Sentiment Classification of Customer Reviews Using LSTM and BERT-based Transformers

Carlos Eduardo Cerqueira, Felipe Freire, Ivon Luiz Garcia, Eduardo F. Simas Filho
Departamento de Engenharia Elétrica e de Computação
Universidade Federal da Bahia
{cerqueira.carlos, felipe.freire, ivon.luiz, eduardo.simas}@ufba.br

***Abstract*—In today's digital landscape, online customer reviews play a crucial role in shaping consumer decisions and business strategies. However, the vast volume of reviews makes manual sentiment analysis impractical. This paper presents a comparative study of Natural Language Processing (NLP) techniques for sentiment classification in Portuguese-language e-commerce reviews. We evaluate multiple approaches, including Deep Neural Networks (DNNs), Long Short-Term Memory (LSTM) networks, and fine-tuned BERT-based transformers, using a Brazilian e-commerce dataset. Despite challenges such as inconsistencies between review text and assigned ratings, our findings show that effective preprocessing paired with LSTM architectures yields competitive accuracy (88.24%), while fine-tuning a Portuguese BERT model further improves performance (91.07% accuracy). The proposed methods provide scalable solutions for businesses to analyze customer sentiment and enhance product offerings.**

***Index Terms*—Sentiment Analysis, Natural Language Processing, Customer Reviews, Deep Learning, BERT.**

I. Introduction

In the digital era, e-commerce has rapidly emerged as a dominant alternative to traditional brick-and-mortar retail stores. With the convenience of online shopping, customers are no longer bound by geographical or temporal constraints, leading to a significant shift in consumer behavior. In this vast and competitive global marketplace, where product options are abundant and constantly evolving, consumers increasingly rely on peer-generated content—particularly customer reviews—for reassurance and informed decision-making. These reviews serve as valuable sources of information, providing firsthand experiences and opinions that can significantly influence the final purchasing decision [1].

The openness, authenticity, and relatability of customer reviews contribute not only to a better understanding of products but also to the formation of a virtual community where consumers share insights and recommendations. This sense of community fosters trust and brand loyalty. For businesses operating in the digital landscape, online reviews have become a strategic asset. Companies that actively engage with customer feedback and integrate it into product development and service refinement are more likely to remain competitive and demonstrate a customer-centric approach. [2].

To remain responsive to customer needs and expectations, businesses must go beyond simply collecting reviews—they must understand the sentiments and opinions embedded within them. However, on large-scale platforms such as Amazon and eBay, the volume of customer-generated content is overwhelming, making manual review analysis impractical and inefficient [3]. This challenge has prompted the growing adoption of Natural Language Processing (NLP), a subfield of artificial intelligence focused on enabling machines to interpret and analyze human language.

In the context of customer feedback, NLP offers robust methodologies to extract meaningful insights from textual data. These include identifying themes, detecting sentiment, and classifying opinions. One of the most critical applications is sentiment analysis, which involves determining whether the expressed opinion in a piece of text is positive, negative, or neutral [4]. This technique can provide a high-level overview of how consumers perceive a product or service, highlighting strengths and exposing areas needing improvement.

Sentiment analysis—also referred to as opinion mining—is a computational approach to evaluating subjective information. It categorizes sentiments in user-generated text to determine the author's attitude toward a given entity, such as a product, service, or brand. Recent advancements in machine learning, particularly deep learning models like Long Short-Term Memory (LSTM) networks and Bidirectional Encoder Representations from Transformers (BERT), have significantly enhanced the accuracy and depth of sentiment analysis. These models are capable of understanding complex linguistic structures and contextual cues, making it possible to capture nuanced emotional tones and implicit meanings that traditional methods might miss.

In this paper, we present a tool designed to analyze customer sentiment within individual reviews. The tool aims to identify both the positive attributes and the shortcomings of a given product, offering actionable insights that businesses can use to improve customer satisfaction and product quality.

