electronics

Article

Sentiment Analysis in Portuguese Restaurant Reviews: Application of Transformer Models in Edge Computing

Alexandre Branco [1,2,†], Daniel Parada [1,2,†], Marcos Silva [1,2], Fábio Mendonça [1,2,*], Sheikh Shanawaz Mostafa [2] and Fernando Morgado-Dias [1,2]

[1] Faculty of Exact Sciences and Engineering, University of Madeira, 9000-082 Funchal, Portugal; alexandre.branco@arditi.pt (A.B.); daniel.parada@arditi.pt (D.P.); 2073511@student.uma.pt (M.S.); morgado@staff.uma.pt (F.M.-D.)
[2] Interactive Technologies Institute (ITI/LARSyS and ARDITI), 9020-105 Funchal, Portugal; sheikh.mostafa@tecnico.ulisboa.pt
[*] Correspondence: fabioruben@staff.uma.pt
[†] These authors contributed equally to this work.

Abstract: This study focuses on improving sentiment analysis in restaurant reviews by leveraging transfer learning and transformer-based pre-trained models. This work evaluates the suitability of pre-trained deep learning models for analyzing Natural Language Processing tasks in Portuguese. It also explores the viability of utilizing edge devices for Natural Language Processing tasks, considering their computational limitations and resource constraints. Specifically, we employ bidirectional encoder representations from transformers and robustly optimized BERT approach, two state-of-the-art models, to build a sentiment review classifier. The classifier's performance is evaluated using accuracy and area under the receiver operating characteristic curve as the primary metrics. Our results demonstrate that the classifier developed using ensemble techniques outperforms the baseline model (from 0.80 to 0.84) in accurately classifying restaurant review sentiments when three classes are considered (negative, neutral, and positive), reaching an accuracy and area under the receiver operating characteristic curve higher than 0.8 when examining a Zomato restaurant review dataset, provided for this work. This study seeks to create a model for the precise classification of Portuguese reviews into positive, negative, or neutral categories. The flexibility of deploying our model on affordable hardware platforms suggests its potential to enable real-time solutions. The deployment of the model on edge computing platforms improves accessibility in resource-constrained environments.

Keywords: sentiment analysis; natural language processing; Portuguese language; edge computing; BERT; transformers

Citation: Branco, A.; Parada, D.; Silva, M.; Mendonça, F.; Mostafa, S.S.; Morgado-Dias, F. Sentiment Analysis in Portuguese Restaurant Reviews: Application of Transformer Models in Edge Computing. *Electronics* **2024**, *13*, 589. https://doi.org/10.3390/electronics13030589

Academic Editors: Arkaitz Zubiaga and Ping-Feng Pai

Received: 27 November 2023
Revised: 23 January 2024
Accepted: 30 January 2024
Published: 31 January 2024

1. Introduction

Machine Learning (ML) and natural language processing (NLP) have evolved with increased computational power and data availability and have been applied in various fields, including language translation and sentiment analysis (SA), as demonstrated in various scientific articles [1–3]. SA, a subfield of NLP, extracts sentiments from text and more recently, usually employ deep learning (DL) models [4,5].

Classifying sentiments in restaurant reviews are a prominent problem in the NLP area [6–8], as it demands substantial annotated data and computational resources. Automated SA can help monitor restaurant reputation, identify common customer issues, improve restaurant brand monitoring, influence business decision making, and improve products and services.

Developing efficient DL methods is of utmost relevance, and the standard approach in NLP is to employ transfer learning (TL), leveraging pretrained models like bidirectional encoder representations from transformers (BERT) [1], robustly optimized BERT pretraining approach (RoBERTa) [2], and generative pretraining transformer 2 (GPT-2) [9]. In SA, TL can involve using a pre-trained DL model to extract features for sentiment classification. In this work, the pre-trained DL models were transferred and then used as weak learners in ensemble learning (EL) with boosting.

The classification of sentiment in restaurant reviews in Portuguese, involves determining the overall sentiment expressed in a written review. However, classifying sentiments based on text is challenging due to ambiguity, irony and sarcasm, cultural context, subjectivity, and language evolution. Furthermore, there is a lack of data for the Portuguese-language restaurant reviews.

More precisely, this work focuses on improving sentiment classification in restaurant reviews using DL techniques. The primary goal is to enhance the accuracy (ACC) above 0.8, as this is a reasonable target in NLP problems of sentiment classification, categorizing them as negative, neutral, or positive. This research also aims to make models deployable on various low-cost hardware platforms, potentially enabling real-time solutions. This approach could have applications in social media monitoring, market research, restaurant brand monitoring, business decision making, and customer feedback analysis, offering a valuable tool for researchers and practitioners in the field of SA.

The challenge at hand revolves around the current state of knowledge, which indicates a lack of NLP models specifically tailored for sentiment classification in Portuguese restaurant reviews. This issue holds significance as restaurant owners can leverage sentiment analysis to pinpoint the areas for improvement. For instance, negative sentiments in reviews can highlight areas requiring attention, while positive sentiments can identify aspects praised by customers. The neutral class facilitates an evaluation of potential areas for enhancement.

While generic solutions may involve using LLM, these often come with associated costs, depending on token usage, or may necessitate substantial computational power if opting for open models locally. Consequently, this work proposes a tailored solution, avoiding the need for extensive computational resources and achieving high performance in sentiment analysis.

The structure of this article is as follows: Section 2 provides a theoretical introduction to provide a comprehensive understanding of the subject; In Section 3, related work is presented; Section 4 provides the research problematic; Section 5 presents the materials and methods; Section 6 presents and discusses the results; Section 7 concludes the article.

2. Theoretical Background

This section describes a comprehensive theoretical introduction for the developed solution. Two transformer-based approaches (applying TL) and two ensemble-based models were examined for this SA classification problem. The transformer-based models were BERT and RoBERTa. The ensemble model was performed using Adaptive Boosting (Adaboost) with two classifiers and soft voting.

2.1. Transfer Learning

TL refers to leveraging the knowledge from one task to improve performance on another related task. This approach has gained significant popularity due to its ability to enhance model performance, reduce training time, and alleviate the need for large amounts of labeled data [10,11].

2.1.1. Methods and Concepts

TL methods can be categorized based on different criteria, such as the homogeneity of source and target data, the label-setting of source and target data, and the applied approaches. The most commonly applied approaches are model based, which use pretrained models and adjust them for target data by freezing, finetuning, or adding layers. Finetuning involves taking a pretrained model and adapting it to a new task by further training on a smaller dataset.

