# Sentiment Analysis using BERT to Classify Amazon Beauty Product Reviews

**Presented by**:
Christopher S. Bennett

**Course Name:**
IFSC 7325: Deep Learning Theory

**Professor**:
Dr. Xiaowei Xu

**Teaching Assistant**:
Hanuman Lakamsani

University of Arkansas Little Rock
Department of Information Science
Donaghey College of Science, Technology, Engineering, and Mathematics
Spring 2023 Semester

# Sentiment Analysis using BERT to Classify Amazon Beauty Product Reviews

## Abstract

Sentiment analysis, or opinion mining, is the computational study of people's opinions, attitudes, and emotions towards entities. It is challenging yet valuable for businesses and consumers. With the growth of social media, finding and analyzing opinionated content online is difficult due to diverse sources, large volumes of text, and human biases (Liu & Zhang, 2012). Automated opinion mining systems are needed to overcome these limitations. In the research project, a pre-trained BERT model was developed and fine-tuned for classifying customer sentiments relative to Amazon beauty product reviews as "positive" or "negative" based on their associated ratings. The model was evaluated using several metrics including Accuracy, F1-Score, Precision, Recall, AUC (ROC curve) and Cross-Entropy Loss Function. The BERT model proved to be very robust with an *overall accuracy*, a *precision, recall* and *F1-score* of 92%, 90% and 91% respectively. The *ROC-AUC* is 91% and the *Binary-cross-entropy* loss function declined steadily across all 10 training epochs.

## 1. Introduction

Sentiment analysis of e-commerce reviews is a hot topic in the e-commerce product quality management space. Manufacturers have been increasingly leveraging this technology to gain insight into public opinion about their products (Liu, Y, Yang, J. & Mao, 2020; Liu & Zhang, 2012). This type of analysis is beneficial for e-commerce companies, as it can provide valuable information regarding customer satisfaction with products, price, quality and delivery times. By understanding public sentiment, manufacturers can make changes to their products or services to ensure they are meeting customer expectations and staying competitive in the market. With sentiment analysis, companies can also identify areas of opportunity to potentially increase sales, or identify areas of improvement to increase customer satisfaction. Ultimately, sentiment analysis of e-commerce reviews is a powerful tool that can help manufacturers gain valuable insight into the public opinion of their products.

The objective of this paper is to determine how well a BERT (Bidirectional Encoder Representations from Transformers) model classifies Amazon beauty product reviews as positive or negative based consumer ratings of the same products.

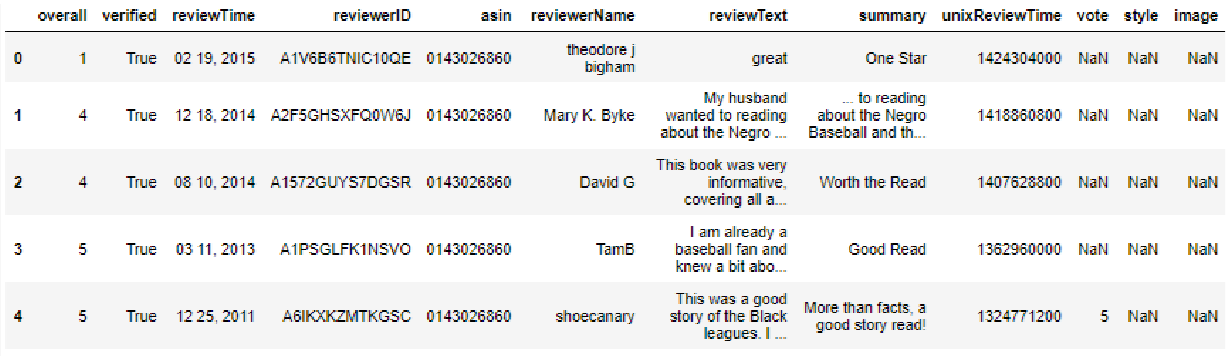
## 2. Related Work

There exists a fairly significant body of research at the intersection of product reviews and sentiment analysis or opinion mining. Tan, Wang and Xu (2023) investigated the relationship between Amazon electronics product reviews and the ratings consumers gave the associated products, employing both traditional machine learning techniques, including Naive Bayes analysis, Support Vector Machines, K-nearest neighbor method, and deep-learning frameworks such as Recurrent Neural Network (RNN). The researchers discovered that the RNN (LSTM) model demonstrated higher accuracy compared to traditional machine learning algorithms. Gope, Tabassum, Mabrur, Yu and Arifuzzaman (2022) conducted a similar analysis on the same Amazon electronics product dataset using a slightly different mix of traditional machine learning and deep learning techniques, including Linear Support Vector Machine, Random Forest, Multinomial Naive Bayes, Bernoulli Naive Bayes, Logistic Regression and RNN with LSTM; and just as in the previous case, the RNN-LSTM emerged as the most performant with an accuracy of 97.52%. AlQahtani (2021) conducted both multiclass classification and binary classification of sentiments around on Amazon mobile phone reviews utilizing various traditional machine learning and deep learning techniques, including Logistic Regression, Random Forest, Naive Bayes, Bidirectional Long-Short Term Memory (LSTM) and BERT. Notably, the BERT models achieved superior performances in both the multiclass classification and binary classification, with accuracy scores of 94% and 98% respectively. Selvakumar and Lakshmanan (2022) developed a BERT model for classifying customer sentiments on IMDB and Amazon Fine Food reviews. The BERT model was shown to have outperformed traditional machine learning and deep learning models that were developed on the same data.