The remainder of this paper is organized as follows: Section II provides an overview of the current state of the art in NLP and sentiment analysis. In Section III, we introduce and describe the proposed sentiment analysis tool. Section IV presents the experimental results and analysis. Finally, Section V concludes the paper with a summary of key findings and outlines potential directions for future research.

II. Related Work

This work focuses on customer reviews written in Portuguese, a language that poses significant challenges in the

context of Natural Language Processing (NLP). Unlike English, which benefits from a rich ecosystem of pre-trained models, annotated datasets, and community resources, Portuguese has comparatively limited tools and resources available for NLP tasks. This disparity restricts the development of effective sentiment analysis models for the Portuguese language, especially in domains like e-commerce where nuanced customer opinions are key. In this section, we present related research that tackles similar challenges, specifically addressing the use of the Portuguese language and the application of NLP techniques for customer review analysis.

A survey presented in [5] addresses the work developed in the domain of Sentiment Analysis (SA) in Portuguese and states that further advances are still required to make better use of the language. In fact, due to the difference in maturity of available tools, they claim that translating data into English and then using tools developed for this language may lead to better results.

Duarte et al. [6] proposed an approach that used emojis and emoticons to reduce the impact of using other languages. They applied Naive Bayes and Support Vector Machine (SVM) for classifying and predicting emojis. Although useful, this approach can be limiting as not all reviews contain emoticons or emojis, and these symbols may not fully express the sentiment conveyed in the entire text.

Saal et al. [7] used a public corpus of tweets for SA and traditional classifiers (SVM, Random Forest, Decision Trees, and Logistic Regression). They concluded that Decision Trees outperformed the other algorithms. This study employed Term Frequency-Inverse Document Frequency (TF-IDF) to represent each tweet, a traditional approach that weights the importance of words in a document relative to a collection of documents. Other studies compared this representation with one produced by a Bidirectional Encoder Representations from Transformers (BERT) [8] model pre-trained for Portuguese [9], concluding that while TF-IDF presents a good balance of computational cost and performance, BERT representations typically achieve higher scores.

Several current studies rely on transformers such as BERT. Roy et al. [10] compared various traditional approaches with a BERT-based model for SA, in particular hate detection. They determined that approaches like Logistic Regression, K-NN, Naive Bayes, and Long Short-Term Memory Networks (LSTM) can perform equally or even better than BERT in certain contexts. Fernandez et al. [11] used manually-labeled data and an Indonesian BERT model for SA on Indonesian messages containing slang. This approach outperformed previous studies and achieved high accuracy in sentiment prediction and slang recognition. Transformers have also been combined with other models such as Conditional Random Field (CRF), LSTM, or simple fully connected layers for various languages [12], [13].

CRF has been commonly used for sequence tagging tasks, such as Named Entity Recognition (NER) or Part-of-Speech tagging, due to its architecture that allows for contextual exploitation [14]. Souza et al. [15] used a BERT-CRF architecture for NER in Portuguese, achieving better performance than studies using a fine-tuning approach. They suggest future experimentation with another transformer, RoBERTa [16], which they claim can be more efficient. Tan et al. [17] combined this model with an LSTM classifier for SA in English, achieving high F1-scores across different datasets. However, none of these previous works considered context for SA.

Ling et al. [18] used a concatenation strategy to include context in short dialogues, addressing the first of three factors mentioned; however, their work focused on response generation. Wang et al. [19] applied SA to customer service dialogues and proposed a topic-aware approach, which also seems to tackle the first factor. They experimented with several classifiers, including BERT, LDA-LSTM, and LDA-BERT, and various multi-task scenarios involving topic information, which allowed them to outperform several baselines.

