loyalty, as well as long-term success. It is an invaluable asset for

[*]Corresponding author.

businesses that want to remain competitive in a rapidly evolving market while responding to the target audience's changing needs [14].

Sentiment analysis is some powerful tool businesses use to analyze and interpret customer sentiment behind product reviews. NLP and sentiment analysis algorithms are powerful tools that allow for the automated extraction and comprehension of sentiment expressed in text, allowing for the processing of large volumes of customer reviews in a timely and precise manner [15]. Machine learning models can categorize feedback into positive, negative, and neutral sentiments and detect nuances in tone of voice, sarcasm, and context, enabling businesses to gain valuable insights. Sentiment analysis technology can be used to track customer sentiment over time, evaluate the effectiveness of marketing campaigns, monitor customer reaction to product updates, and identify areas for improvement [16]. It is an essential tool in managing reputation, identifying emerging issues, and adapting products and services to align with customer preferences in the digital age. Using sentiment analysis technology, businesses can make informed decisions to improve product quality and customer experience [17].

The insights gained from product reviews can inform product development and refine strategies. This is why sentiment analysis is essential to NLP [1]. This project dives deep into sentiment analysis with the help of advanced technologies like BERT [2] and deep learning frameworks like TensorFlow [18]. With the emergence of cutting-edge NLP models, such as BERT, and the transformative power of the deep learning framework TensorFlow, we can now explore the complexities of product reviews more deeply and comprehensively than ever. In today's digital world, where billions of text messages are exchanged daily, understanding the sentiment behind these reviews is essential [19]. The goal of our project is to leverage the power of BERT, as well as the flexibility offered by TensorFlow, to build a powerful sentiment analysis framework for product reviews. The primary goal of this sentiment analysis framework is to categorize product reviews into positive and negative sentiment categories to help businesses understand consumer feedback, make data-driven decisions, and ultimately improve products and services [20].

So, this project mainly focuses on using technology to contribute to businesses and their customers. So, the project's main aim is to develop a methodology that can classify product reviews into positive or negative sentiment categories. There are many ways, and this can be implemented with the help of various technology [21]. Machine learning algorithms can be used to analyze large datasets of existing texts and identify patterns and structures that can be used to classify the reviews. Many technologies are used for Sentiment Analysis that has been researched [22]. A lexicon-based approach to sentiment analysis relies on predefined dictionaries or lexicons that associate words with sentiment scores [23].

In this method, each word in a piece of text is assigned a sentiment polarity (positive, negative, or neutral), and the overall sentiment of the text is determined by aggregating these individual scores. This approach assumes that the sentiment of a sentence or document can be inferred by summing or averaging the sentiments of its constituent words [4]. Weighted word embeddings refer to an extension of traditional word embedding techniques. Each word's contribution to the overall representation is weighted based on its importance or relevance to a specific task. In the context of sentiment analysis, this approach aims to enhance the ability of word embeddings to capture the sentiment-carrying nuances of words within a given piece of text [5]. Support Vector Machines (SVM) and Naive Bayes classifiers are two common machine-learning algorithms used for sentiment analysis [6].

Another method to perform sentiment analysis is by creating word2vec. Word2Vec is a popular word embedding technique used in sentiment analysis to enhance the representation of words in a continuous vector space [7]. In the context of sentiment analysis, Word2Vec transforms words into dense vectors, capturing semantic relationships and contextual nuances. This is achieved by training a neural network to predict the context of words based on their surrounding words in a large corpus [24]. The resulting word embeddings encode similarities between words, allowing for a more nuanced understanding of their meaning. In sentiment analysis, these embeddings are often used as features for machine learning models [8].

The objectives are comprehensive and multifaceted. First, we delve into the exploration and preprocessing of the sentiment analysis dataset, addressing its unique characteristics and preparing it for subsequent model training [9]. Next, we detail the architecture and configuration of the sentiment analysis model, customizing it to suit the specific nuances of product reviews. Subsequently, we undertake a rigorous training and evaluation process, employing established metrics to assess the model's performance in sentiment classification.

Finally, a comparative analysis is conducted, aligning our findings with existing literature in sentiment analysis to delineate the distinctive contributions and potential areas for improvement [25]. As we navigate through the various facets of this project in the subsequent sections, beginning with a comprehensive examination and preparation of the sentiment analysis dataset, the overarching goal is to contribute insights and advancements to the evolving landscape of sentiment analysis research [26].

2. Existing System

The current state of sentiment analysis systems exhibits a broad spectrum of methodologies, reflecting the continual evolution of natural language processing (NLP). Historically, sentiment analysis often relied on rule-based systems, where predetermined linguistic rules and patterns were employed to categorize textual data into sentiment classes [27]. These rule-based approaches, though foundational, faced limitations in adapting to the dynamic and nuanced nature of natural language, leading to challenges in accurately capturing the complexity of sentiments expressed [28].

A paradigm shift has occurred in recent years with the ascendancy of machine learning techniques in sentiment analysis. Supervised learning models, such as Support Vector Machines (SVM) and Naive Bayes classifiers, gained popularity due to their ability to learn sentiment patterns from annotated training data. However, these models often grappled with the intricate nuances of sentiment expression and exhibited limitations in generalizing effectively to diverse datasets [29].