In this paper, a pre-trained BERT model will be deployed and fine-tuned for sentiment analysis on the customer reviews relative to an Amazon beauty products dataset.

## 3. Data

The project dataset is composed of consumer reviews of beauty products sold on Amazon and is curated by Ni (2023); also it has appeared in the scholarly research of Ni & McAuley (2019). The dataset contains 371,345 records and 12 attributes (columns), including text reviews and consumer ratings of the products ranging from a low of 1 to a high of 5, which are given in the “Overall” column. The name of the dataset is All_Beauty.json, which is in the json format and is read into a Pandas dataframe using the Python library named read_json. **Table1** below shows the first five records (head) of the dataframe.



| | overall | verified | reviewTime | reviewerID | asin | reviewerName | reviewText | summary | unixReviewTime | vote | style | image |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | True | 02 19, 2015 | A1V6B6TNIC10QE | 0143026860 | theodore j bigham | great | One Star | 1424304000 | NaN | NaN | NaN |
| **1** | 4 | True | 12 18, 2014 | A2F5GHSXFQ0W6J | 0143026860 | Mary K. Byke | My husband wanted to reading about the Negro ... | ... to reading about the Negro Baseball and th... | 1418860800 | NaN | NaN | NaN |
| **2** | 4 | True | 08 10, 2014 | A1572GUYS7DGSR | 0143026860 | David G | This book was very informative, covering all a... | Worth the Read | 1407628800 | NaN | NaN | NaN |
| **3** | 5 | True | 03 11, 2013 | A1PSGLFK1NSVO | 0143026860 | TamB | I am already a baseball fan and knew a bit abo... | Good Read | 1362960000 | NaN | NaN | NaN |
| **4** | 5 | True | 12 25, 2011 | A6IKXKZMTKGSC | 0143026860 | shoecanary | This was a good story of the Black leagues. I ... | More than facts, a good story read! | 1324771200 | 5 | NaN | NaN |

Table 1. First five records of the All_Beauty.json dataframe

## 3.1. Data Preprocessing

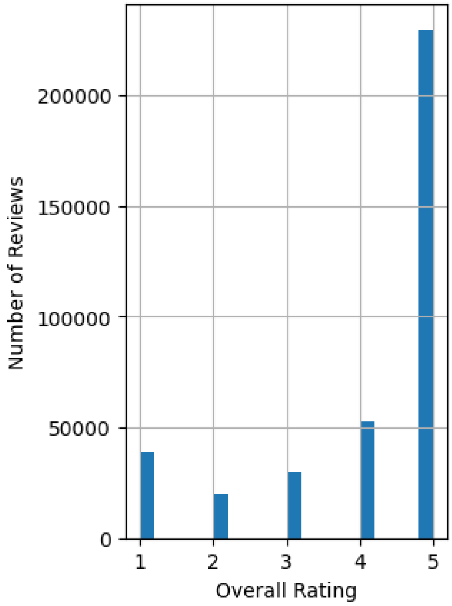


Fig. 1. Distribution of records by review rating

The first step in the data preprocessing phase involves performing exploratory data analysis around the ratings attribute in order to assess the distribution of the records across customer ratings. As is evidenced in **Fig. 1** above, customer reviews associated with 5-star ratings dominate the dataset, followed at some distance by reviews with a 4-star rating.