Another approach for considering context is Few-Shot Learning (FSL) [20], which allows for the meta-training of classifiers with just a few labeled samples. Large models, e.g. GPT-4 [21], are typically used for this approach, with their main task being text generation. These large models are trained on large amounts of data, primarily based on The Pile [22]. Due to their extensive training, these models are more likely to generalize with minimal extra information. Hosseini-Asl et al. [23] employed GPT-2 in a few-shot learning strategy to perform aspect-based SA. Their results showed that GPT-2 [24] outperformed BERT-based approaches while using less than 20% of the training data.

III. Proposed Tool

In this section, we present a high-level overview of the tool architecture and the procedures utilized to obtain the results. As illustrated in Figure 1, the proposed systems comprises preprocessing techniques (such as metadata characters removal, stemming and tokenization), followed by a deep neural network-based system trained to predict the customers sentiment.

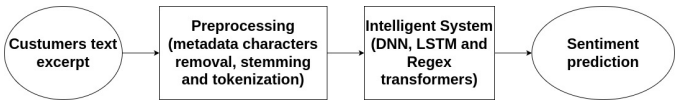

Fig. 1. Proposed System.

A. Dataset

We utilized the Brazilian E-Commerce dataset [25], a public dataset with information from 100k orders in Portuguese from 2016 to 2018 made across multiple marketplaces in Brazil. After dropping all the reviews with an empty comment field, we retained 41k reviews with Portuguese comments. The organization of the dataset can be visualized in Figure 2, which shows the relationship between the various tables available.

The dataset contains a 1 to 5 star rating scale, a standard convention for customer reviews. Scores of 4 (good) and 5 (excellent) typically signify a positive experience, while scores of 1 and 2 represent clear dissatisfaction. The middle ground

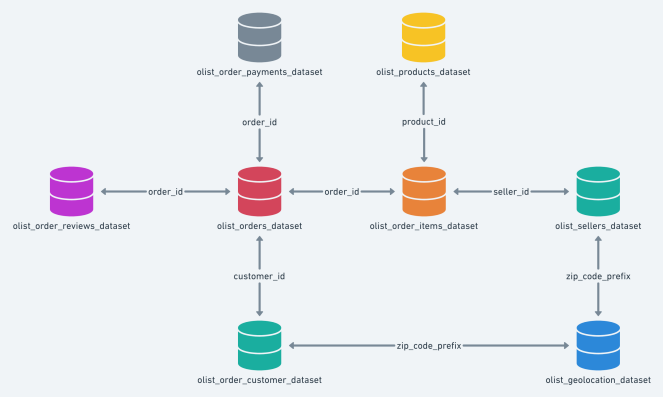

Fig. 2. Organization of Olist dataset.

score of 3 should indicate a neutral or mixed opinion, signaling that the customer was not fully satisfied. For the proposed tool, we considered a binary classification problem, categorizing customer reviews into a positive feedback group and a negative one. A threshold was established where ratings greater than or equal to four are labeled as 'positive', or labeled as 'negative' otherwise. This approach allows for a clear distinction between customers who are satisfied and those who express any level of dissatisfaction, which should be characteristics captured by our models. An example of the data used can be found in Table I.

TABLE I
ORIGINAL DATA COMMENTS WITH CORRESPONDING SCORES

Id	Score	Comment
0	5	Recebi bem antes do prazo estipulado.
1	5	Parabéns lojas lannister adorei comprar pela I...
2	4	aparelho eficiente. no site a marca do aparelh...
3	4	Mas um pouco ,travando...pelo valor ta Boa.
4	5	Vendedor confiável, produto ok e entrega antes...

Before training, the dataset was split into 80% for training and 10% each for testing and validation, respectively.

B. Preprocessing

After changing the dataset to the desired format, we applied preprocessing techniques to ensure good quality data, resulting in better models performance.

At this stage, several regular expressions (Regex) are used. These are sequences of characters that define a search pattern, used in this context to find and manipulate specific textual patterns, such as removing, standardizing, or replacing them to facilitate feature extraction by our models.

