Sentiment Analysis Using BERT on Amazon Reviews

Received: 31 January 2023; Accepted: 16 February 2023 *Research Article*

Tea Bogatinoska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
bogatinoskatea@gmail.com

Jana Trpkovska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
trpkovskajana@gmail.com

Tamara Mitrevska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
mitrevskaat@gmail.com

Georgina Mirceva
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
georgina.mirceva@finki.ukim.mk

***Abstract*—With the growth of social medias, blogs, discussion forums, online review sites, etc., major companies have come to realize that being sentiment-aware can help them gain insights into user behavior, track and manage their online presence and image and use that information to boost brand loyalties and advocacy, marketing message, product development, monitor competitive intelligence, etc. In this paper, we focus on the research task for sentiment analysis on Amazon reviews data. We used the BERT-base-cased model from Hugging Face. Some experimental results are presented and discussed in this paper.**

Keywords—deep learning, BERT, sentiment analysis, transformers

I. INTRODUCTION

Through the reviews left on the e-commerce applications, customers freely share experiences and opinions with other customers. Users tend to express a variety of sentiments in their reviews, therefore these posts provide invaluable insight into how the users think. Since humans express their thoughts and feelings more openly than ever before, sentiment analysis is fast turning into a crucial technique for tracking and comprehending sentiment in all kinds of data.

Sentiment Analysis, also known as Opinion Mining and Emotion AI, is used to determine the opinions of the masses about a specific topic. It is contextual text mining that recognizes and extracts subjective information from source material. The most popular text categorization tool that determines if an incoming message is positive, negative, or neutral by analyzing the underlying sentiment. Polarity categorization is a crucial component of sentiment analysis. Polarity refers to the overall sentiment conveyed by a particular text, phrase or word. This polarity can be expressed as a numerical rating known as a 'sentiment score'. The polarity is considered as a class attribute, so solving the sentiment analysis task can be considered as solving classification task in combination with NLP methods for text analysis.

We analyzed the existing publications for solving this task. The publications include studies using TF-IDF [1] or Word2Vec [2] algorithms, as well as some deep learning architectures [3] including transformer models [4], [5], [6]. In [7], a detailed review of various approaches for solving this task are presented. In this paper we focus on the transformer models, which are the most popular nowadays and have shown as the most powerful models for solving various tasks.

First presented and described by Google in a 2017, transformer models are among the newest and one of the most powerful classes of models that exist today. For sure they are driving a wave of advances in machine learning and a paradigm shift in AI according to a 2021 paper by Stanford researchers. Transformer model is a neural network that learns context and meaning by tracking relationships in sequential data (like the words in a sentence). These models apply an evolving set of mathematical techniques, called attention or self-attention, in order to detect subtle ways to make even distant elements in a series dependent on each other. Since its debut in 2017, the transformer architecture has evolved and branched out into many different variants, expanding beyond language tasks into other areas. Transformer models are applied in many areas for a variety of purposes. They are translating text and speech in near real-time, helping researchers the chains of genes in DNA, detecting trends and anomalies to prevent fraud, making recommendations etc. Google is also using it to enhance its search engine results. Every sequential text, image or video is a great candidate for transformer models. Created with large datasets, transformers make accurate predictions that drive their wider use, generating more data that can be used to create even better models. Before transformers arrived, users had to train neural networks with large, labeled datasets that were costly and time-consuming to produce. By finding patterns between elements mathematically, transformers eliminate that need. Like most neural networks, transformer models are basically large encoder/decoder blocks that process data. Transformers use positional encoders to tag data elements coming in and out of the network. Attention units follow these tags, calculating a kind of algebraic map of how each element relates to the others. With these tools, computers can see the same patterns the humans see.

The aim of this paper is to build a model for solving sentiment analysis task for Amazon reviews data. For that purpose, we use the BERT model [8] as one of the most well-known transformer models. The rest of this paper is organized in the following way. In section 2 we present the dataset that is used. In section 3 we present our approach, and we give details regarding data preparation, the BERT model and the process of training and testing the model. In section 4 we present some experimental results of the evaluation of the

sentiment analysis model. Finally, section 5 concludes the paper and identifies some directions for further research.