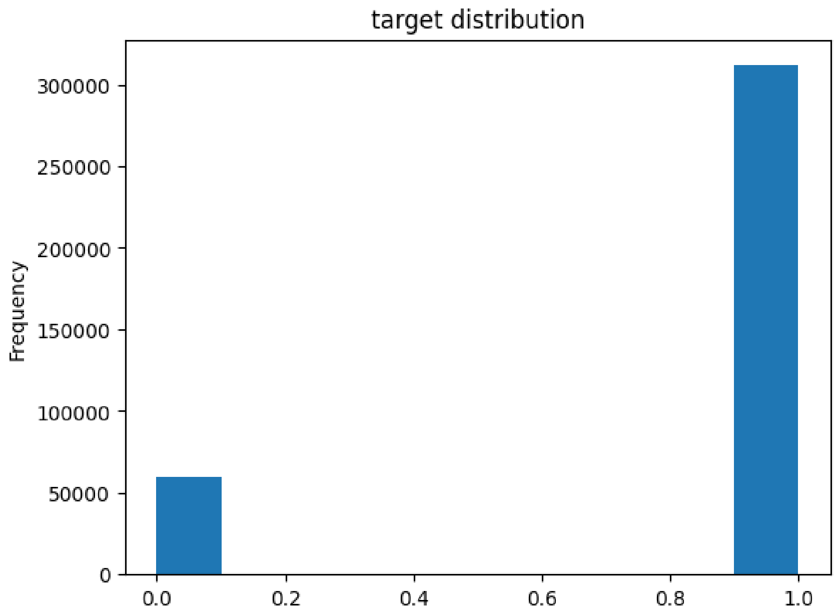


Fig. 2. Distribution of the number of records by sentiment before resampling

Next, consumer review scores less than or equal to 2 are labeled as “negative”; whereas scores greater than or equal to 3 are assigned a label of “positive.” Then the “positive” and “negative” values are re-coded as 1 and 0 respectively and those values are assigned to an attribute designated “target,” which represents the outcome or dependent variable for sentiment classification task. Also, the dataframe was reduced to just two attributes or columns called “text” and “target” with the former containing the consumer textual reviews and the latter containing the consumer sentiments represented by 1’s and 0’s.

## 3.1. Data Resampling

**Fig. 2** depicts the distribution of the records in the dataset across the values of the “target” variable; there are significantly more records or cases having a value of 1 (“positive sentiments”) than those exist those having a value of 0 (“negative sentiments”). Therefore, the dataset is imbalanced in favor of records with “positive” reviews or sentiments. The data imbalance is remedied by randomly oversampling the minority class (“negative”) and undersampling the majority class (“positive”) such that there are an equal number of records with “positive” sentiments as those with “positive” sentiments.

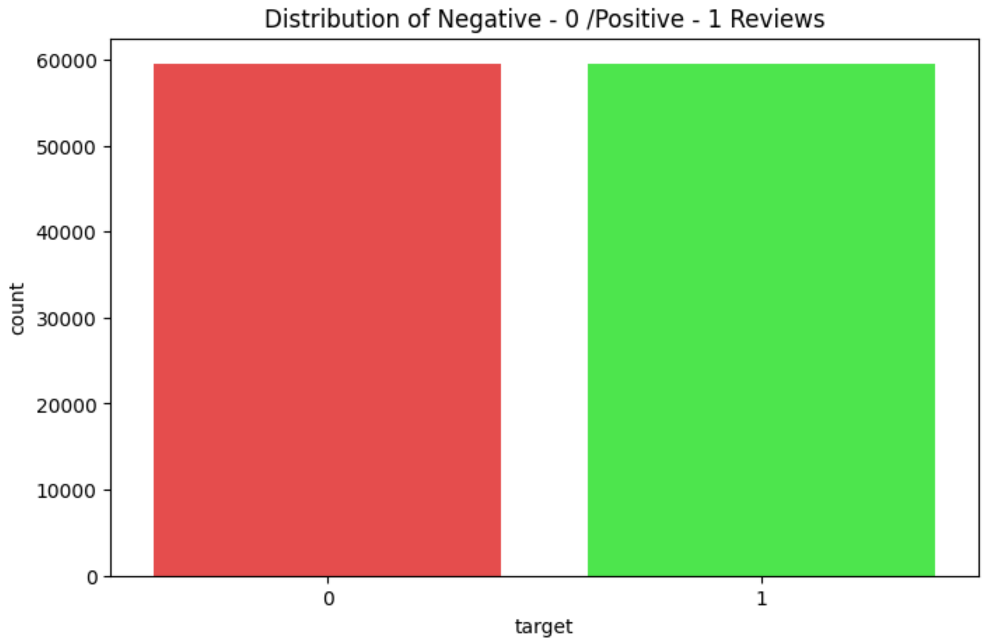


Fig. 3. Distribution of the records by sentiment after resampling

**Fig. 3** above illustrates the distribution of records between the two classes after performing the rebalancing procedure; notice that the red bar is the same height as the green bar, indicating that the classes are now balanced. The new dataframe now contains 118,987 records (reviews); 59,510 "negative" reviews (0's) and 59,477 "positive" reviews (1's).