Adding new layers on top of the pretrained model's layers allow the model to perform a lighter training over the last layers (and eventually over the pretrained layers, if not frozen) to capture complex task-specific patterns that allow the model to make prediction about the input, requiring less hardware, time, and data resources [11].

2.1.2. Pretraining Techniques

Pretraining techniques are an integral part of TL. Pretraining involves training a model on a large dataset and then using the learned representations as a starting point for a target task.

There are two main challenges: catastrophic forgetting and overly biased pretrained models [11,12]. Catastrophic forgetting occurs when a pretrained model loses its previous knowledge when finetuned on a new task. Overly biased pretrained models occur when a pretrained model cannot learn new features from target data due to the frozen layers. Possible solutions include progressive learning, which adds new layers to a frozen pretrained model, and vertical expansion, which adds new nodes to the frozen pretrained layers.

2.2. Ensemble and Boosting

EL and boosting are two distinct approaches employed in ML to enhance the performance and precision of predictive models. Nevertheless, they vary in terms of their fundamental principles and methodologies. EL combines diverse models for enhanced performance, deriving the final prediction through voting or averaging. Independently created models use various algorithms, architectures, or data subsets, each contributing equally to the final prediction. They can train independently and in parallel without explicit feedback or adjustment [13].

Boosting enhances weak learners iteratively, correcting misclassifications through sequential model training. The final prediction involves combining the weak learners' outputs via weighted voting. AdaBoost, Gradient Boosting, and XGBoost are prominent algorithms that dynamically adjust sample weights to emphasize misclassified instances. Their base learner selection and weight update criteria vary. AdaBoost, notably popularized by Freund and Schapire [14], is one of the most renowned boosting algorithms [15].

Adaboost Algorithm

Freund and Schapire introduced the AdaBoost algorithm [14,16,17], revolutionizing boosting methods by utilizing weighted versions of training data, eliminating the need for an extensive dataset. AdaBoost is recognized as a widely studied technique for constructing high-performing classifier ensembles. The algorithm sequentially generates a set of classifiers through a weak learner, with re-weighted training data based on the ACC of prior classifiers. Ensuring the appropriate weak learners are selected is crucial to preventing excessive weight on outliers and noise and maintaining the effectiveness of the algorithm [15].

While AdaBoost has excelled in two-class classification tasks, its performance in multiclass problems, although adapted for such scenarios (Freund and Schapire 1997) [15], may not be as remarkable. As a result, a variation, SAMME (Stagewise Additive Modeling using a Multiclass Exponential loss function), was utilized for multiclass boosting [18].

2.3. Voting Classifier

The process of generating an ensemble often involves a choice between two fundamental voting strategies: hard and soft. These methods play a pivotal role in consolidating predictions from individual models within the ensemble, ultimately shaping the collective decision-making process.

2.3.1. Hard Voting

Hard voting predicts the final class label based on the most frequently predicted class label among the classification models [19]. In hard voting, each model in the ensemble independently predicts the class label for a given input, and the final prediction is determined by a majority vote. The class that receives the most individual votes from the models is selected as the ensemble's final prediction.

In a hard voting ensemble with three classifiers (Model A, Model B, and Model C) for a three-class classification task (Class 1, Class 2, and Class 3), let's consider a specific input:

- Model A predicts Class 1,
- Model B predicts Class 2,
- Model C predicts Class 3.

In hard voting, the final prediction is based on the majority vote. Each model gets one vote, and the class with the highest number of votes is chosen. For this input, each class receives one vote, resulting in a tie. In cases of a tie, the final prediction might remain undetermined, and additional strategies or mechanisms may be implemented for resolution, depending on the specific implementation of the hard voting ensemble. The effectiveness of hard voting often relies on having diverse models within the ensemble to capture distinct aspects of the underlying data patterns.

2.3.2. Soft Voting

On the other hand, in soft voting, the class labels are predicted by considering the predicted probabilities p from each classifier, as follows:

$$\hat{y} = argmax \sum_{j=1}^{m} w_j p_{ij}, \tag{1}$$

where w_j is the weight that can be assigned to the jth classifier.

In a multiclass classification task with class labels represented as $i \in \{0, 1, 2\}$, let us consider an example where our ensemble of classifiers makes the following prediction for a given input x:

$$C_1(x) \rightarrow [p0_1, p1_1, p2_1], C_2(x) \rightarrow [p0_2, p1_2, p2_2], C_3(x) \rightarrow [p0_3, p1_3, p2_3].$$

Assigning the weights $\{w_1,\ w_2,\ w_3\}$ to the classifiers, and computing the average probabilities would yield the predicted class label $\hat{y}$ as follows:

$$p(i_0 \mid x) = w_1 \times p0_1 + w_2 \times p0_2 + w_3 \times p0_3 \tag{2}$$

$$p(i_1 \mid x) = w_1 \times p1_1 + w_2 \times p1_2 + w_3 \times p1_3 \tag{3}$$

$$p(i_2 \mid x) = w_1 \times p2_1 + w_2 \times p2_2 + w_3 \times p2_3 \tag{4}$$

$$\hat{y} = argmax[p(i_0 \mid x), p(i_1 \mid x), p(i_2 \mid x)] \tag{5}$$

This approach is only recommended if the classifiers are well-calibrated, as it yields more reliable and accurate results.

3. Related Work

In the landscape of NLP tasks, a transformative moment came with the emergence of transformers [20], quickly becoming the standard in nearly all NLP tasks due to their remarkable efficacy. Large pre-trained models like BERT, RoBERTa, and XLNet are prominent examples of these context-dependent architectures, demonstrating their utility across various NLP tasks [21], namely in SA [22]. According to [23], utilizing a monolingual pre-trained BERT model, specifically in Portuguese, yields superior outcomes to multilingual BERT.

In addition, articles demonstrate that transformer-based models achieve superior results compared to previous models, such as the case of LSTM and CNN [24]. It is possible to adjust the hyperparameters in the training of the BERT and RoBERTa models to achieve better results. Some of these experiments have been carried out recently [25]. It is necessary to be careful with these adjustments to avoid problems such as the fading of the gradient, or the instability when finetuning is completed [26].