II. Dataset

In this research, we used the Amazon Earphones Reviews dataset [9], which contains 14337 Amazon reviews, star ratings, for 10 latest (as of mid-2019) bluetooth earphone devices. We used this data to analyze what customers are saying about Bluetooth earphone devices, discover insights into consumer reviews and train our model to determine whether a review is positive or negative. As shown in Table 1, each record consists of a ReviewTitle, a preview of the review, a ReviewBody (the review in detail), ReviewStar (rating that sums up the review) and Product column with the name of the product for which the review was left. We use ReviewBody and ReviewStar in the analysis made in this research.

On Fig. 1 we give evidence about the distribution of the dataset based on the class attribute ReviewStar that ranges between 1 and 5. It could be seen that the largest class is the class with the reviews with Review score (the attribute ReviewStar) equal to 5, then the classes with reviews with Review score 4, 1 and 3 follow, while the class for the reviews with Review score 2 is the smallest one.

III. Our Approach

Our approach for sentiment analysis is illustrated on Fig. 2. First, data preprocessing is made where we used three values for scoring, such as positive, negative and neutral. Pre-trained BERT model is used, and it is trained using the training set, and is fine-tuned using the validation set. In the evaluation, the test set is used to evaluate the obtained model.

TABLE I. Description of the Dataset

Attribute	Description
ReviewTitle	a preview of the review
ReviewBody	the review in detail
ReviewStar	rating that sums up the review
Product	name of the product for which the review was left

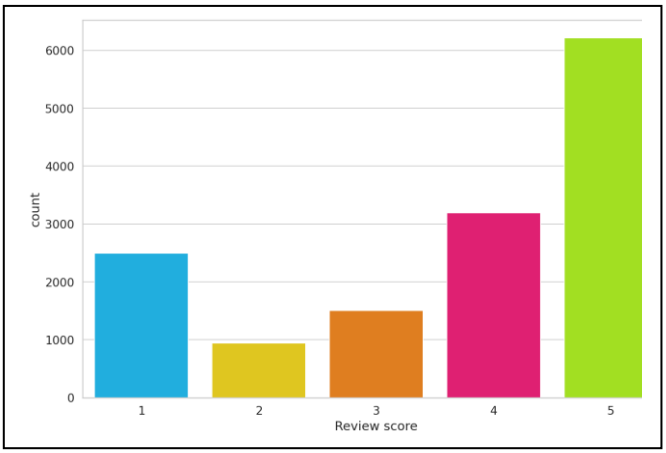

Fig. 1. Distribution in the initial dataset

A. Data Preparation

This section covers the data preparation and preprocessing steps that we did before training the transformer model. We converted the rating into negative, neutral and positive sentiment. So rating with 1 or 2 stars will be negative, 3 stars is neutral and 4 or 5 stars is positive. In this way, we obtained a data with a distribution as shown on Fig. 3. Next, we split the dataset into three smaller data sets: training set, validation set and testing set. The training set consists of 12903 reviews. The validation set that we will use to validate our model's performance during training consists of 717 reviews. The testing set will be used to evaluate the obtained model, and it consists of 717 reviews. In Table 2 we show the distribution in the training, validation, and test sets.