A pivotal breakthrough emerged by introducing deep learning models, particularly those rooted in transformer architectures like BERT (Bidirectional Encoder Representations from Transformers). BERT, introduced by Devlin et al. in 2018, marked a revolutionary advancement by leveraging bidirectional context to comprehensively capture semantic information within textual data. This bidirectional understanding empowers BERT-based models to excel in discerning sentiment's intricate and context-dependent nature in natural language [30].

The studies conducted by researchers such as Mohbey [1], Biswas et al., [2], and Alaparthi and Mishra [3] underscore the transformative shift from conventional sentiment analysis approaches to the adoption of sophisticated deep learning techniques. These investigations highlight the effectiveness of transformer-based models, particularly BERT, in achieving robust sentiment analysis outcomes across diverse domains. Despite the strides made by existing systems, the escalating intricacy of user-generated content and the demand for fine-grained sentiment analysis necessitate ongoing research and innovation to propel the field forward.

3. Proposed System

The proposed sentiment analysis system constitutes a substantial leap forward from conventional methodologies, harnessing cutting-edge deep learning techniques, particularly emphasising the BERT (Bidirectional Encoder Representations from Transformers) architecture. Our approach aims to surmount the limitations of traditional rule-based systems and address the shortcomings of earlier machine learning models by capitalizing on the contextual understanding capabilities inherent in transformer-based architectures [31].

At the heart of our proposed system lies the integration of BERT, a pre-trained transformer model renowned for its proficiency in capturing bidirectional contextual information within textual data. BERT's capacity to discern nuanced relationships between words and phrases positions it as an optimal choice for sentiment analysis, where contextual nuances play a pivotal role in accurately assessing sentiment orientation [32]. By incorporating BERT into our sentiment analysis framework, we aim to attain a more sophisticated and nuanced understanding of sentiment in product reviews.

The architecture of the proposed system encompasses a series of preprocessing steps meticulously designed to optimize the input data for BERT-based analysis [33]. This includes tokenization, attention mask generation, and encoding of input sequences. Our system is engineered to dynamically adapt to varying review lengths, ensuring flexibility in handling diverse textual inputs. Furthermore, a deep neural network, specifically TFBertForSequence Classification, is employed to fine-tune the pre-trained BERT model for the specific task of sentiment analysis [34].

The training process involves a rigorous evaluation of model performance on labelled datasets, underscoring the significance of hyperparameter tuning to optimize for accuracy and generalization [35]. Leveraging TensorFlow and the Transformers library, our system capitalizes on the efficiency of GPU acceleration, ensuring prompt convergence during training [36].

The proposed sentiment analysis system aligns with the broader objectives of enhancing sentiment analysis methodologies, drawing inspiration from various impactful research contributions. Works such as those by Mohbey [1], Biswas et al., [2], Alaparthi and Mishra [3], Kurniasari and Setyanto [8], and Munikar et al., [10] exemplify the transformative potential demonstrated by transformer-based models in diverse contexts. Through meticulous experimentation and evaluation, we anticipate contributing valuable insights to sentiment analysis, addressing the unique challenges posed by product reviews and showcasing the capabilities of advanced deep-learning architectures.

4. Methodology

Figure 1 represents the architecture diagram. The intricate architecture comprises several key components, each pivotal in understanding user sentiments comprehensively. Commencing with a diverse dataset of product reviews, the project initiates the pre-processing phase, ensuring data uniformity and readiness for analysis [37]. The deployment of the powerful BERT (Bidirectional Encoder Representations from Transformers) layer stands as a cornerstone, enabling the extraction of contextual relationships within the text [38]. The subsequent conversion of BERT outputs readies the data for training, where a labelled subset is utilized to teach the sentiment analysis model [39].

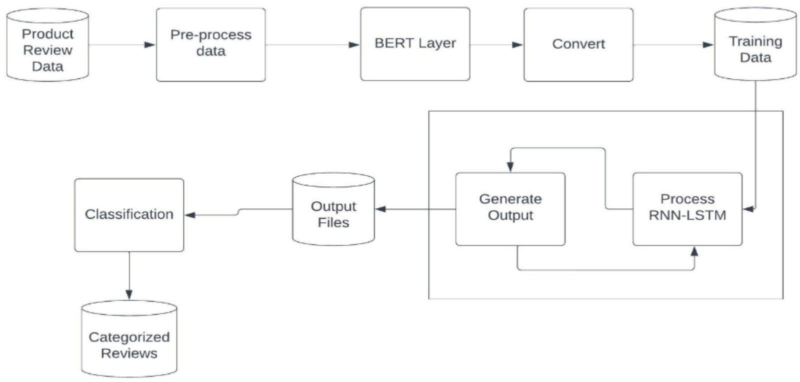

Figure 1: Architecture Diagram

The trained model then performs sentiment classification on unseen reviews, generating structured output files. In addition to BERT, a Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) architecture is employed to capture sequential patterns within the reviews [40]. The project culminates in categorising reviews based on sentiment labels, offering a nuanced understanding of customer perceptions for businesses to glean valuable insights and effectively refine their products or services [41]. The project aims to significantly contribute to sentiment analysis on product reviews through this holistic approach.