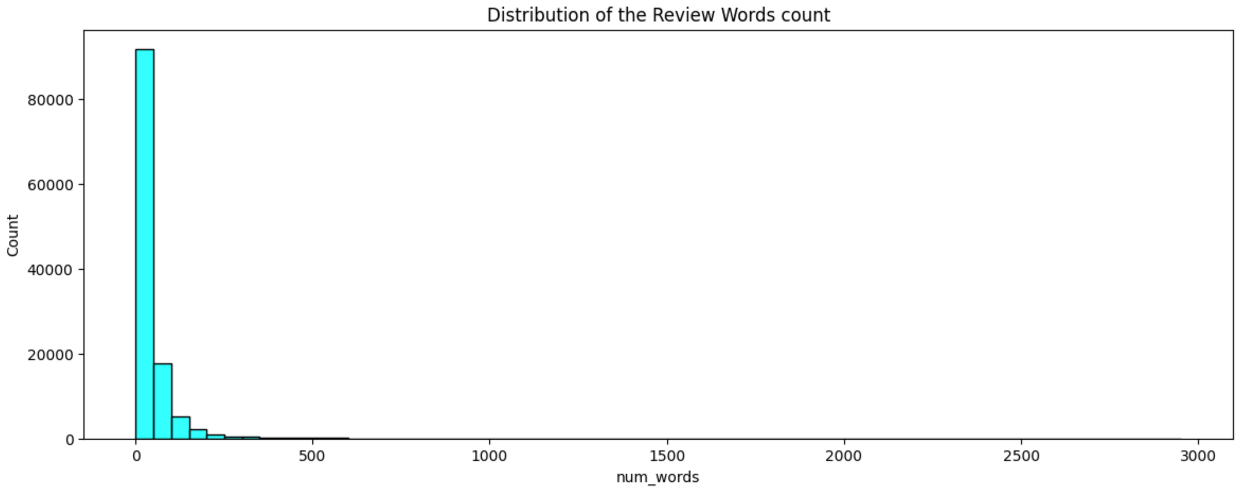
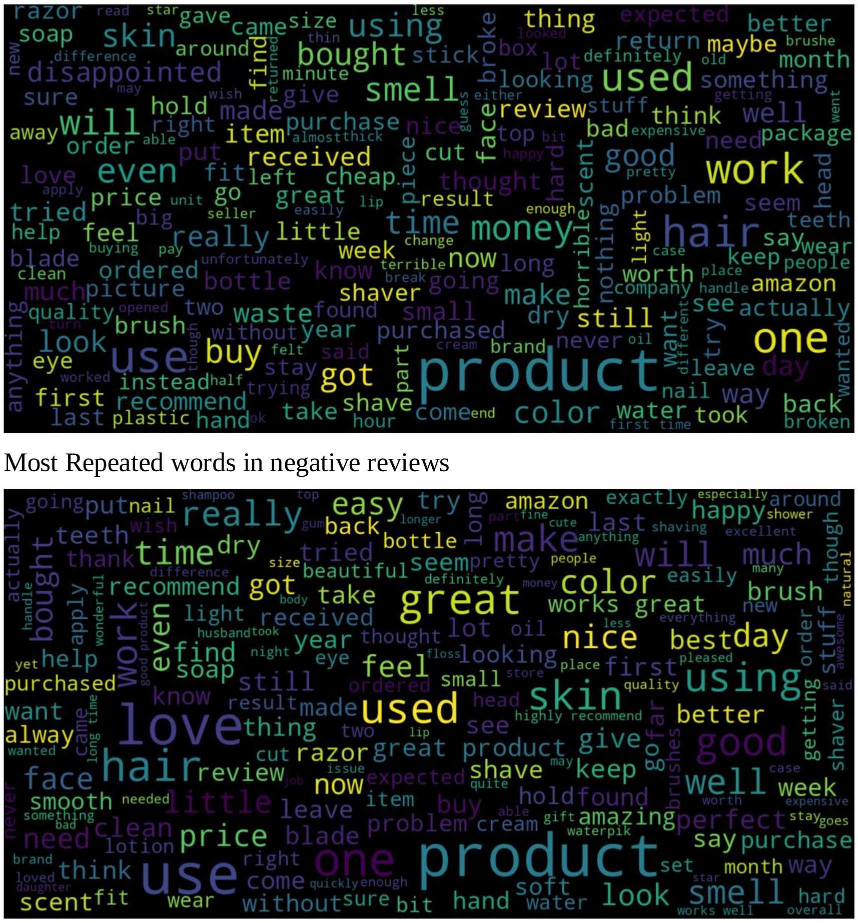


Fig. 4. Distribution of the number of words per record

The chart in **Fig. 4** above depicts the distribution of the word count per review. Notably, the majority of the reviews have a similar length or word count.



Most Repeated words in negative reviews

Most Repeated words in positive reviews

Fig. 5. Word cloud maps of positive and negative commonly used words

The above charts in **Fig. 5** represent “word clouds” that were generated using the Python *wordcloud* library. The first “word cloud” reflects the most frequently used words associated with “negative” customer sentiment; whereas the second illustrates the most popular words relative to “positive” customer sentiment.

## 4. Methodology

The BERT modeling strategy for the proposed project will mirror the previous work of Mishinev (2023), which entailed building a classification model to isolate “real” news from “fake” news. However, the proposed BERT model will classify consumer beauty product reviews as “positive” or “negative,” based on consumer product ratings between 1 and 5 stars. As noted earlier, review scores less than or equal to 2 were labeled as “negative”; whereas scores greater than or equal to 3 were assigned a label of “positive.” The model will be implemented in Python leveraging the various packages including keras, tensorflow and transformers. The model performance will be assessed using several metrics including *overall classification accuracy*, *precision*, *recall*, the *F1-score* and the *ROC-AUC* metrics.