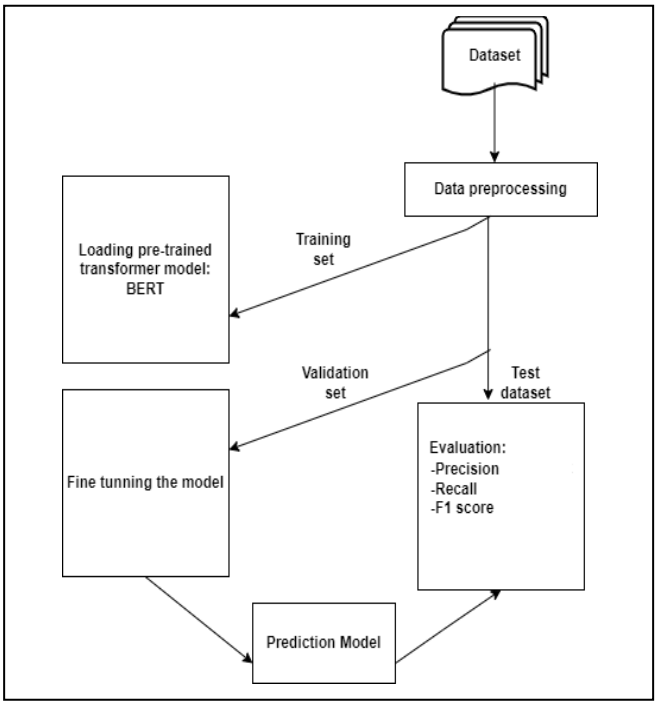

Fig. 2. Our approach for sentiment analysis

Fig. 3. Distribution in the final (converted) dataset

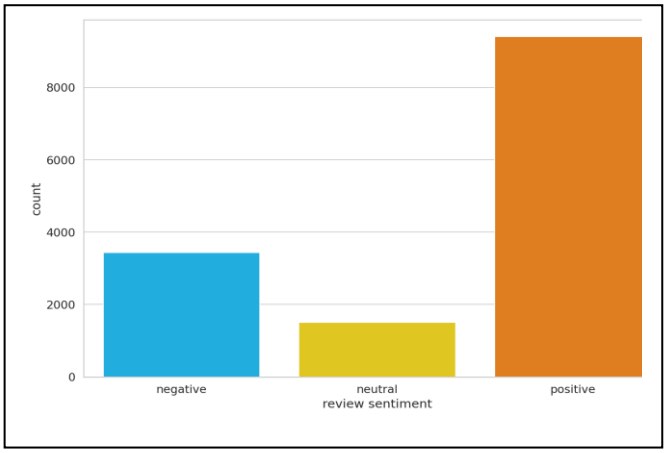

TABLE II. DISTRIBUTION OF THE CLASSES IN DIFFERENT SETS

Set	Class		
	positive	***neutral***	***negative***
Train	8457	1350	3096
Validation	471	75	171
Test	474	78	165

The next part of the data preparation is tokenization. Tokenization is the process of encoding a string of text into transformer-readable token ID integers. We used a pre-trained BertTokenizer to create the pipeline for tokenization as shown on Fig. 4. The first step of the tokenization process is the actual transformation of the review from the ReviewBody column into a sequence of tokens. Each string of text is translated into a token. The result is an array of tokens for each word or punctuation sign.

The next step in the tokenization process is to choose the maximum length of the sequences of tokens that we will use to train, validate and test our model. BERT works with fixed-length sequences. As shown on Fig. 5, most of the reviews seem to contain less than 128 tokens, but just to be on the safe side, we chose a maximum length of 160.

After the max length of the token sequence is chosen we, can continue on and map each token into an id that is readable by the transformers. When the algorithm is processing the sequences of tokens, it will need to know where a sequence starts and ends at least. For this reason we are using tokens for starting a sentence [CLS], for ending a sentence [SEP], for padding [PAD] and an unknown token [UNK] for everything else. Using all these tokens we create a token ID Tensor and based on which we create an attention mask. The transformer model will calculate attention for tokens in the token IDs tensor only if the attention mask tensor equals 1 at the respective position.

The last thing we have to do as part of the data preparation process is to create a PyTorch dataset and data loaders. PyTorch provides many tools to make data loading easy and to make our code more readable. It provides two data primitives: the torch.utils.data.DataLoader and the torch.utils.data.DataSet The DataSet library enables us to implement functions specific to the particular data, and the DataLoader is an iterator that provides batching, shuffling and loading the data.

B. BERT Model

A big advantage of the transformer models is that they can be trained through self-supervised learning or unsupervised methods. For example, BERT (Bidirectional Encoder Representations from Transformers) [8], that is a state-of-the-art machine learning model for NLP tasks, does much of its training by taking large amounts of unlabeled text, masking parts of it and trying to predict the missing parts. After that, BERT tunes its parameters based on how much its predictions were close or far from the actual data. By continuously going through this process, BERT captures the statistical relations between different words in different contexts. After this pre-training phase, BERT can be fine-tuned for a downstream task such as question answering, text summarization, or sentiment analysis by training it on a small number of labeled examples.