In ML, ensembles of classifiers are usually built by combining multiple learners (weak or strong), following the strategy of boosting rather than relying on a single strong classifier. This idea has gained interest in recent years [27], as it is often easier to train and combine several simple classifiers than to learn a complex one. There is a growing trend of utilizing different ensemble techniques and single-language pre-trained models, such as RoBERTa and BERT, rather than multi-language models to achieve new state-of-the-art results [21,22,28].

Gomes et al. [28], in 2022, employed cutting-edge transformer models to tackle two specific subtasks within the realm of aspect term extraction (ATE) and sentiment orientation extraction (SOE). They assert that they have attained the highest level of performance in both subtasks, surpassing previously established benchmarks and setting new standards for the Portuguese language.

In the case of ATE, this methodology involved the utilization of an ensemble comprising models from RoBERTa and mDeBERTa, which were trained on Portuguese and multilingual datasets, respectively, achieving 67.1% of ACC. A voting ensemble consisting of PTT5 large models was employed for the SOE subtask, without reliance on external data sources.

Regarding SOE, the optimal outcomes were realized through the utilization of PTT5 Large in conjunction with the conditional text generation training strategy, reaching 82.4% of ACC. This approach entailed presenting the complete review alongside the aspect term as an input to the model without relying on external data sources.

Lopes et al. [22] present an approach designed to extract aspects from Portuguese-language reviews using pre-trained BERT models. They conducted a performance comparison between Google's multilingual BERT and BERTimbau. Remarkably, BERTimbau attains a balanced ACC rate of up to 93% when applied to a corpus of hotel reviews. To assess the efficacy of these models in aspect extraction, the authors employ an ACC and F1 score as evaluation metrics, with their findings indicating that BERTimbau outperforms the multilingual BERT model across both metrics.

Furthermore, the authors incorporate an additional step termed post-training. This post-training phase involves enhancing BERTimbau to cater to a specific domain, aligning it with its intended field of application. When post-training is not employed, the initial results yield an F1 score of up to 70% using polarity auxiliary sentences. The introduction of post-training with polarity sentences yields enhanced performance, with the most noteworthy results reaching a 77% F1 score after 5k and 10k post-training iterations, albeit with a decrease in stability observed after the 10k steps. The model's ACC also reaches up to 80% with post-training.

Moura et al. [21] in their article, evaluates different methods for creating sentence embeddings that can be used to cluster user intents in dialog data. They compared six transformer-based models (BERT, RoBERTa, GPT-2, XLNet, ALBERT, and ELECTRA) for text representation. They also evaluated two pre-trained Siamese transformers (SBERT and SRoBERTa) and studied the impact of retraining them on domain data. Moreover, the article explores the use of ensemble methods to combine different embeddings and clustering algorithms. The article systematically assesses various approaches to generate sentence embeddings aimed at clustering user intents within dialog data. Moreover, the article delves into the exploration of ensemble methods, seeking to amalgamate diverse embeddings and clustering algorithms for enhanced results.

4. Research Problems

The central challenge we confront pertains to the current knowledge landscape, which reveals a dearth of NLP models specifically crafted for sentiment classification in Portuguese restaurant reviews. This issue bears significance as it directly influences the ability of restaurant proprietors to discern areas warranting improvement. Specifically, negative sentiments within reviews offer insight into potential shortcomings, while positive sentiments highlight the commendable aspects. Incorporating a neutral class facilitates a comprehensive assessment, pinpointing specific areas ripe for enhancement.

While conventional remedies may involve the utilization of LLM, these often come with associated costs, contingent on token usage. Alternatively, opting for open models locally may demand substantial computational power. In response to these challenges, this work advocates for a tailored solution that sidesteps the need for extensive computational resources, ultimately achieving exemplary performance in sentiment analysis.

5. Materials and Methods

5.1. Research Methodology

In this chapter, we delve into the methodology employed. The chosen methodology, presented in Figure 1, is designed to facilitate an examination of the approach undertaken.

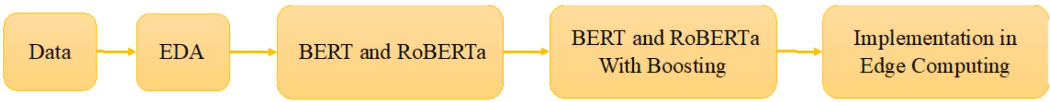


Figure 1. Representation of the methodology.

The methodology unfolds through a sequential progression, commencing with data acquisition and transitioning into Exploratory Data Analysis (EDA) to gain insight into the dataset's characteristics. Following this, individual models undergo training, honing their predictive capabilities. The next stage involves training these models with boosting techniques to enhance their overall performance. Finally, the models are implemented in an edge computing environment, ensuring practical applicability in real-world scenarios.

For those interested in exploring the open-source implementation, the code repository can be accessed via the following link: [https://github.com/Alex-Branco/RRSO_Portugal.git] (accessed on 1 January 2024).

5.2. Exploratory Analysis

The dataset for this study stemmed from a collaboration between Zomato's technical team in the Restaurant Review Sentiment Output (RRSO) project, utilizing the data provided by the online platform Zomato, stored using MongoDB. The dataset included 537,083 reviews with two columns, "text review" and "rating", post-removal of null entries. These reviews, assessing restaurants, were accompanied by customer-assigned scores collected between 1 April 2014 and 2 September 2022. Minimal preprocessing was conducted, and the reviews generally comprised 20 to 100 words, as depicted in Figure 2.

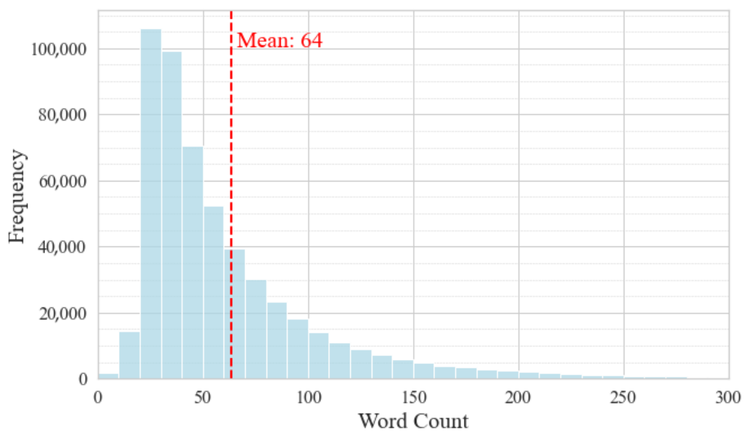


Figure 2. Word count distribution.