## 4.1. Deep-learning Model : BERT

According to Devlin, Chang, Lee and Toutanova (2018), the BERT (Bidirectional Encoder Representations from Transformers) model, is a neural network model for transfer learning. Liu, Lu, Yang, and Mao (2020) note that, unlike traditional models such as RNN (Recurrent Neural Network) and LSTM (Long Short-Term Memory) that read text sequentially in one direction, BERT applies bidirectional training using transformers to learn contextual relations between words and tokens (Tok1,...,TokM) in text; the transformer consists of an encoder (E1,...,EN) that reads the text input and a decoder that makes predictions for the task. BERT has gained widespread attention in the machine learning community due to its state-of-the-art results in various NLP (natural language processing) tasks and beyond. Further, they maintain, one of key features of BERT is that it alleviates the limitations of bidirectional constraints by using a "masked language model" (MLM) for pre-training natural language processing; that in MLM, some of the tokens from the input are randomly masked, and the objective is to predict the original vocabulary of the masked word based on its context (Liu, Lu, Yang & Mao, 2020). This bidirectional training allows BERT to have a deeper understanding of language context and learn the contextual relationships of a word with all its surrounding words (Liu, Lu, Yang & Mao, 2020).

## 4.2. Defining the BERT Model

This research project employs the BERT language model for text classification. To preprocess the text data, the maximum length of input sequence is set to 100. The AutoTokenizer module from the Hugging Face transformers library is used to tokenize the text data, which is then encoded with special tokens, truncated to fit the maximum length, and padded with zeros. The neural network architecture for the text classification task is defined with a series of layers. The model takes two inputs, *input_ids* and *input_mask*, which are passed through the pre-trained BERT model to obtain the embeddings for the input text. A dropout layer is then applied to prevent overfitting, followed by a dense layer with 64 units and a ReLU activation function. Another dropout layer is applied before the final dense layer with a sigmoid activation function that outputs the probability of the input text belonging to a particular class.

Next, the model is compiled with the Adam optimizer and binary_crossentropy loss function, with the learning rate set to 1e-05, epsilon to 1e-08, decay to 0.01, and clipnorm to 1.0. To visualize the architecture of the neural network, the tf.keras.utils.plot_model() function is utilized. **Fig.6** below provides a visual illustration of the BERT model architecture.