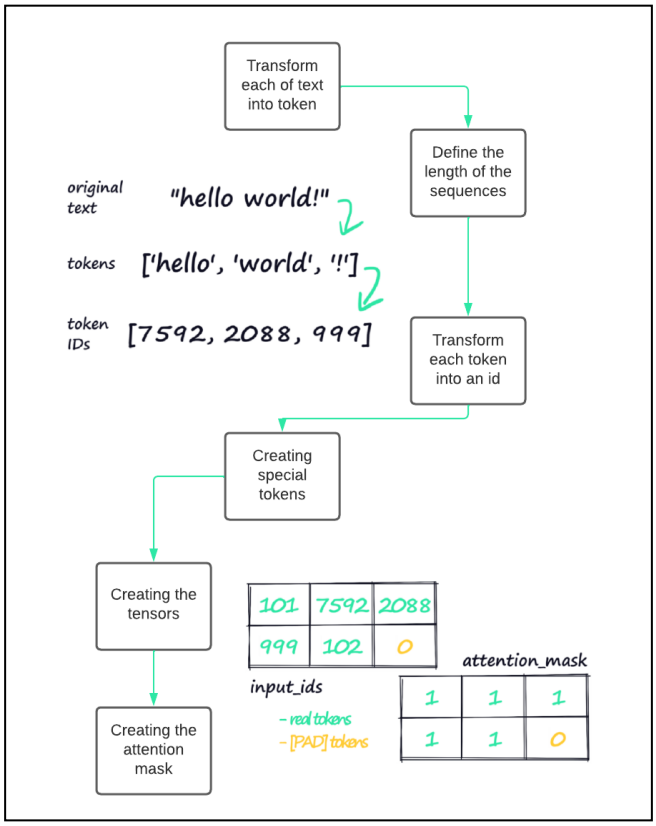

Fig. 4. Tokenization process

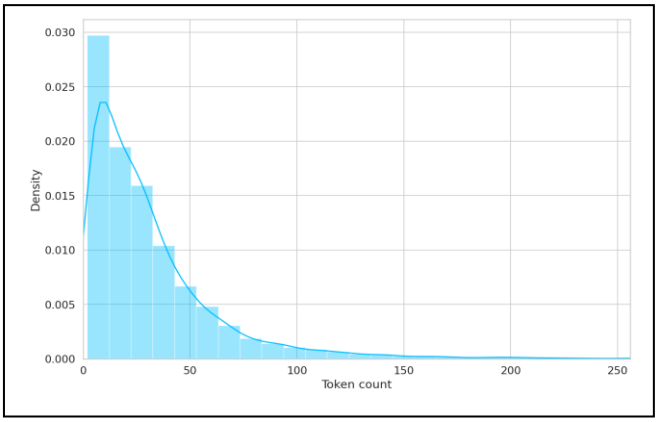

Fig. 5. Distribution of tokens

In our project we used the BERT base model that is a transformer model pre-trained on a large corpus of English data in a self-supervised fashion and is built of 12 encoders with 12 bidirectional self-attention heads.

C. Training the Model

Hugging Face [10] is a community and data science platform that provides tools that enable users to build, train and deploy ML models based on open source (OS) code and technologies. It is a place where a broad community of data scientists, researchers and ML engineers can come together and share ideas, get support and contribute to open source projects.

For training the model we used BERT-base-cased model [11] retrieved from Hugging Face that is a transformer model pre-trained on a large corpus of English data. This model is case-sensitive.

To create a Sentiment classifier that uses the BERT model, we will use the pooled-output. The pooled-output is obtained by applying the BertPooler on the last-hidden-state, which is a sequence of hidden states of the last layer of the model. The Sentiment classifier that we created delegates most of the heavy lifting to the BertModel. We used a dropout layer for some regularization and a fully-connected layer for our output. The optimal probability of retention is set to 0.3. For fine-tuning, we used the Adam optimizer that BERT was originally trained with. This optimizer minimizes the prediction loss and does regularization by weight decay (not using moments), which is also known as AdamW. After the training is completed, we looked at the training vs validation accuracy and we realized that the training accuracy starts to approach 100% after 10 epochs, see Fig. 6. In this paper, we trained the model for 10 epochs.