Considering the average of 64 words, and since the chosen models use all the punctuation, which makes the punctuation count as words, it was decided, heuristically, to make an increment of about a third, thus using a maximum length of 100 words.

5.2.1. Data Visualization and Insights

The label distribution (which corresponds to the user score provided by the user when submitting the review, from 1 to 5, where 1 is the worst and 5 is the best) is shown in Figure 3, and it is heavily imbalanced, making it challenging for supervised training.

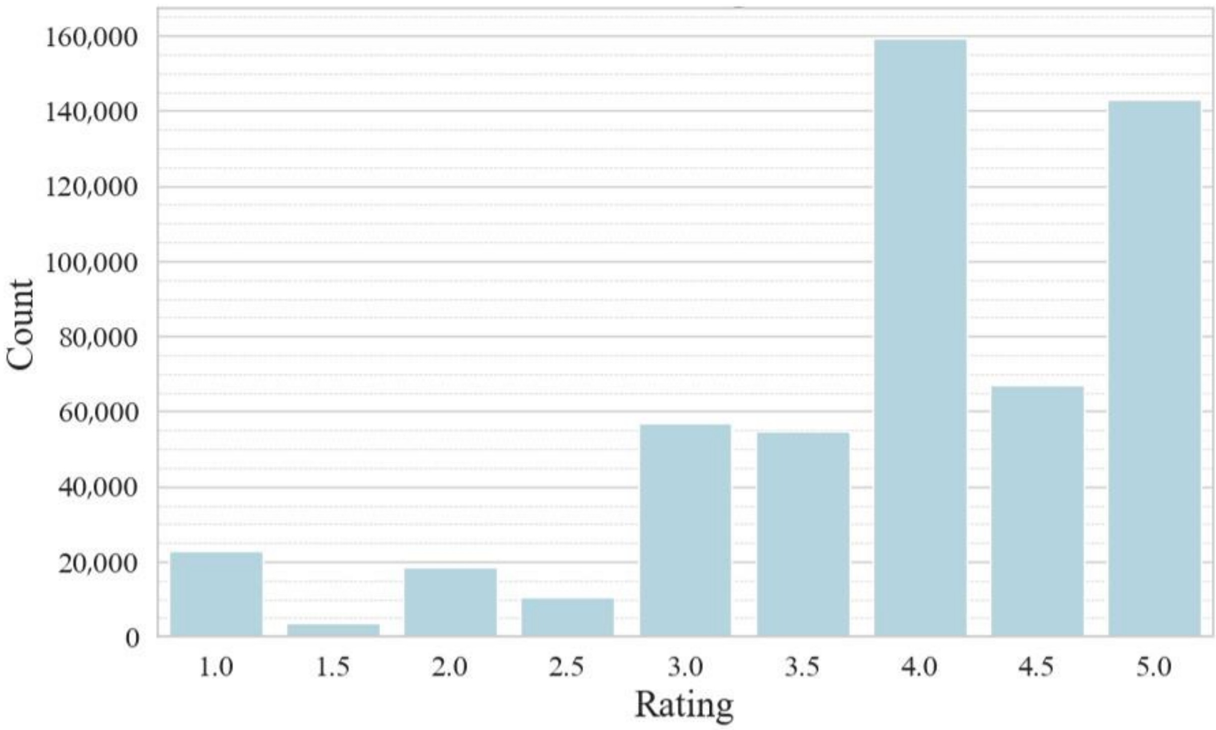


Figure 3. Raw dataset rating distribution.

The work was carried out using 3 class classification models, with the original nine label values being re-sampled to negative class (0), around 45 thousand, neutral class (1), around 120 thousand, and positive class (2), around 370 thousand. The separation was performed based on a heuristic decision [29].

5.2.2. Preprocessing

There was a need to handle an imbalanced dataset because the data were not equally distributed. This can cause a bias towards the more represented classes in the model. The inverse class frequency method is applied to balance the data, assigning weights to each class, with higher weights given to underrepresented classes and lower weights given to more represented classes. This increases the significance of underrepresented classes during the training process, leading to a more accurate capture of patterns in the data.

Figure 4 shows the representation of the training dataset and the weights assigned to each class.

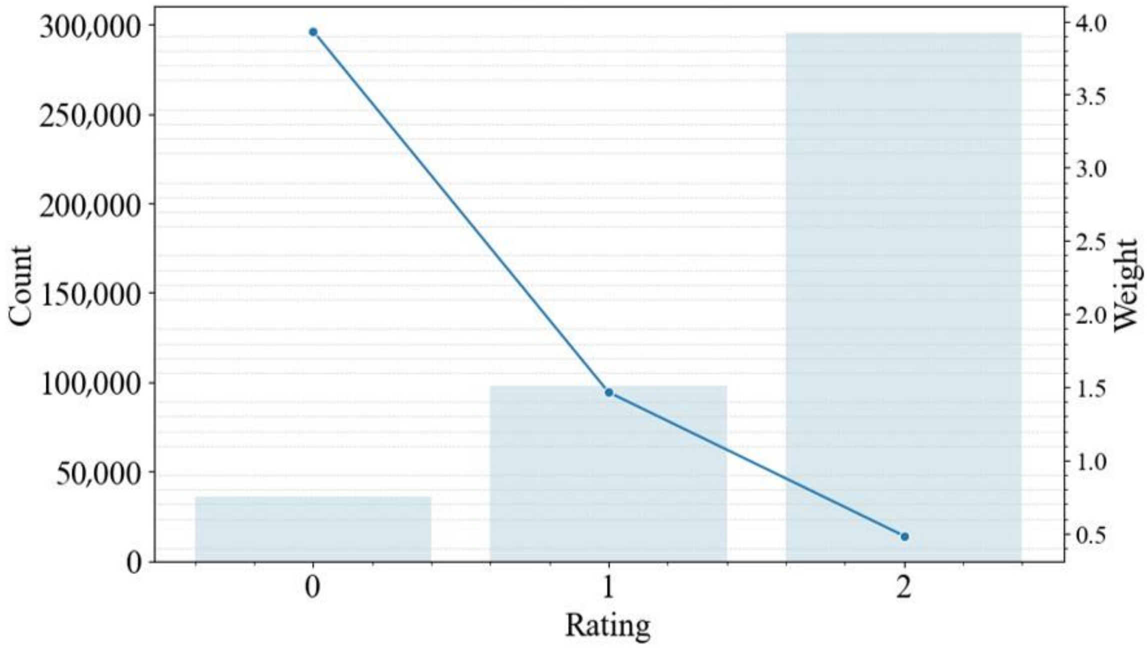


Figure 4. Train dataset weight rating distribution and weight assigned.