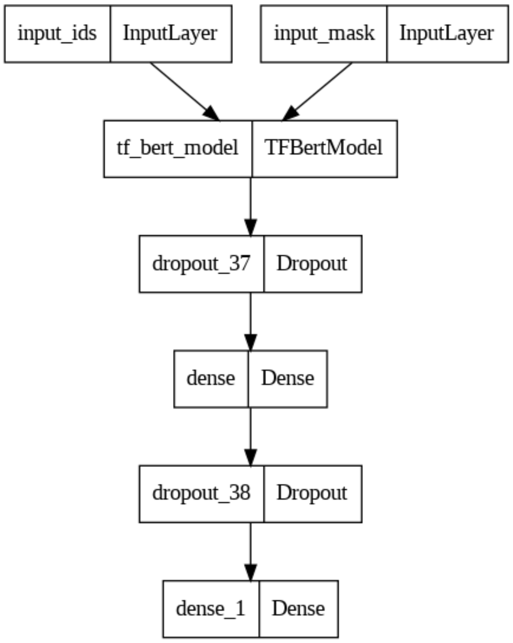


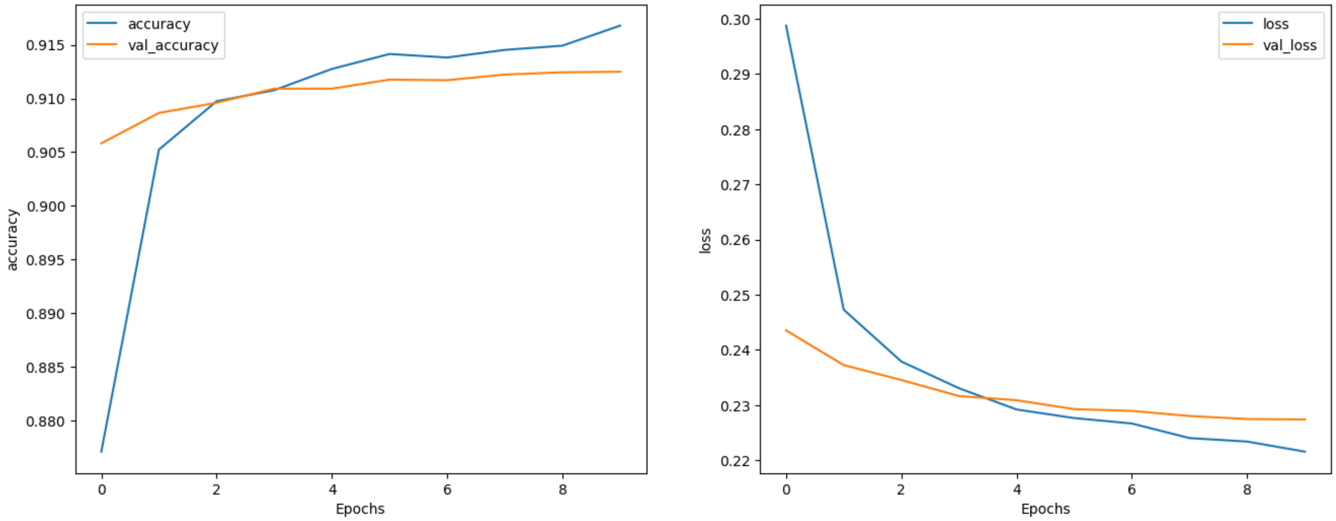
Fig. 6. Architecture of the BERT model

## 5. Results

Before training the BERT model, the dataset, containing 118,987 records (reviews) and 2 variables, i.e. the textual reviews and the consumer sentiments, is partitioned into a training dataset and testing dataset, in an 80:20 percentage ratio; 80% of the data is used for training the model and the remaining 20% is used for validation. The data is split using a stratified sampling technique to ensure that the distribution of the target variable is maintained in both the training and testing sets. Also, a random state parameter is employed to ensure reproducibility of the results. Next, a custom function called get_tokens() function is used to preprocess the text data. This function converts the input text into sequences of token IDs and creates attention masks, which are required for the model input.

Finally, the *fit()* method is used to train the deep learning model. The *x* parameter is set to a dictionary of input data with two keys, *input_ids* and *input_mask*, corresponding to the token IDs and attention masks of the training set. The **y** parameter is set to the target variable *y_train*. The *epochs* parameter is set to 10, indicating that the model will be trained for 10 epochs. The *validation_split* parameter is set to 0.2, indicating that 20% of the training data will be used for validation during training. The *batch_size* parameter is set to 32, indicating that each training batch will contain 32 samples. The *callbacks* parameter is set to a list containing an instance of the *EarlyStopping* callback, which monitors the validation accuracy and stops the training process if it does not improve for 3 consecutive epochs. The *restore_best_weights* parameter is set to *True*, indicating that the weights of the best-performing model on the validation set will be restored at the end of the training process.

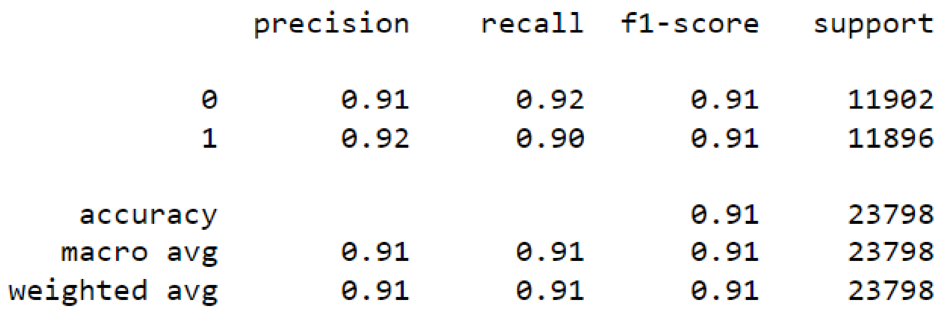
## 5.1. Model Evaluation



(a) (b)

Fig. 7. Comparing the accuracy and loss metrics for training and validation datasets

The two charts (labeled *a* and *b*) in **Fig. 7** above illustrate the progression of the model *accuracy* and *loss* respectively for both the training and validation datasets. Both charts suggest that the model is very robust, with both the model *accuracy* and *binary cross entropy loss* function for both the training and validation data increasing and decreasing respectively over the 10 modeling epochs. Notably, the metrics for the validation data tracks those for the training data fairly closely, suggesting that the model fits the data well.



```
              precision    recall  f1-score   support

           0       0.91      0.92      0.91     11902
           1       0.92      0.90      0.91     11896

    accuracy                           0.91     23798
   macro avg       0.91      0.91      0.91     23798
weighted avg       0.91      0.91      0.91     23798
```