IV. Evaluation

In this section, we present the results from the evaluation of our model. We started by calculating the accuracy on the test data. Our model seemed to generalize well, since the accuracy is about 1% lower on the test set. This shows that the model is not overfitted. From the obtained results for Precision, Recall and F1-measure for each class shown in Table 3, we can see that our model is appropriately classifying both positive and negative reviews, however it is having difficulties classifying the neutral reviews. On Fig. 7 the confusion matrix is given. From this figure we can see that the model makes mistakes for the negative and positive classes at a roughly equal frequency. From the confusion matrix it is also evident that for each class the number of samples that are misclassified in the other classes is approximately equal to the number of samples from the other classes that are wrongly classified in the inspected class. Due to this, the Precision and Recall is identical or almost identical for the cases for all three classes, thus also leading to the same value for F1-measure as the value for Precision and Recall that are obtained for the particular class. This is very important and indicates that the model is not biased towards a given class.

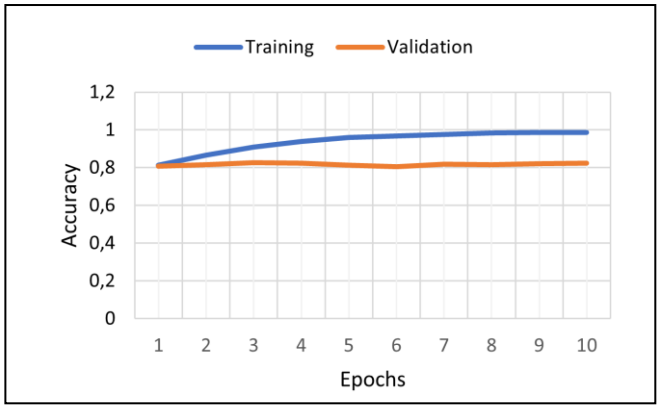

Fig. 6. Accuracy on the training and validation dataset over epochs

TABLE III. Experimental Results

Class	Evaluation measure		
	Precision	***Recall***	***F1-measure***
positive	0.92	0.92	0.92
neutral	0.36	0.36	0.36
negative	0.78	0.79	0.79

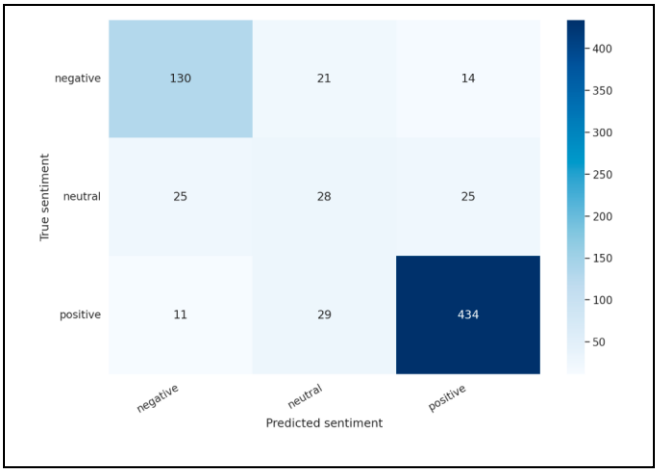

Fig. 7. Confusion matrix

We also made comparison of our approach with several existing approaches. For that purpose, we considered the models given in [12] and [13]. We want to mention that in the comparison, the division into training, validation and test set is different for all examined approaches. Moreover, in [12] a subset of the dataset is used in the experimental evaluation as described below. In this analysis we use the results reported by the authors of these models.

In [12], a fraction of the dataset was used where the reviews for three products are considered. TF-IDF was used for extracting the features and the model is built using logistic regression. It is also worth to mention that in [12] two classes are considered. Although the dataset is smaller, the results are not satisfactory since the class for negative sentiment obtains significantly lower recall than in our case. Namely, since the task at hand is to determine the sentiment of a given review, the customers are more interested in reading the negative reviews in order to make a decision whether to buy a given product, so the recall for the negative class is of highest interest.