5.3. Training Parametrization

Leveraging a Graphics Processing Unit (GPU) enabled the use of larger batches, expediting the training process. The adoption of PyTorch-based models, notably BERTimbau, prompted the code to use PyTorch and Python [30]. Specifically, the used libraries were "torch 1.13", "numpy 1.23.5", and "pandas 1.5.3". Based on insights from previous chapters and scientific discourse, the following hyperparameters were established: max_len = 100; batch_size = 128; epochs = 20; patience = 10; delta = 0.0003; dropout = 0.2; learning_rate = 2×10^{-5}.

Data separation prioritizes training, with 80% for training, 14% for validation, and 6% for testing. This division emphasizes greater attention to the training process, particularly since the pre-trained model requires finetuning the output layers. A scheduler function controls the learning rate, starting at 0 and gradually increasing during the warmup phase before following a linear schedule. The warmup phase covers 10% of the total steps, preventing suboptimal solutions. To address domain-specific biases in the NLP models caused by a limited access to generalized datasets, 2-fold cross-validation is used for comprehensive assessment and validation of the SA model's generalization capabilities.

5.4. Sentiment Analysis Using BERTimbau

BERTimbau, developed by NeuralMind [30], is a Portuguese-tailored language model based on the BERT architecture. Available in Base (12 layers) and Large (24 layers) sizes, it is compatible with TensorFlow 2 and PyTorch 2, catering to various computational requirements. Table 1 details insights about these two models and the RoBERTa base, showing that the large model and RoBERTa are almost double the size of the base model.

Table 1. Parameters of the examined BERT and RoBERTa models.

	$\mathbf{BERT_{BASE}}$	$\mathbf{BERT_{LARGE}}$	$\mathbf{RoBERTa_{BASE}}$
Layers	12	24	24
Hidden Size	768	1024	1024
Heads	12	16	16
Parameters	110 M	335 M	355 M

BERTimbau was used, since it was shown to be capable of exceling in standard NLP tasks [23], and its adaptability through finetuning with smaller datasets allows it to meet domain-specific requirements and enhance task performance.

In this study, the model's adaptability to specific domains was leveraged for sentiment classification with three classes, utilizing the Portuguese learned during pretraining. Notably, two additional layers were meticulously engineered during the finetuning process, providing a unique contribution to this research. A regularization layer was used to enhance the model's robustness, followed by a fully connected dense layer to bolster its capacity for learning intricate patterns. These customized layers, not adopted from existing models, were developed for this research. The architecture is shown in Figure 5. Finally, a Softmax function was incorporated for prediction computation.

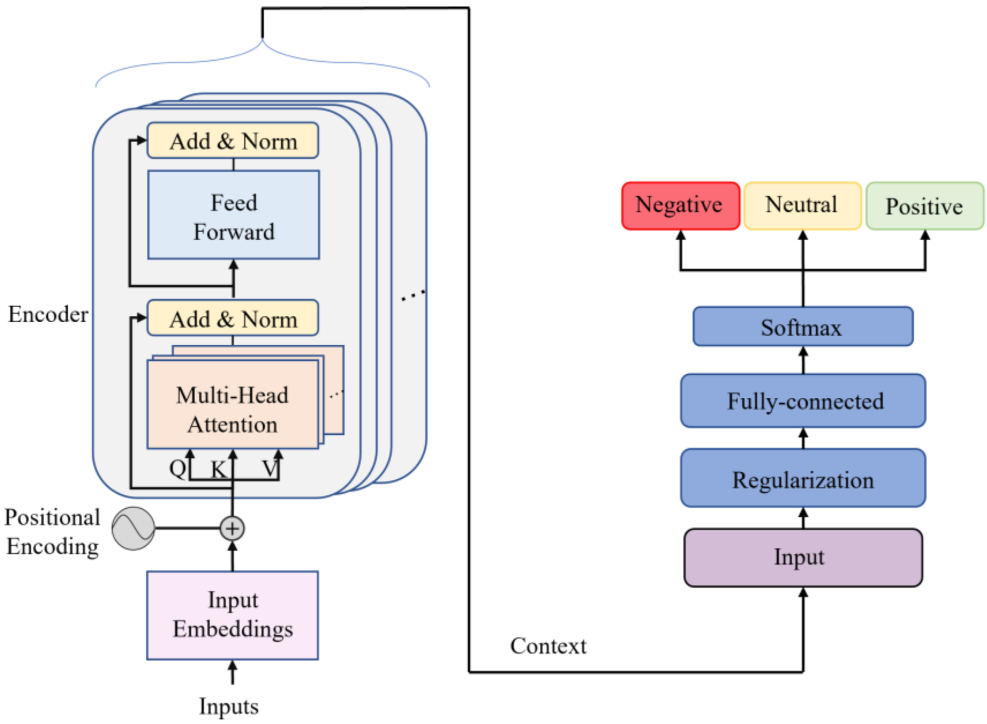


Figure 5. Representation of a fully developed model.

The pretrained model was loaded into the SentimentClassifier along with the tokenizer. Following this, the dataset was split, and a data loader object was created to manage the appropriately sized and allocated samples during training. Both the pre-trained model and tokenizer are part of the transformer's library.

5.5. Sentiment Analysis Using RoBERTa

RoBERTa, a variant of BERT, involves adjustments to hyperparameters and embeddings. Its architecture mirrors that of the original BERT model. We incorporated the same layers used in the BERTimbau model, including the regularization layer and fully connected layer, to build the sentiment classifier block as shown in Figure 6. A Softmax function was then applied to the output, as depicted in Figure 5.

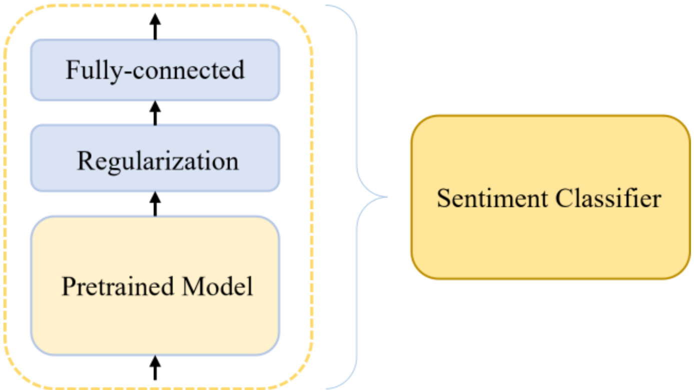


Figure 6. Architecture developed for SA.