1) Metadata characters removal: For the first step of preprocessing, we replaced specific textual elements (e.g., dates, currency, numbers, URLs, and stop words) with metadata that represents their existence while preserving the context of the phrase. By implementing this approach, we were able to generalize concepts, making it easier for the model to extract meaningful information from the data.

Stop words are frequently occurring words that provide little semantic value. By removing them, we reduce the dimensionality and noise in the data. To perform this task, we utilized the Natural Language Toolkit (NLTK) [34] list for Portuguese stop words.

TABLE II
METADATA REMOVAL EXAMPLES

Before	After
... dia 14/12/17 empresa falsa ...	... dia data empresa falsa ...
... impresso como 3desinfector ...	... impresso como numero desinfector ...

2) Stemming: Stemming is a text normalization process used in Natural Language Processing (NLP) to reduce words to their root or base form. The goal is to group together different forms of a word so they can be analyzed as a single item. For example, in Portuguese, words like "correndo" (running), "corredor" (runner), and "correu" (ran) can all be reduced to the root word "corr-". This process helps reduce vocabulary size and allows the model to identify words with similar meanings despite different suffixes.

3) Tokenization: As the final step, words were transformed into tokens. Following the stemming process, it's assured that different words with the same radical are represented by the same token, making the context more generic. This approach enables the neural network to extract more information from semantically similar words. In essence, the tokenization process creates a numerical representation of the text that can be efficiently processed by machine learning models.

C. Neural Networks

For the SA, we evaluated the performance of distinct models. These architectures were specifically chosen to provide a comprehensive comparison, exploring a logical progression in both model complexity and feature engineering (when done):

- **Deep Neural Network (DNN):** Used as a foundational baseline. Its simple, non-sequential architecture helps to establish a performance benchmark and highlights the benefits of more complex, sequence-aware models.
- **Long Short-Term Memory (LSTM) network:** As a powerful type of recurrent neural network (RNN), the LSTM was chosen because it is specifically designed to learn from sequential data such as text. It represents a classic and effective deep learning approach for capturing word order and contextual dependencies.
- **A Dense Model with TF-IDF Vectorizer:** DNN with more robust text preprocessing Count Vectorizer, Bag of Words, TF-IDF, and Word2Vec. Used to compare with standard model without the feature engineering and try to match sequence-aware architectures, such as LSTMs.
- **BERT-based transformer fine-tuned:** Included to benchmark our results against the current state-of-the-art in NLP. Pre-trained transformers like BERT excel at understanding deep contextual nuances in language, providing a high-performance standard for comparison.

This comparative approach allows us to systematically evaluate the trade-offs between model complexity and performance on our specific task. A key finding this study explores is whether an effective preprocessing pipeline can elevate a simple model's performance to match or exceed that of more complex architectures.

IV. Results

In this section, we discuss the results obtained by the different Neural Networks.

1) Deep Neural Network: The neural network used is a dense model implemented using TensorFlow [26]. It comprises an input layer, three hidden layers with 50 neurons each using ReLU activation, and dropout layers to prevent overfitting. The output layer utilizes a sigmoid activation function for binary classification. The performance of this model, including accuracy and loss curves, are shown in Figure 3.

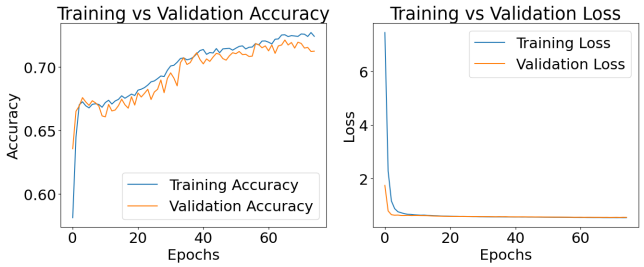

Fig. 3. Accuracy and Loss of the DNN.

This model achieved 71.84% accuracy on the test set, indicating potential for further enhancement.