The workflow diagram in Figure 2 commences with the "Collection of Product Review Data," the project acquires a diverse dataset encompassing a spectrum of user sentiments across various products. This dataset serves as the foundational input for subsequent analysis [42]. Moving to the "Pre-process the Data" stage, a meticulous pre-processing routine is applied, involving tasks such as text cleaning and normalization. This step ensures the uniformity and cleanliness of the data, paving the way for more effective analysis.

The pre-processed data is then channeled into the "Transfer to BERT" phase, where the powerful Bidirectional Encoder Representations from Transformers (BERT) architecture comes into play. BERT's contextual understanding capabilities are harnessed to extract nuanced features from the textual data. The output from BERT is then directed towards "Training the Data Using RNN-LSTM," where a Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) architecture is implemented. This step allows the model to capture temporal dependencies and sequential patterns within the reviews, enhancing its ability to discern intricate relationships within the data.

These preparatory steps culminate in the pivotal "Implementing the Model" stage. Here, the integrated model, enriched by both BERT and RNN-LSTM, undergoes training on the pre-processed and contextualized data. The training process refines the model's understanding of sentiment dynamics, enabling it to make accurate predictions on unseen reviews. The implemented model is poised to provide businesses valuable insights into customer sentiments, facilitating informed decision-making and enhancing overall customer satisfaction.

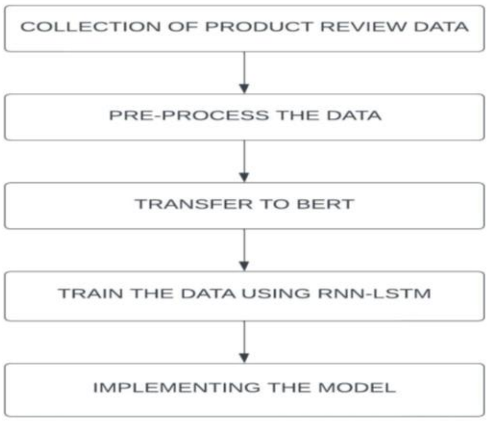

Figure 2: Workflow Diagram

5. Module Description

The entire process is divided into three modules.

5.1. Module 1: Data collection and preprocessing

The research's inception involves a meticulous approach to data collection, where a diverse set of product reviews is systematically amassed to form the raw material for sentiment analysis. This collected dataset then undergoes rigorous pre-processing involving text cleaning, normalization, and tokenization tasks. These processes ensure the uniformity and quality of the data, laying the foundation for subsequent stages in the workflow. The concerted efforts in data collection and pre-processing are pivotal, as they not only shape the integrity of the dataset but also play a crucial role in refining the model's ability to discern nuanced sentiments within product reviews, ultimately contributing to the project's overarching goal of advancing sentiment analysis (figure 3).

	A	B
1	review	sentiment
2	My lovely	2
3	Despite th	2
4	I bought th	1
5	Check out	2
6	Reviewed	2
7	I also bega	1
8	I love the s	1
9	I cannot sc	1
10	Exotic tale	2
11	Firstly,I enj	1
12	I currently	2
13	If you wan	1
14	This game	1
15	I have bee	2
16	Was hopin	1
17	First of all,	1
18	This is an A	2
19	For the pri	2
20	Y ESTE LIB	2
21	This is an e	2
22	I've read tl	1
23	This produ	1
24	The magaz	2
25	I believe I	2
26	If you purc	1
27	Like many	1
28	i bought th	2

Figure 3: Raw Data

Following data collection, the focus shifts to the “Pre-process the Data” stage, where the acquired dataset undergoes meticulous transformations. Textual data is cleaned thoroughly, including removing irrelevant characters, correcting typos, and standardizing text formats. Additionally, normalization techniques are applied to ensure consistency in representing words and phrases. Tokenization breaks down the textual data into smaller units, facilitating more granular analysis. This rigorous pre-processing not only enhances the quality and uniformity of the dataset but also prepares it for effective analysis by subsequent components in the workflow, such as the BERT layer and the RNN-LSTM model (Figure 4).

```
#preprocessing the data
TAG_RE = re.compile(r'<[^>]+>')

def preprocessing(df):

    #remove stop words
    df['review'] = df['review'].apply(lambda x: ' '.join([word for word in x.split() if word not in (stop)]))


    #remove puncaution marks
    df["review"] = df["review"].str.replace('[{}]'.format(string.punctuation), '')


    #converting to lower case
    df["review"] = df["review"].str.lower()

    def remove_tags(text):

      return TAG_RE.sub('', text)

    def remove_special_characters(text, remove_digits=True):
        pattern=r'[^a-zA-z0-9\s]'
        text=re.sub(pattern,'',text)
        return text
        #Apply function on review column
    df['review']=df['review'].apply(remove_special_characters)
    df['review'] = df['review'].apply(remove_tags)


preprocessing(df)
```

Figure 4: Data Preprocessing

5.2. Module 2: Model Creation

Creating the sentiment analysis model in this project involves a sophisticated and multifaceted approach, combining advanced natural language processing techniques to enhance the model’s understanding of nuanced user sentiments. The Bidirectional Encoder Representations from Transformers (BERT) layer is at the model’s core and renowned for capturing contextual relationships within textual data. BERT serves as a dynamic feature extractor, decoding the intricacies of language by considering the surrounding context of each word. This contextualized output forms a crucial foundation for the subsequent stages of the model creation process.