This RoBERTa model is available in the Hugging Face model repository by team thegoodfellas [31] and is a finetuned version of the xlm-roberta-base on the BrWac dataset. The base parameters that RoBERTa uses are presented in Table 1, as in the original paper [2].

5.6. Boosting Sentiment Classifier Models

The sentiment classifier, represented in Figure 6, consists of a pre-trained model and two preceding layers before its output. Whether using BERTimbau or RoBERTa, the pre-trained model seamlessly integrates into the sentiment classifier. Figure 7 showcases the ensemble architecture, considering its adaptability with either pre-trained model, ensuring easy integration and versatility for future applications.

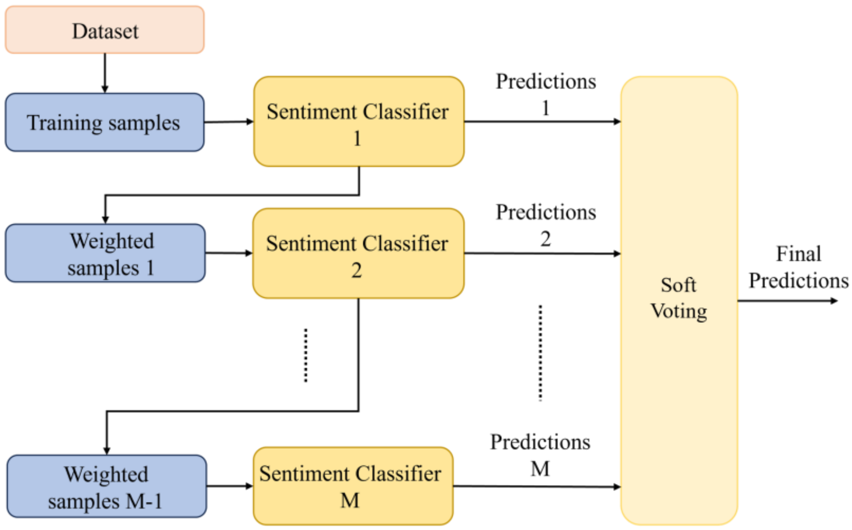


Figure 7. Architecture of the ensemble applied to each sentiment classifier.

5.7. Implemented Model

The framework below illustrates the review analysis process and the model's role, depicting how textual content is transformed into numerical data using robust computational techniques. Figure 8 enhances understanding, showcasing the model's structure and the transformation of textual content into a computationally significant format.

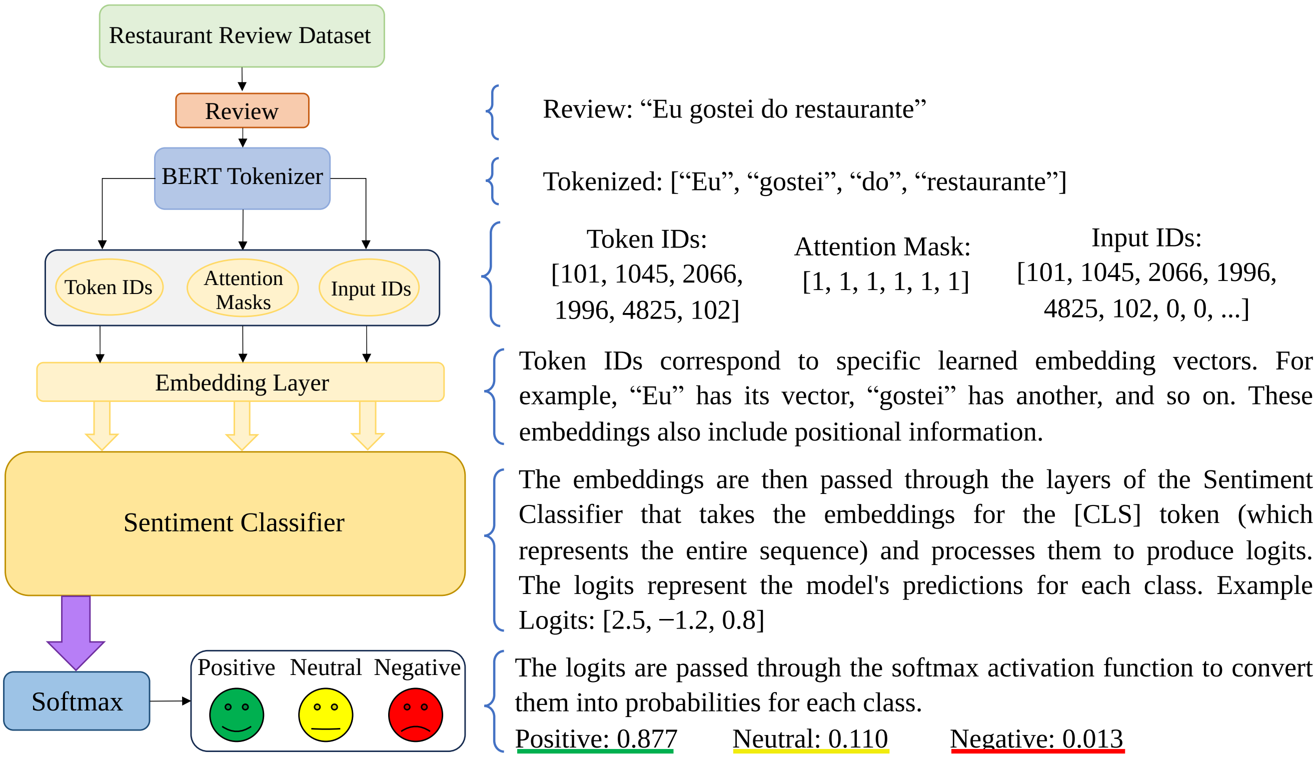


Figure 8. Full model architecture developed.

These probabilities reflect the model's confidence in the input sentence belonging to each sentiment class. In this example, the highest probability is for the "positive" class, suggesting that the model predicts a positive sentiment for the input sentence "Eu gostei do restaurante".