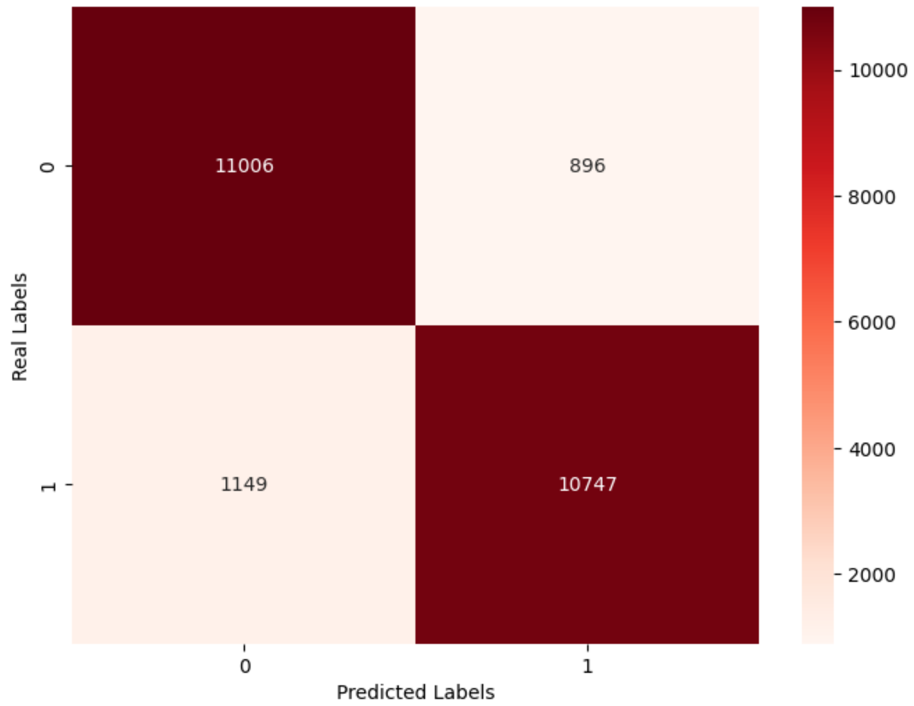


Fig. 8. Classification Report

Also, the model's performance on the validation data may be appraised based on its overall *classification accuracy*, *precision*, *recall*, the *F1-score* and the *ROC-AUC* metrics. An assessment of the model's performance on the validation dataset is necessary to avoid overfitting. Overfitting refers to a phenomenon where the model does well classifying records in the training dataset but does poorly on the validation data, i.e. data the model has not seen before- not used to train the model.

Within the context of classification modeling, *accuracy* indicates the total number of observations from both classes that were classified correctly; please note, however, that accuracy is not a good measure for unbalanced data as it does not reflect the performance of each class, particularly that of the class we are interested in, i.e. the minority class. However, that is not a concern in this case as we made sure to balance the data before training the model. The *recall (sensitivity)* measure indicates the total number of observations from the minority or “positive” class that was correctly classified or identified by the model. The *precision* metric, on the other hand, denotes the total number of observations that were identified as belonging to the minority class that were actually “positive” or of the minority class. The *F1-score* represents the weighted harmonic mean of the *precision* and *recall* scores; maximizing the *F1-score* ensures the optimization of the value of the *precision* measure relative to that of the *recall*. Finally, the *ROC-AUC* measure is the most used to evaluate the performance of classification algorithms.

The *accuracy*, *precision*, *recall*, and *ROC-AUC* scores vary between *0* and *1*, with 1 being the highest; the closer the value of these measures is to1, the better is the performance of the classifier. In addition, a *ROC-AUC* score of *0.5* is equivalent to the performance of a naive model based on random guessing; therefore, a classifier whose *ROC-AUC* is greater than 0.5 is deemed to be more robust than merely guessing randomly. The BERT model estimated in this study is very robust when judged by the aforementioned metrics relative to the minority class (“positive sentiment”): the *precision*, *recall* and *F1-score* are 0.92, *0.9* and *0.91* respectively (see **Fig. 8** above). Also, as reflected in **Fig. 9** below, the *ROC-AUC* is *0.91*.

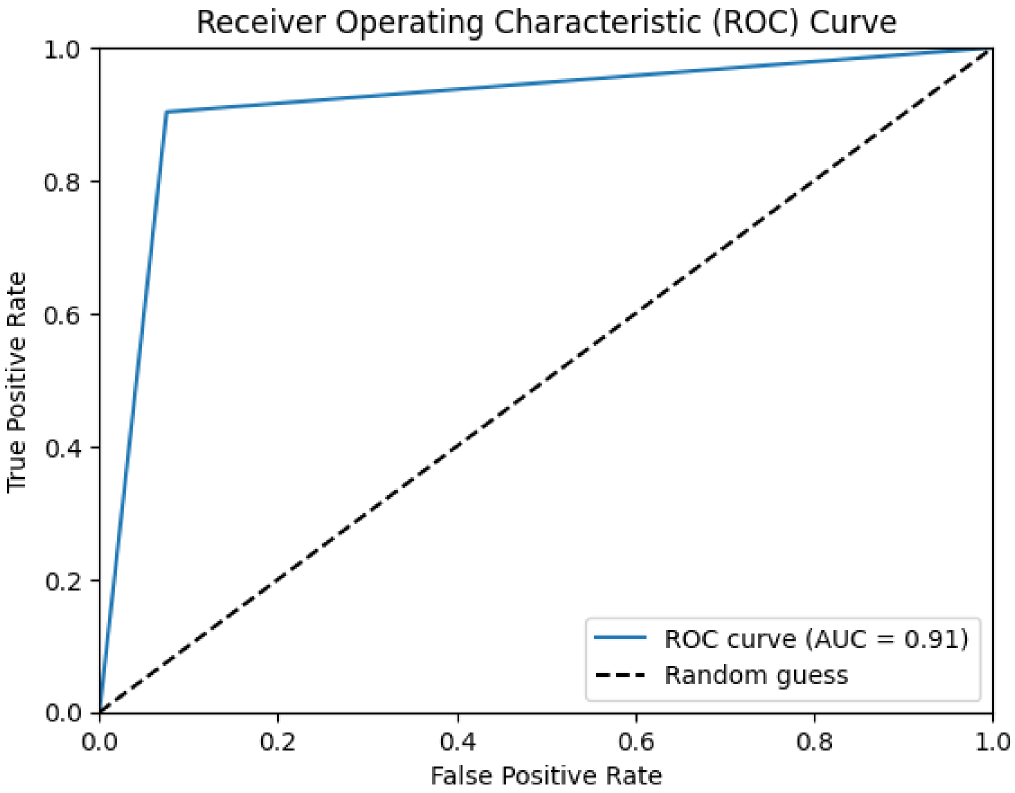


Fig. 9. ROC-AUC Curve

All metrics have values very close to 1, indicating that the BERT model fits the data very well.