The contextualized features extracted by BERT are seamlessly integrated into a Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) architecture, introducing a temporal dimension to the model. The RNN-LSTM component captures sequential patterns and long-term dependencies within the product reviews. This is particularly essential in sentiment analysis, where understanding the temporal flow of language can provide valuable insights into the evolution of user opinions throughout a review.

The model is fine-tuned through an iterative training process, where it learns to optimize its parameters for accurate sentiment classification. Leveraging a labelled subset of the pre-processed data, the model refines its understanding of the diverse sentiments expressed in the reviews. The training process involves adjusting the weights of the model’s connections to minimize the difference between predicted and actual sentiments. This iterative learning approach ensures that the model becomes adept at discerning sentiment nuances within the dataset.

The synergy between the BERT layer and the RNN-LSTM architecture is a distinguishing feature of this sentiment analysis model. While BERT excels in capturing contextual semantics, the RNN-LSTM augments this by capturing sequential dependencies, creating a comprehensive understanding of the reviews’ linguistic intricacies. This detailed model creation process aims to provide businesses with a powerful tool for extracting nuanced insights from user-generated content, enabling them to make informed decisions and enhance overall customer satisfaction.

5.3. Module 3: Model Training

The model training process in this project represents a pivotal phase, where the sentiment analysis model refines its parameters and hones its understanding of user sentiments within the product reviews. Commencing with preparing a meticulously pre-processed dataset, a subset of labelled data is strategically chosen to train the model. This dataset is divided into batches, ensuring efficient processing and resource utilization during training. The selected subset is then subjected to an iterative training regimen, wherein the model adjusts its internal parameters to minimize the difference between predicted and actual sentiments.

In this case, the Adam optimizer's chosen optimisation algorithm facilitates the gradual adjustment of the model's weights, ensuring that it converges towards an optimal configuration for accurate sentiment classification. The learning rate, set at 2e-5, governs the step size in each iteration, striking a balance between the model's adaptability to new information and stability during training. The Sparse Categorical Cross entropy loss function is employed to quantify the dissimilarity between predicted and actual sentiments, serving as a guide for the model to fine-tune its parameters (Figure 5).

```
# train dataset
start=time.time()
ds_train_encoded = encode_examples(ds_train).shuffle(10000).batch(batch_size)
print("Done with Training Dataset",time.time()-start)
# test dataset
start=time.time()
ds_test_encoded = encode_examples(ds_test).batch(batch_size)
print("Done with Testing Dataset",time.time()-start)
```

```
2023-10-23 16:25:03.788145: W tensorflow/tsl/platform/profile_utils/cpu_utils.cc:128] Failed to get CPU frequency: 0 Hz
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicit
ly truncate examples to max length. Defaulting to 'longest_first' truncation strategy. If you encode pairs of sequences (GLUE-s
tyle) with the tokenizer you can select this strategy more precisely by providing a specific strategy to `truncation`.
/Users/hemalathat/anaconda3/envs/testenv/lib/python3.11/site-packages/transformers/tokenization_utils_base.py:2606: FutureWarni
ng: The `pad_to_max_length` argument is deprecated and will be removed in a future version, use `padding=True` or `padding='lon
gest'` to pad to the longest sequence in the batch, or use `padding='max_length'` to pad to a max length. In this case, you can
give a specific length with `max_length` (e.g. `max_length=45`) or leave max_length to None to pad to the maximal input size of
the model (e.g. 512 for Bert).
  warnings.warn(
```

```
Done with Training Dataset 58.91490578651428
Done with Testing Dataset 59.07609510421753
```

```
# recommended learning rate for Adam 5e-5, 3e-5, 2e-5
learning_rate = 2e-5
# we will do just 1 epoch, though multiple epochs might be better as long
# as we will not overfit the model
number_of_epochs = 1
```

```
model = TFBertForSequenceClassification.from_pretrained('bert-base-uncased')
```

```
Downloading model.safetensors: 100%|██████████| 440M/440M [00:14<00:00, 29.6MB/s]
All PyTorch model weights were used when initializing TFBertForSequenceClassification.

Some weights or buffers of the TF 2.0 model TFBertForSequenceClassification were not initialized from the PyTorch model and are
newly initialized: ['classifier.weight', 'classifier.bias']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
```

Figure 5: Training Module

The model's performance is assessed through relevant metrics, particularly on Sparse Categorical Accuracy. This metric gauges the model's ability to accurately classify sentiments within the training data. The training process is executed for a specified number of epochs, with careful consideration given to avoiding overfitting. In this scenario, the model becomes overly tailored to the training data, compromising its ability to generalize to unseen data. The model's progress and performance are monitored throughout the training process, and a validation dataset is employed to assess its generalization capabilities. This dataset, distinct from the training set, serves as an independent benchmark, enabling the model to showcase its ability to make accurate predictions on previously unseen reviews. The interplay of these components in the model training process reflects a meticulous and iterative approach, aiming to enhance the model's proficiency in discerning nuanced sentiments within product reviews and ensuring its adaptability to diverse user-generated content.