2) LSTM: Using the same dataset, an LSTM model was developed with TensorFlow, achieving improved results as anticipated. This model, configured with two layers of 128 units each, attained an accuracy of 87.95%.

These findings suggest that LSTM networks are well-suited for the task. We therefore conducted further experiments using K-Fold Cross-Validation, exploring different network architectures by varying the number of hidden layers and neurons per layer. Following best practices from [30], the training process evaluated

This indicated that LSTM networks could be a good solution for this problem. Consequently, we conducted further experiments using K-Fold Cross-Validation to explore different network architectures, varying both the number of hidden layers and neurons per layer. Following best practices from [30], the training process evaluated 9 different configurations (3 different number of layers $\times$ 3 options for number of neuron $\times$) with 5 folds, resulting in 45 different initializations for training. Early stopping was also used to prevent overfitting and speed up the training process.

The optimal configuration had a mean validation accuracy of 89.41%, using three hidden layers with 32 neurons. After retraining this architecture with early stopping, the resulting learning curve is shown in Figure 4, and its performance on the test set is presented in the confusion matrix in Table III, with 88.24% accuracy.

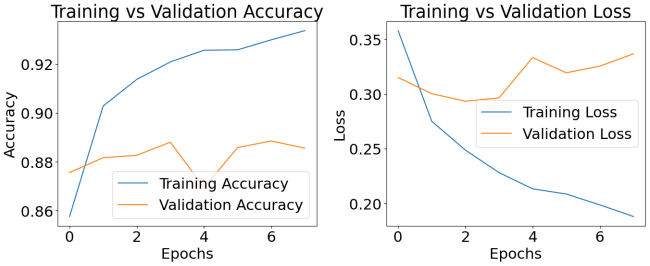

Fig. 4. Accuracy and Loss of the LSTM.

TABLE III
Confusion Matrix of the LSTM.

Actual / Predicted	Positive	Negative
Positive	1112	286
Negative	196	2504

3) Dense Model with Regex Transformers + TF-IDF Vectorizer: After applying regular expressions, stopword removal, and stemming, several vectorization methods were utilized, including Count Vectorizer, Bag of Words, TF-IDF, and Word2Vec to extract meaning from the data.

Count vectorization is a technique in NLP that converts text documents into a matrix of token counts. Tokens can be words, characters, or n-grams. Each token represents a column in the matrix, and the resulting vector for each document has counts for each token.

In the Bag of Words approach, a vocabulary is constructed from unique words. Each document is then represented as a vector indicating the presence (1) or absence (0) of each word. However, this method treats all words equally, which may not reflect their true importance. To address this, TF-IDF (Term Frequency–Inverse Document Frequency) was used, implemented via scikit-learn [27], using the following formulas:

$$TF = \frac{\text{Frequency of a word in the document}}{\text{Total words in the document}} \tag{1}$$

$$IDF = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents with the word}}\right) \tag{2}$$

TF-IDF reduces the weight of frequently occurring words that are less informative, thereby improving the performance of text-based machine learning models.

Using this pre-processing pipeline, we trained several dense model configurations similar to the approach in Section IV-1. The best model achieved an accuracy of 88.51%, comparable to the LSTM.

TABLE IV
Confusion Matrix of the DNN with Regex Transformers and TF-IDF Vectorizer.

Actual / Predicted	Positive	Negative
Positive	1156	242
Negative	229	2471

These results indicates that, with effective pre-processing, a simple dense model can perform comparably to more complex RNN architectures like the LSTM.

4) Fine-Tuning BERT-based transformer: In this work, we utilized the pre-trained 'neuralmind/bert-base-portuguese-cased' model, a BERT-based transformer model specifically trained on a large corpus of Portuguese text, to perform sentiment analysis. This model was fine-tuned on our labeled dataset to classify the comments.