# 6. Conclusion

## 6.1. Discussion & Limitations

Consumer online product reviews provide valuable feedback to consumer goods manufacturers regarding the products sold on e-commerce platforms, allowing them to enhance the quality of their products. The aim of this research project was to determine how well a BERT model classifies Amazon beauty product reviews as “positive” or “negative” based consumer ratings. The results of this study indicate that the BERT model is an effective framework for conducting sentiment analysis. Notably, the model demonstrated strong performance on the validation data with overall accuracy, precision, recall, F1-score and ROC-AUC scores for the “positive” class of 0.92, *0.90,0.91* and *0.91* respectively.

However, this study is not without limitations. The evaluative content available on the internet, such as product reviews, forum posts, and blogs, has emerged as a valuable source of opinions on various subjects including products, services, events, individuals, and more. However, according to Jindal and Liu (2008) argues that, while researchers have extensively studied these opinion sources using natural language processing and data mining techniques for tasks such as classification and summarization of opinions, an important aspect that has been overlooked thus far is the presence of opinion spam or the trustworthiness of online opinions. Further, based on the analysis of 5.8 million Amazon.com reviews and 2.14 million reviewers, they found widespread opinion spam (Jindal & Liu, 2008).

Also, for the purpose of this study, it is assumed that the sentiment in the reviews corresponds to the number of stars given. However, it is possible that some reviews may not align with this assumption, where a very positive review may be given a low star rating, or vice versa, which could potentially introduce noise into the dataset. Additionally, another important step in machine learning is cross-validation, which helps eliminate the risk of accidentally selecting an "easy-to-classify" subset of data as the test set, which could introduce bias into the results. However, in this project, cross-validation was not implemented for the training and testing phases.

## 6.2. Future Work

In this research project, review ratings with 3 stars and above were coded as a "positive" whereas reviews with 2 stars and below were treated as "negative." However, a 3-star rating is generally treated as "neutral." Therefore, for future research, I would recommend that this project be repeated with the 3-star ("neutral") reviews being excluded from the analysis with a view to determine whether doing so would have a material impact on the model results. Also, this paper focused only on the BERT model, and while the results are spectacular, they do not tell us how well the same model would perform relative to other traditional deep-learning techniques. Vaswani et al (2017) argue that the network architecture, transformer, can not only handle long sequences without the use of recurrent or convolutional layers but also demonstrated that the self-attention mechanisms can outperform traditional neural network layers. I also recommend that the foregoing claim be tested in future research using the data employed in this project.

## 7. References

AlQahtani, A. A. (2021). Product Sentiment Analysis for Amazon Reviews. *International Journal of Computer Science and Information Technology*, *13*(3), 15–30. https://doi.org/10.5121/ijcsit.2021.13302

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805

Gope, J. C., Tabassum, T., Mabrur, M., Yu, K., & Arifuzzaman, M. (2022). Sentiment Analysis of Amazon Product Reviews Using Machine Learning and Deep Learning Models. 2022 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE). https://doi.org/10.1109/icaeee54957.2022.9836420

Jindal, N., & Liu, B. (2008). Opinion spam and analysis. Web Search and Data Mining. https://doi.org/10.1145/1341531.1341560

Liu, B.  & Zhang, L. (2012). A Survey of Opinion Mining and Sentiment Analysis. In Aggarwal, C.C & Zhai, C.Z, Mining Text Data(pp.415-463), New York: NY: Springer

Liu, Y., Lu, J., Yang, J., & Mao, F. (2020). Sentiment analysis for e-commerce product reviews by deep learning model of Bert-BiGRU-Softmax. Mathematical Biosciences and Engineering, 17(6), 7819–7837. https://doi.org/10.3934/mbe.2020398

Mishinev, T. (2023).  Fake News Keras/BERT. Accessed on 3/27/2023 from https://www.kaggle.com/code/tmishinev/fake-news-keras-bert

Ni, J. (2023). Amazon Review Data (2018). Accessed on 3/27/2023 from https://nijianmo.github.io/amazon/index.html

Ni, J., Li, J., & McAuley, J. (2019). Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2(1–2), 1–135. https://doi.org/10.1561/1500000011

Selvakumar, B., & Lakshmanan, B. (2022). Sentimental analysis on user's reviews using BERT. Materials Today: Proceedings, 62, 4931–4935. https://doi.org/10.1016/j.matpr.2022.03.678

Tan, W., Wang, X, & Xu, X. (2023) Sentiment Analysis for Amazon Reviews. Accessed on 3/27/2023 from https://cs229.stanford.edu/proj2018/report/122.pdf

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).