6. Efficiency of this Model

The efficiency of the sentiment analysis model is not confined to its training data; rather, it extends to its adaptability and generalization capabilities. With its pre-trained contextual embeddings, the integration of BERT empowers the model to understand diverse linguistic patterns and sentiments. This adaptability proves instrumental in handling a wide array of user-generated content, ensuring that the model remains effective across different product domains, industries, and linguistic nuances. The model's proficiency in generalizing to previously unseen data positions it as a versatile tool for real-world applications where user sentiments vary widely. Beyond its predictive accuracy, the model's efficiency is also reflected in its computational efficacy and scalability. The careful orchestration of optimization algorithms, such as the Adam optimizer, with fallback mechanisms for specific architectures, contributes to a streamlined training process. The dynamic learning rate expedites convergence during training and allows for efficient adaptation to various datasets. This computational efficiency is pivotal for scalability, enabling the model to handle large volumes of data efficiently. The model's ability to efficiently process and analyze vast datasets positions it as a robust solution for applications demanding real-time or batch sentiment analysis, catering to diverse industry needs.

7. Implementation

Figure 6 shows the implementation of the model to enhance the sentiment analysis model's understanding of sequential patterns within reviews; Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) units is incorporated. This sequential analysis layer is instrumental in capturing dependencies between words that extend beyond a fixed context window. The recurrent nature of RNN allows the model to maintain an internal state, enabling it to consider the entire sequence of words in a review. LSTM units further contribute by addressing the vanishing gradient problem, allowing the model to effectively capture long-term dependencies.

```
In [15]: # recommended learning rate for Adam 5e-5, 3e-5, 2e-5
         learning_rate = 2e-5
         # we will do just 1 epoch, though multiple epochs might be better as long
         # as we will not overfit the model
         number_of_epochs = 1

In [16]: model = TFBertForSequenceClassification.from_pretrained('bert-base-uncased')

         Downloading model.safetensors: 100%|████████| 440M/440M [00:14<00:00, 29.6MB/s]
         All PyTorch model weights were used when initializing TFBertForSequenceClassification.

         Some weights or buffers of the TF 2.0 model TFBertForSequenceClassification were not initialized from the PyTorch model and are
         newly initialized: ['classifier.weight', 'classifier.bias']
         You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.

In [17]: optimizer = tf.keras.optimizers.Adam(learning_rate=learning_rate, epsilon=1e-08)
         loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True)
         metric = tf.keras.metrics.SparseCategoricalAccuracy('accuracy')
         model.compile(optimizer=optimizer, loss=loss, metrics=[metric])

         WARNING:absl:At this time, the v2.11+ optimizer `tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the legacy Ker
         as optimizer instead, located at `tf.keras.optimizers.legacy.Adam`.
         WARNING:absl:There is a known slowdown when using v2.11+ Keras optimizers on M1/M2 Macs. Falling back to the legacy Keras optim
         izer, i.e., `tf.keras.optimizers.legacy.Adam`.
```

Figure 6: Implementation of the model

This integration brings a temporal dimension to the model, enabling it to discern sentiment nuances that rely on the sequential arrangement of words in a sentence. Upon successful training and fine-tuning, the model is deployed to categorize user reviews based on sentiment. The output is generated as categorized reviews, tagging each review as positive, negative, or neutral. This categorization facilitates a clear understanding of the sentiment distribution within the dataset. Moreover, the model's predictions can be utilized for downstream applications such as recommendation systems, customer feedback analysis, and market trend forecasting. The ability to generate meaningful and categorized output positions the implemented sentiment analysis model as a valuable tool for extracting actionable insights from user-generated content (Figure 7).

```
# Test model on two individual reviews
test_reviews = ['This was an awesome movie. I watch it twice my time watching this beautiful movie if I have known it was t
                'One of the worst movies of all time. I cannot believe I wasted two hours of my life for this movie']
test_reviews_with_len = [[tokenize_reviews(test_reviews[0]),
                          1,
                          len(tokenize_reviews(test_reviews[0]))],
                         [tokenize_reviews(test_reviews[1]),
                          0,
                          len(tokenize_reviews(test_reviews[1]))]]
test_sorted_reviews_labels = [(review_lab[0], review_lab[1]) for review_lab in tqdm.tqdm(test_reviews_with_len)]

predict_input = tf.data.Dataset.from_generator(lambda: test_sorted_reviews_labels, output_types=(tf.int32, tf.int32))
BATCH_SIZE = 2
test_batched_dataset = predict_input.padded_batch(BATCH_SIZE, padded_shapes=((None, ), ()))

# Get model prediction
text_model.predict(test_batched_dataset)

00%|██████████| 2/2 [00:00<00:00, 4662.93it/s]
/1 [==============================] - 0s 270ms/step

array([[0.9996767 ],
       [0.00493998]], dtype=float32)
```

Figure 7: Output

8. Discussions

The implementation of the sentiment analysis model marks a significant stride in the realm of natural language processing, leveraging state-of-the-art techniques to discern user sentiments from product reviews. As we delve into the discussion, several key aspects emerge, emphasizing the model's strengths, potential limitations, and avenues for future enhancements.

The model showcases commendable accuracy in categorizing sentiments, a testament to the power of BERT-based architectures and sequential analysis with RNN-LSTM. By considering the contextual dependencies between words and the sequential nature of language, the model excels in capturing subtle nuances, providing a more nuanced understanding of sentiment in user reviews. The meticulous data preprocessing contributes to the model's robustness, allowing it to handle diverse linguistic patterns and variations in user-generated content.