The fine-tuning process involved adding a classification head to the pre-trained model and training it using the PyTorch framework and the Hugging Face Transformers library [28]. The dataset was split was kept the same as previous. During training, we used the AdamW optimizer with a learning rate of 2×10^{-5} and a linear learning rate scheduler with warm-up steps. Early stopping was employed to prevent overfitting, with a patience of 3 epochs.

The training loop included the following steps:

- Tokenizing the input text using the tokenizer provided by the 'neuralmind/bert-base-portuguese-cased' model.
- Feeding the tokenized input (input IDs and attention masks) into the model.
- Computing the loss using the model's built-in loss function for classification tasks.
- Backpropagating the loss and updating the model's weights.

During fine-tuning, the best-performing model on the validation was saved to be used later on the testing phase. It achieved a test accuracy of 91.07%, surpassing previous approaches.

TABLE V
CONFUSION MATRIX OF THE BERT-BASED TRANSFORMER.

Actual / Predicted	Positive	Negative
Positive	1279	189
Negative	177	2453

A. Test data comparison

The performance of each classification model was evaluated using standard metrics: accuracy, precision, recall, and F1-score. The results are presented in Table VI. A clear performance hierarchy is visible among the evaluated approaches.

TABLE VI
CLASSIFICATION REPORT BY MODEL ON TEST DATA

Model	Class	Precision	Recall	F1-score	Accuracy
Dense	0	0.72	0.68	0.70	0.72
	1	0.73	0.77	0.75	
LSTM	0	0.84	0.81	0.83	0.88
	1	0.90	0.92	0.91	
Dense + TF-IDF	0	0.83	0.83	0.83	0.89
	1	0.91	0.92	0.91	
BERT	0	0.88	0.87	0.87	0.91
	1	0.93	0.93	0.93	

The basic Dense architecture (IV-1) achieved the lowest scores, due to its limited capacity to capture the complexity of textual data. Introducing TF-IDF preprocessing (IV-3) significantly boosted the Dense model's performance—bringing its F1-score on par with the LSTM model (IV-2) and even slightly surpassing it in overall accuracy. This demonstrates how effective preprocessing and feature engineering can enhance simpler models, making them viable alternatives in resource-constrained settings.

The BERT-based transformer model (IV-4) outperformed all other approaches across every metric. This superior performance reflects the benefits of leveraging large pre-trained language models, which excel in handling nuanced linguistic features such as sarcasm, negation, and context-dependent word meanings.

B. Misclassifications review

To better understand the limitations of our models, we performed a qualitative analysis of misclassified examples. Table VII presents a sample of reviews where the model's predictions diverged from the ground truth.

TABLE VII
EXAMPLES OF MISCLASSIFIED REVIEWS.

Review	True	Pred	Observation
Me arrependi de não ter comprado antes!	Pos	Neg	"Arrependi" suggests regret, despite a positive meaning.
Entrega super rápida, mas o produto é ruim.	Neg	Pos	Positive start overshadowed the negative conclusion.
Produto chegou conforme esperado. Para quem estiver interessado, pode mandar vê!!	Neg	Pos	Tone is positive despite the negative label.
a espera do reembolso. ou cancelarei o pagamento	Neg	Pos	Mentions refund, but gave a good score.
entrega rápida e de boa qualidade	Neg	Pos	Positive comment with an unexpected negative label.
Sem comentários	Neg	Pos	No content for inference.
Bom	Neg	Pos	Short positive word with a negative label.
Recomendo	Neg	Pos	Single-word recommendation misaligned with score.
Produto como na descrição. Mas demorou muito pra chegar.	Pos	Neg	Delay outweighed the otherwise positive review.

These misclassifications highlight broader challenges in sentiment analysis. One particularly difficult aspect is detecting sarcasm, which often inverts the literal sentiment of words. In these cases, the actual emotional tone is not reflected in the surface text, and, without context, models have difficulty detecting the author's true intent.

Another problem arises from polysemous words, whose meanings vary depending on the context.