5.8. Performance Analysis

This study evaluated the performance through hold-out validation for its simplicity and computational efficiency and employed stratified two-fold cross-validation to assess the result variability. The examined performance metrics were AUC, ACC, F1-Score, Sensitivity, Specificity, and Precision, computed for the multiclass problem during the examination, utilizing macroaveraging to aggregate results. Furthermore, categorical cross-entropy was used as the loss function during the models' training.

5.9. Hardware Implementation

This sub-section explores sentiment classification on devices suitable for edge computing, Jetson Nano and Raspberry Pi, against the RTX 3090 graphics card (used as a benchmark in this context). SA involves determining text sentiment, enabling real-time decision-making, and enhancing privacy by processing sensitive data locally, reducing dependence on cloud services.

The comparison between the Raspberry Pi 4 and the Jetson Nano reveals distinctive attributes. While the Raspberry Pi 4 stands out for its cost-effectiveness, built-in Wi-Fi, and Bluetooth, it lags behind the Jetson Nano in terms of memory performance and the utilization of its 64-bit hardware architecture with a 32-bit operating system. On the other hand, the Jetson Nano shines with its powerful GPU, making it an optimal choice for beginners in ML applications. However, for projects that do not demand intensive ML models, the Raspberry Pi 4 offers adequate power at a lower cost.

The models were preloaded onto their platforms to ensure smooth operation and prevent potential internet-related delays. Thus, this approach is aligned with the goal of utilizing edge computing, avoiding costly cloud computing resources, despite occasional space constraints.

6. Results and Discussion

This section presents the main results derived from the proposed solution. Following the presentation, a discussion will be provided to delve deeper into the implications and findings of the results.

The analysis encompasses a detailed examination of each classifier's performance. Initially, a standard English-based BERT model is used to compare against the performance of a Portuguese-based BERT model, benchmarking the performance. Then, boosting is examined for the Portuguese-based BERT model. Afterward, the experiment is performed on RoBERTa, examining the performance with and without boosting. A discussion is carried out with a comparative analysis with state-of-the-art works. Lastly the result of the edge computing analysis were examined.

6.1. BERT Models

The initial results established a baseline, using BERT pretrained for English-spam classification. Then, the dataset was translated to English, reaching an ACC of 70%. For the initial training using the BERTimbau model and 60,000 balanced samples, heuristic hyperparameters were selected including: maximum input length of 100; batch size of 128; maximum number of epochs of 200; patience for early stopping of 10; minimum improvement for early stopping of 0.005. These parameters were aligned with the default values for the problem.

The test (referenced as BaseModel), utilizing a balanced dataset sample, demonstrated BERTimbau's promise, achieving 0.77 ACC. Referencing the base model, the finetuning process involved adjusting the scheduler and optimizer, applying inverse frequency balancing to a dataset comprising 60,000 samples. Various learning rates (3×10^{-9}, 3×10^{-6}, 2×10^{-5}, and 1×10^{-4}) were tested alongside two optimizers, AdamW and AdaGrad.

The tests identified the best-performing model when using a learning rate of 2×10^{-5} with AdaGrad. Based on these initial assessments, an initial warmup of 10% of the total steps was applied to the scheduler, leveraging the model's optimal performance in the initial training epochs. This way, now using the complete dataset, it was possible to reach 80% ACC, and examining the ACC plot, shown in Figure 9a, it is evident that the best epoch was twenty due to the monitorization of AUC, presented in Figure 9b; although, it is visible that from epoch eight the model begins to have difficulty evolving more. These results are referenced as SentAnalysisPt. In the figures, the dashed line indicates the epoch with the highest validation AUC (thus selected by early stopping as the optimal epoch).

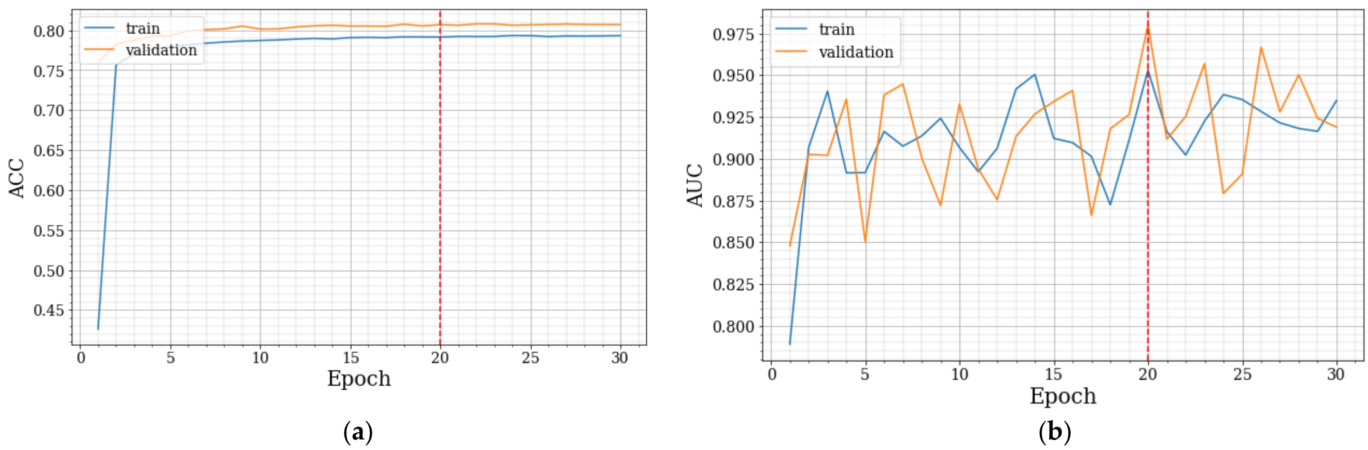
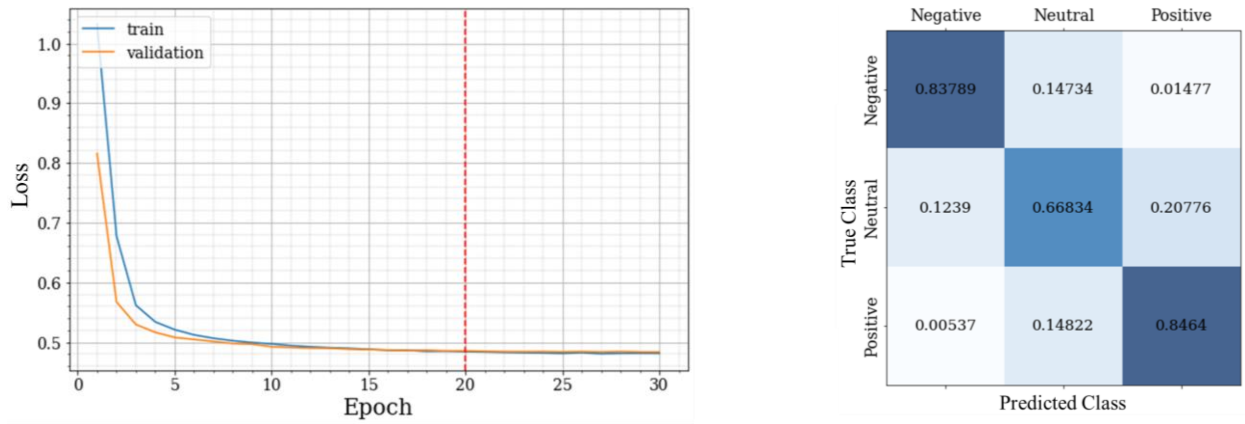