Despite the model's success, certain challenges merit consideration. The efficiency of BERT-based models comes at the cost of computational intensity, and optimizations are crucial for practical deployment. The trade-off between model complexity and real-time application efficiency requires careful consideration. Additionally, the model's performance can be sensitive to the quality and representativeness of the training data. Ensuring a diverse and balanced dataset is paramount to avoid biases and enhance generalization across various domains.

Looking ahead, there are several avenues for future research and development. Fine-tuning the model for domain-specific sentiment analysis could unlock its potential in specialized industries. Exploring techniques to address computational intensity without compromising accuracy is another promising direction. Furthermore, the model's adaptability to multilingual sentiments and its potential integration into real-time systems open doors for broader applications, from social media monitoring to personalized user experiences in e-commerce platforms.

The sentiment analysis model presented in this project is a robust tool for extracting valuable insights from user-generated content. Its accuracy, adaptability, and potential for future enhancements make it a valuable asset for businesses seeking to understand and respond to customer sentiments. The continuous evolution of natural language processing techniques and insights from practical applications ensures that sentiment analysis remains a dynamic and evolving field with vast possibilities for innovation and impact.

9. Conclusion

In conclusion, the implemented sentiment analysis model, amalgamating the power of BERT-based contextual embeddings and sequential analysis with RNN-LSTM, represents a significant stride in natural language processing. The model exhibits commendable accuracy and robustness in discerning sentiments from diverse user-generated product reviews. The meticulous preprocessing of data and the model's ability to capture contextual dependencies and sequential patterns contribute to its effectiveness across a broad spectrum of linguistic nuances. While challenges such as computational intensity and data biases warrant careful consideration, the model's potential for domain-specific fine-tuning, multilingual sentiment analysis, and integration into real-time systems opens avenues for future research and development. The discussed project offers a practical solution for businesses to understand customer sentiments and contributes to the broader landscape of sentiment analysis, emphasizing the dynamic nature of natural language processing. As we navigate through an era where user-generated content plays a pivotal role in shaping decision-making processes, the insights derived from this model hold substantial promise for industries ranging from e-commerce to social media monitoring, showcasing sentiment analysis's enduring relevance and potential impact in our evolving digital landscape.

9.1. Future Enhancement

One avenue for future enhancement lies in optimizing the model's efficiency and scalability. While the current implementation showcases robust sentiment analysis capabilities, further research could focus on streamlining the computational intensity associated with BERT-based models. Techniques such as model pruning, quantization, or exploring lightweight architectures tailored for sentiment analysis may balance accuracy and computational efficiency, facilitating real-time applications and deployment on resource-constrained devices. Scalability considerations could involve exploring distributed training strategies to handle larger datasets and accommodate the model's integration into high-traffic platforms. Future enhancements could involve domain-specific fine-tuning to extend the model's applicability across diverse industries. Tailoring the sentiment analysis model to understand and interpret sentiments within specific domains, such as healthcare, finance, or technology, would enhance its relevance and accuracy in specialized contexts. This approach requires curating domain-specific datasets and adapting the model's training regime to capture nuances unique to each industry. The result would be a more versatile and adaptable sentiment analysis tool capable of providing nuanced insights tailored to the specific needs of different sectors.

Expanding the model's capabilities to encompass multilingual sentiment analysis represents another promising avenue for future development. With the global nature of online content, the ability to accurately analyze sentiments expressed in various languages is crucial. Integrating multilingual embeddings or exploring language-agnostic approaches could enable the model to decipher sentiments in user-generated content across diverse linguistic landscapes. This enhancement would open doors for businesses with international reach, allowing them to gain comprehensive insights into customer sentiments across language barriers. Ensuring the model's sustained relevance and effectiveness requires a commitment to continuous evaluation and iterative improvements. Regularly updating the model with new datasets, staying abreast of evolving linguistic trends, and incorporating user feedback contribute to its adaptability and resilience. Implementing an agile development cycle that embraces advancements in natural language processing research ensures that the sentiment analysis model remains at the forefront of innovation and can address emerging challenges in the dynamic landscape of online communication.

Acknowledgement: The support of all my co-authors is highly appreciated.

Data Availability Statement: This study uses online benchmark data to conduct the research. This is a fresh study done by the authors.

Funding Statement: No funding has been obtained to help prepare this manuscript and research work.

Conflicts of Interest Statement: No conflicts of interest have been declared by the author(s). This is the authors' fresh work. Citations and references are mentioned as per the used information.

Ethics and Consent Statement: The consent has been obtained from the colleges during data collection and has received ethical approval and participant consent.