Finally, additional errors occur when there is a mismatch between the sentiment expressed in the comment and the numerical rating provided by the user, such as a positive textual review accompanied by a low rating. Moreover, it is important to note that the accuracy of the model is inherently limited by inconsistencies and ambiguities in the user-generated content

itself. Human errors, such as contradictory statements, typos, or unclear expressions, introduce noise that can mislead even the most advanced models.

Such cases underscore that even humans can struggle with sentiment judgment in ambiguous reviews. Despite these complexities, transformer-based architectures like 'neuralmind/bert-base-portuguese-cased' demonstrate strong performance and generalization capabilities. However, these subtle linguistic challenges remain more problematic for smaller or less context-aware models, such as the ones presented on IV-2 and IV-3.

V. Conclusion

This paper presented a comparison of several models using an NLP network for SA in a customer-based dataset. With the huge amount of data collected each day in websites like Amazon and EBay, online customer reviews have become essential for businesses that want to be competitive, responsive and customer centric in the digital marketplace.

Given the vast volume of data, manual review by humans is impractical. Generating NLP models like the ones proposed here becomes a viable alternative for businesses aiming to remain competitive, responsive, and customer-focused in the digital marketplace in the costumer response e-commerce analysis.

The results demonstrated that effective preprocessing, combined with relatively simple architectures, can yield strong performance in sentiment classification tasks. Notably, the LSTM and Dense + TF-IDF models achieved overall accuracies of 88.51% and 88.24%, respectively, highlighting the potential of well-optimized lightweight approaches. However, the BERT-based model outperformed all others, achieving an accuracy of 91.07% on the test set. This underscores the model's superior ability to understand contextual text data.

The choice between lightweight and more complex models should be guided by specific constraints and application requirements, such as hardware limitations, latency, or real-time processing needs. Additionally, the quality and consistency of the input data play a crucial role in determining the level of model robustness required.

For future work, is possible to evaluate additional model architectures and compare their performance with those presented here. Further, applying and fine-tuning transformer models across multiple languages would enable handling international review datasets and increase model applicability. Another promising direction is the extraction of sentiment related to specific product features—such as price, delivery, or quality.

VI. Acknowledgments

The authors would like to thank CNPq and CAPES for the financial support.

References

[1] Y. K. Dwivedi et al., "Setting the future of digital and social media marketing research: Perspectives and research propositions," Int J Inf Manage, vol. 59, p. 102168, 2021.

[2] R. Thakur, "Customer engagement and online reviews, " Journal of Retailing and Cosumer Services, vol 41, pp. 48-59, 2018.

[3] E. Sezgen, K. J. Mason, and R. Mayer, "Voice of airline passenger: A text mining approach to understand customer satisfaction," Journal of Air Transport Management, vol. 77, pp. 65-74, 2019

[4] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," Information fusion, vol. 36, pp. 10-25, 2017

[5] D. A. Pereira, "A survey of sentiment analysis in the Portuguese language," Artif. Intell. Rev., vol. 54, no. 2, pp. 1087–1115, Feb. 2021.

[6] L. Duarte, L. Macedo, and H. Gonçalo Oliveira, "Exploring emojis for emotion recognition in Portuguese text," in Proc. 19th Conf. Artif. Intell. (EPIA), vol. 11805. Vila Real, Portugal: Springer, Sep. 2019, pp. 719–730.

[7] S. E. Saad and J. Yang, "Twitter sentiment analysis based on ordinal regression," IEEE Access, vol. 7, pp. 163677–163685, 2019.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pretraining of deep bidirectional transformers for language understanding,"in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., vol. 1, Jun. 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[9] F. Souza, R. Nogueira, and R. Lotufo, "BERTimbau: Pretrained BERT models for Brazilian Portuguese," in Proc. Brazilian Conf. Intell. Syst. (BRACIS), vol. 12319. Cham, Switzerland: Springer, 2020, pp. 403–417.