The other model used in the comparison [13] is an LSTM model. This model gives accurate predictions for the neutral class with recall equal to 0.98 for this class, which is significantly better than the recall for the neutral class obtained with our model. However, the LSTM model misclassifies the samples from the other classes leading to lower recall (0.67 for the negative and 0.85 for the positive class). Although this LSTM model makes better predictions and correctly identifies the neutral reviews, it has significantly lower recall for the negative class that is of highest interest for this particular task for sentiment analysis of reviews for products. Namely the potential customers that are looking at the products are not interested in these neutral reviews, they are interested mostly to find out whether the previous customers that bought that product have stated some negative facts about that product that sometimes could be the decision factor whether they will make an order for that product. This is from customers point of view. On the other hand, from the point of view of the company that sells the products, the company is also interested to find out what is the negative thinking from the customers regarding their products in order to improve the products by adding the characteristics required by the customers etc.

From this analysis, we can conclude that our model outperforms the models given in [12] and [13], especially regarding the negative class.

We want to mention that the research made in [3] considers different deep learning architectures including BERT model. However, the analysis in [3] is made using another dataset, therefore those models are not considered in the comparison.

V. Conclusion

In this paper, we presented a study where we utilized NLP-based methods for sentiment analysis on Amazon reviews. In particular, this study has taken advantage of deep learning techniques via powerful state-of-the-art NLP models such as transformers. The study begins with data preparation and preprocessing. After data preparation and splitting the data in three smaller sets, the next part is tokenization. The next step was training and testing the model. For training we used the BERT-base-cased model retrieved from Hugging Face and we fine-tuned this model for solving the particular task.

We measured the performance of our model by considering the test reviews. The results shows that the model makes good predictions for the positive and negative reviews, but it has difficulties for the neutral reviews. However, if the positive and negative reviews are of higher interest, the model would be applicable for the particular task.

We also made an analysis to compare this model with several existing models obtained using this dataset. The model that utilizes TF-IDF and logistic regression, as well as the LSTM model are not good predictors of the negative class that is of highest interest for the potential buyers of the product. Due to that, our model is more appropriate for solving the task for sentiment analysis of products' reviews.

This study could be extended by applying other algorithms for creating prediction model. Other well-known classification models could be used, including other deep learning architectures and transformer models.

As future work we also plan to perform sentiment analysis using datasets in Macedonian language, as well as other languages from our geographical region. For that purpose, pretrained models would be used and would be fine-tuned for solving the task at hand. However, the biggest challenge for this will be to obtain labeled datasets that would be used in the analysis.

Acknowledgment

This work was partially financed by the Faculty of computer science and engineering at the "Ss. Cyril and Methodius University in Skopje", Skopje, Macedonia.

References

[1] B. Das and S. Chakraborty, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation".

[2] J. Acosta, N. Lamaute, M. Luo, E. Finkelstein, and A. Cotoranu, "Sentiment Analysis of Twitter Messages Using Word2Vec," Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS 2019), Granada, Spain, 2019.

[3] U. Singh, A. Saraswat, H.K. Azad, K. Abhishek, and S. Shitharth, "Towards improving e-commerce customer review analysis for sentiment detection," Scientific Reports, vol. 12, 21983, 2022.

[4] S.T. Kokab, S. Asghar, and S. Naz "Transformer-based deep learning models for the sentiment analysis of social media data," vol. 14, 100157, 2022.

[5] X. Gong, W. Ying, S. Zhong, and S. Gong, "Text Sentiment Analysis Based on Transformer and Augmentation," Front. Psychol., vol. 13, 906061, 2022.

[6] J. Hartmann, M. Heitmann, C. Siebert, and C Schamp, "More than a Feeling: Accuracy and Application of Sentiment Analysis," International Journal of Research in Marketing, https://doi.org/10.1016/j.ijresmar.2022.05.005.

[7] W. Trisna and H.J. Jie, "Deep Learning Approach for Aspect-Based Sentiment Classification: A Comparative Review," Applied Artificial Intelligence, vol. 36, no. 1, doi: 10.1080/08839514.2021.2014186, 2022.

[8] J. Devlin, M-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.

[9] https://www.kaggle.com/datasets/shitalkat/amazonearphonesreviews.

[10] https://huggingface.co/.

[11] https://huggingface.co/docs/transformers/model_doc/bert.

[12] https://www.kaggle.com/code/foolwuilin/sentiment-analysis-for-3-earphones/.

[13] https://www.kaggle.com/code/mervetas/sentiment-analysis-with-lstm-model.