References

1. K. K. Mohbey, "Sentiment analysis for product rating using a deep learning approach," in 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), India, 2021, pp. 121-126, 2021.
2. E. Biswas, M. E. Karabulut, L. Pollock, and K. Vijay-Shanker, "Achieving reliable sentiment analysis in the software engineering domain using BERT," in 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME), Adelaide, SA, Australia, pp. 162-173, 2020, doi: 10.1109/ICSME46990.2020.00025.
3. S. Alaparthi and M. Mishra, "Bidirectional Encoder Representations from Transformers (BERT): A sentiment analysis odyssey," arXiv [cs.CL], 2020.
4. L. Yang, Y. Li, J. Wang, and R. S. Sherratt, "Sentiment analysis for E-commerce product reviews in Chinese based on sentiment lexicon and deep learning," IEEE Access, vol. 8, pp. 23522–23530, 2020.
5. A. Onan, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks"," Concurrency Computat Pract Exper, vol. 33, 2021.
6. S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey, "A comparative study of support vector machine and naive Bayes classifier for sentiment analysis on Amazon product reviews," in 2020 International Conference on Contemporary Computing and Applications (IC3A), India, 2020, pp. 217-220, 2020.
7. N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment analysis based on deep learning: A comparative study," Electronics (Basel), vol. 9, no. 3, p. 483, 2020.
8. L. Kurniasari and A. Setyanto, "Sentiment Analysis using Recurrent Neural Network," J. Phys. Conf. Ser., vol. 1471, no. 1, p. 012018, 2020.
9. H. Fouadi, H. El Moubtahij, H. Lamtougui, and A. Yahyaouy, "Sentiment analysis of Arabic comments using machine learning and deep learning models," Indian J. Comput. Sci. Eng., vol. 13, no. 3, pp. 598–606, 2022.
10. M. Munikar, S. Shakya, and A. Shrestha, "Fine-grained sentiment classification using BERT," in 2019 Artificial Intelligence for Transforming Business and Society (AITB), Kathmandu, Nepal, pp. 1-5, 2019, doi: 10.1109/AITB48515.2019.8947435.
11. A. B. Naeem, B. Senapati, M. S. Islam Sudman, K. Bashir, and A. E. M. Ahmed, "Intelligent road management system for autonomous, non-autonomous, and VIP vehicles," World Electric Veh. J, vol. 14, no. 2, 2023.
12. A. Jain, K. K. Ramachandran, S. Sharma, T. Sharma, P. Pareek, and B. Pant, "Detailed investigation of influence of machine learning (ML) and big data on digital transformation in marketing," in 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 1061-1065, 2022, doi: 10.1109/ICACITE53722.2022.9823810.
13. A. K. Sharma, A. Panwar, P. Chakrabarti, and S. Viswakarma, "Categorization of ICMR Using Feature Extraction Strategy and MIR with Ensemble Learning"," Procedia Computer Science, vol. 57, pp. 686–694, 2015.

14. A. K. Sinha, A. Shankar Hati, M. Benbouzid, and P. Chakrabarti, “ANN-based Pattern Recognition for Induction Motor Broken Rotor Bar Monitoring under Supply Frequency Regulation”,” Machines, vol. 9, 2021.
15. A. M. Soomro, “Constructor development: Predicting object communication errors,” in 2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICES&T), Bahawalpur, Pakistan, 2023, pp. 1-7, doi: 10.1109/ICEST56843.2023.10138846.
16. A. Magare, M. Lamin, and P. Chakrabarti, “Inherent Mapping Analysis of Agile Development Methodology through Design Thinking”,” Lecture Notes on Data Engineering and Communications Engineering, vol. 52, pp. 527–534, 2020.
17. A. Prasad and P. Chakrabarti, “Extending Access Management to maintain audit logs in cloud computing,” Int. J. Adv. Comput. Sci. Appl., vol. 5, no. 3, 2014.
18. B. Nagarjuna, K. K. Ramachandran, A. Nautiyal, S. P. Singh, B. B. Nayak and P. Ganguly, "Sustainability in the field of Supply Chain Using Technolgy: A deep review," 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 843-847, 2023. doi: 10.1109/ICACITE57410.2023.10183000.
19. B. Senapati and B. S. Rawal, “Quantum communication with RLP quantum resistant cryptography in industrial manufacturing,” Cyber Security and Applications, no. 100019, p. 100019, 2023.
20. B. Senapati, J. R. Talburt, A. Bin Naeem, and V. J. R. Batthula, “Transfer learning based models for food detection using ResNet-50,” in 2023 IEEE International Conference on Electro Information Technology (eIT), Romeoville, IL, USA, pp. 224-229, 2023, doi: 10.1109/eIT57321.2023.10187288.
21. B. T. Hung and P. Chakrabarti, “Parking lot occupancy detection using hybrid deep learning CNN-LSTM approach,” in Algorithms for Intelligent Systems, Singapore: Springer Nature Singapore, 2022, pp. 501–509.
22. D. Uike, S. Agarwalla, V. Bansal, M. K. Chakravarthi, R. Singh and P. Singh, "Investigating the Role of Block Chain to Secure Identity in IoT for Industrial Automation," 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 2022, pp. 837-841, 2022, doi: 10.1109/SMART55829.2022.10047385.
23. G. Saravana Kumar, K. K. Ramachandran, S. Sharma, R. Ramesh, K. Qureshi, and K. V. B. Ganesh, “AI-assisted resource allocation for improved business efficiency and profitability,” in 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 54-58, 2023, doi: 10.1109/ICACITE57410.2023.10182679.
24. H. Patidar, P. Chakrabarti, and A. Ghosh, “Parallel Computing Aspects in Improved Edge Cover based Graph Coloring Algorithm”,” Indian Journal of Science and Technology, vol. 10, no. 25, pp. 1–9, 2017.
25. K. K. Ramachandran, A. Perez-Mendoza, K. Joshi, S. M. Parikh, D. Saini, and J. L. A. Gonzáles, “Deep learning-based topic-level examination of social media,” in 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, pp. 1684-1687, 2022, doi: 10.1109/IC3I56241.2022.10073324.
26. K. Verma, P. Srivastava, and P. Chakrabarti, “Exploring structure oriented feature tag weighting algorithm for web documents identification”,” Communications in Computer and Information Science, vol. 837, pp. 169–180, 2018.
27. M. Sabugaa, B. Senapati, Y. Kupriyanov, Y. Danilova, S. Irgasheva, and E. Potekhina, “Evaluation of the prognostic significance and accuracy of screening tests for alcohol dependence based on the results of building a multilayer perceptron,” in Artificial Intelligence Application in Networks and Systems, Cham: Springer International Publishing, 2023, pp. 240–245.
28. M. Tiwari, P. Chakrabarti, and T. Chakrabarti, “Novel work of diagnosis in liver cancer using Tree classifier on liver cancer dataset (BUPA liver disorder)” Communications in Computer and Information Science, vol. 837, pp. 155–160, 2018.
29. N. C. Sattaru, D. Umrao, K. K. Ramachandran, K. K. Karthick, M. Tiwari, and S. Kumar, “Machine learning as a predictive technology and its impact on digital pricing and cryptocurrency markets,” in 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 1077-1080,2022, doi: 10.1109/ICACITE53722.2022.9823734.
30. N. Priyadarshi, A. K. Bhoi, S. K. Sahana, P. K. Mallick, and P. Chakrabarti, “Performance enhancement using novel soft computing AFLC approach for PV power system”,” Advances in Intelligent Systems and Computing, vol. 1040, pp. 439–448, 2020.
31. P. Chakrabarti and P. S. Goswami, “Approach towards realizing resource mining and secured information transfer” International Journal of Computer Science and Network Security, vol. 8, no. 7, pp. 345–350, 2008.
32. P. Chakrabarti, A. Choudhury, N. Naik, and C. T. Bhunia, “Key generation in the light of mining and fuzzy rule” International Journal of Computer Science and Network Security, vol. 8, no. 9, pp. 332–337, 2008.
33. P. Kumar, A. S. Hati, S. Padmanaban, Z. Leonowicz, and P. Chakrabarti, “Amalgamation of transfer learning and deep convolutional neural network for multiple fault detection in SCIM,” in 2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe), Madrid, Spain, 2020, pp. 1-6, doi: 10.1109/EEEIC/ICPSEurope49358.2020.9160712.