[10] S. S. Roy, A. Roy, P. Samui, M. Gandomi, and A. H. Gandomi, "Hateful sentiment detection in real-time tweets: An LSTM-based comparative approach," IEEE Trans. Computat. Social Syst.

[11] E. Fernandez, Anderies, M. G. Winata, F. H. Fasya, and A. A. S. Gunawan, "Improving IndoBERT for sentiment analysis on Indonesian stock trader slang language," in Proc. IEEE Int. Conf. Internet Things Intell. Syst. (IoTaIS), Nov. 2022, pp. 240–244.

[12] R. Bensoltane and T. Zaki, "Towards Arabic aspect-based sentiment analysis: A transfer learning-based approach," Social Netw. Anal. Mining, vol. 12, no. 1, pp. 1–16, Dec. 2022.

[13] M. Li, L. Chen, J. Zhao, and Q. Li, "Sentiment analysis of Chinese stock reviews based on BERT model," Int. J. Speech Technol., vol. 51, no. 7, pp. 5016–5024, Jul. 2021.

[14] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015, arXiv:1508.01991.

[15] F. Souza, R. Nogueira, and R. Lotufo, "Portuguese named entity recognition using BERT-CRF," 2019, arXiv:1909.10649.

[16] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, arXiv:1907.11692.

[17] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network," IEEE Access, vol. 10, pp. 21517–21525, 2022.

[18] Y. Ling, Z. Liang, T. Wang, F. Cai, and H. Chen, "Sequential or jumping: Context-adaptive response generation for open-domain dialogue systems," Appl. Intell., vol. 53, pp. 11251–11266, Sep. 2022.

[19] J. Wang, J. Wang, C. Sun, S. Li, X. Liu, L. Si, M. Zhang, and G. Zhou, "Sentiment classification in customer service dialogue with topic-aware multi-task learning," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 5, 2020, pp. 9177–9184.

[20] M. G. Miller, N. E. Matsakis, and P. A. Viola, "Learning from one example through shared densities on transforms," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2000, pp. 464–471.

[21] OpenAI, "GPT-4 technical report," 2023, arXiv:2303.08774

[22] L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, J. Phang, H. He, A. Thite, N. Nabeshima, S. Presser, and C. Leahy, "The pile: An 800GB dataset of diverse text for language modeling," 2021, arXiv:2101.00027.

[23] E. Hosseini-Asl, W. Liu, and C. Xiong, "A generative language model for few-shot aspect-based sentiment analysis," 2022, arXiv:2204.05356.

[24] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," OpenAI Blog, vol. 1, no. 8, p. 9, 2019.

[25] Olist. 2024, November. Brazilian E-Commerce Public Dataset by Olist. Retrieved November 20, 2024 from https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce.

[26] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey

Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

[28] Hugging Face Transformers library. from: https://huggingface.co/

[29] Agarap, A.F., 2018. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.

[30] Scikit-learn developers, "3.1. Cross-validation: evaluating estimator performance," scikit-learn, [Online]. from: https://scikit-learn.org/stable/modules/cross_validation.html.

[31] Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou, "Semantics-aware BERT for language understanding," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 5, 2020, pp. 9628–9635.

[32] I. Carvalho, H. G. Oliveira and C. Silva, "The Importance of Context for Sentiment Analysis in Dialogues," in IEEE Access, vol. 11, pp. 86088-86103, 2023, doi: 10.1109/ACCESS.2023.3304633.

[33] K. Arava, R. S. K. Chaitanya, S. Sikindar, S. P. Praveen, and D. Swapna, "Sentiment analysis using deep learning for use in recommendation systems of various public media applications," in Proc. 3rd Int. Conf. Electron. Sustain. Commun. Syst. (ICESC), Aug. 2022, pp. 739–744.

[34] Bird, S., Klein, E., Loper, E., 2009. Natural language processing with Python: analyzing text with the natural language toolkit, "Reilly Media, Inc." from: https://www.nltk.org/