34. Prince, A. S. Hati, P. Chakrabarti, J. H. Abawajy, and N. W. Keong, “Development of energy efficient drive for ventilation system using recurrent neural network,” Neural Comput. Appl., vol. 33, no. 14, pp. 8659–8668, 2021.
35. R. R. Chandan, M. Lourens, K. K. Ramachandran, S. V. Akram, R. Bansal, and D. Kapila, “Implementation and execution of blockchain technology in the field of education,” in 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, pp. 1207-1210, 2023, doi: 10.1109/ICACITE57410.2023.10183165.
36. S. K. Umamaheswaran, D. Uike, K. K. Ramachandran, T. Suba, and T. Verma, The Critical Understanding on the Emerging Threats and Defensive Aspects in Cryptocurrencies using Machine Learning Techniques. 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). Greater Noida, India, pp. 1938-1942, 2022, doi: 10.1109/ICACITE53722.2022.9823440.
37. S. Rajeyyagari, B. T. Hung, and P. Chakrabarti, “Applications of artificial intelligence in biomedical image processing,” in 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, pp. 53-57, 2022, doi: 10.1109/ICAIS53314.2022.9742773.
38. S. Sonnad, M. Sathe, D. K. Basha, V. Bansal, R. Singh and D. P. Singh, "The Integration of Connectivity and System Integrity Approaches using Internet of Things (IoT) for Enhancing Network Security," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, pp. 362-366,2022, doi: 10.1109/IC3I56241.2022.10072475.
39. V. Bansal, S. Pandey, S. K. Shukla, D. Singh, S. A. Rathod and J. L. A. Gonzáles, "A Frame Work of Security Attacks, Issues Classifications and Configuration Strategy for IoT Networks for the Successful Implementation," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, pp. 1336-1339, 2022, doi: 10.1109/IC3I56241.2022.10073429.
40. V. Gunturu, V. Bansal, M. Sathe, A. Kumar, A. Gehlot and B. Pant, "Wireless Communications Implementation Using Blockchain as Well as Distributed Type of IOT," 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, pp. 979-982, 2023, doi: 10.1109/AISC56616.2023.10085249.
41. V. Suthar, V. Bansal, C. S. Reddy, J. L. A. Gonzáles, D. Singh and D. P. Singh, "Machine Learning Adoption in Blockchain-Based Smart Applications," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, pp. 372-378, 2022, doi: 10.1109/IC3I56241.2022.10072980.
42. X. Wang, T. Huang, and P. Chakrabarti, “Adaptive RBF neural network control for nonlinear system,” in Complex Systems: Spanning Control and Computational Cybernetics: Foundations, Cham: Springer International Publishing, pp. 435–455, 2022.