Figure 9. Train history of the performance metrics ACC (**a**), and AUC (**b**), using SentAnalysisPt.

As for the graph that represents the losses, presented in Figure 10a, the model did not exhibit erroneous behavior, unlike the base model, and from epoch 14, the loss stabilized. The CM in Figure 10b has a better ACC than the base model due to a more balanced class distribution, ensuring the model performs better across all classes.



(**a**) (**b**)

Figure 10. Train history of the loss (**a**), and the resulting CM (**b**), using SentAnalysisPt.

Table 2 presents the performance of the BERT models and indicates that the ACC improved to 81% from approximately 78% by using the improved configuration, despite a slight precision decrease. Furthermore, the large BERTimbau model was also examined, denoted as SentAnalysisPtBertLarge, only presenting a residual improvement, justifying discarding the larger model due to its higher computational demands, aligning with the objective of implementing it on edge computers.

Table 2. Average metrics of the different models developed using BERT using hold-out validation.

Model	AUC	ACC	F1-Score	Sensitivity	Specificity	Precision
BaseModel	0.892	0.777	0.778	0.778	0.888	0.780
SentAnalysisPt	0.923	0.805	0.752	0.784	0.887	0.727
SentAnalysisPtBertLarge	0.923	0.807	0.759	0.790	0.889	0.735

The class-based metrics, presented in Figure 11, demonstrate a minor decrease in the positive class ACC but an enhancement in the negative and neutral classes, ensuring a more balanced model classification.

When using two-fold cross-validation in the SentAnalysisPt, the average ACC was 0.820 (with a standard deviation of 0.009), while AUC was 0.892 (with a standard deviation of 0.003). Furthermore, the per-class ACC of the model was 0.955, 0.826, and 0.858 for the negative, neutral, and positive classes, respectively. These results indicate the suitability of the proposed models.

Experiments with ensembles ranging from two to six classifiers were conducted, and the results are shown in Table 3 using the SentAnalysisPt model as weak learner. The focus was on identifying the optimal number of classifiers while accounting for the model's size expansion, concluding that when using more than two classifiers, the performance starts to degrade. Such is likely due to the weak learner used, indicating that it is a quite strong learner, as expected.

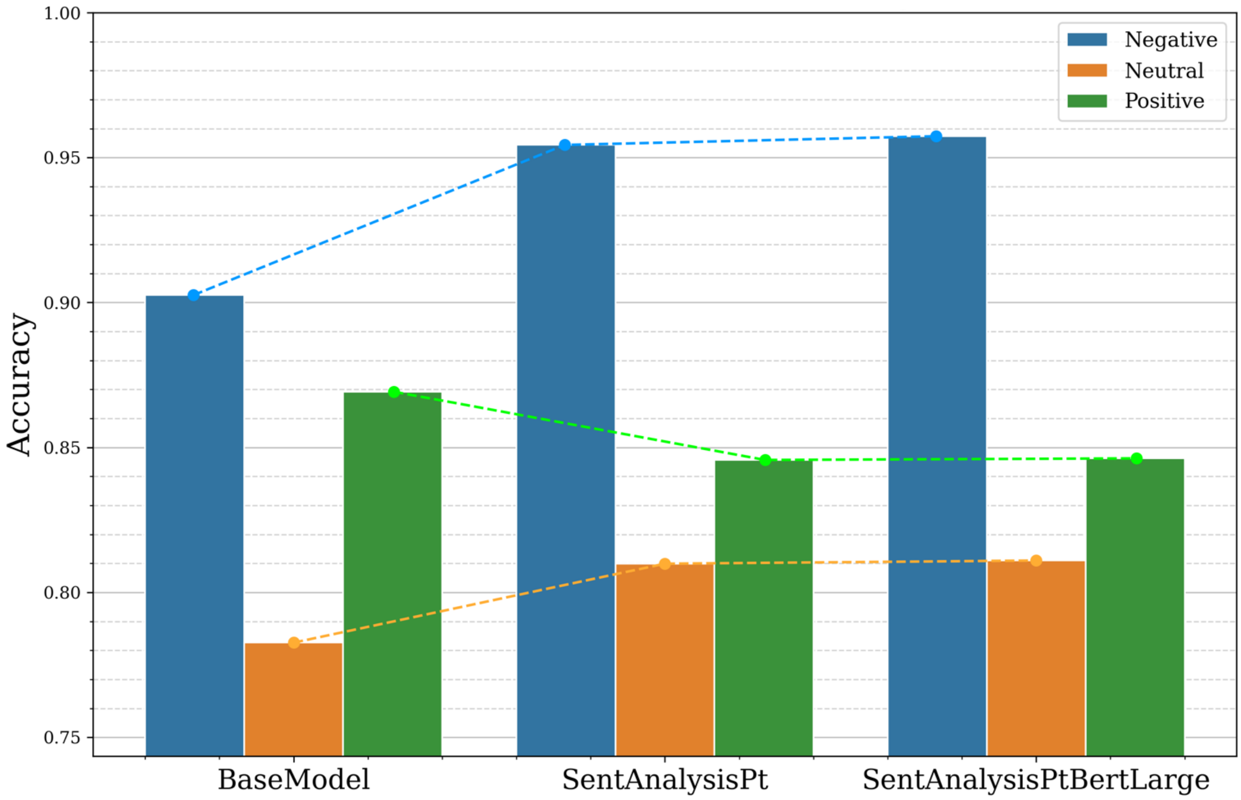


Figure 11. Comparison of the ACC per class of the three models.

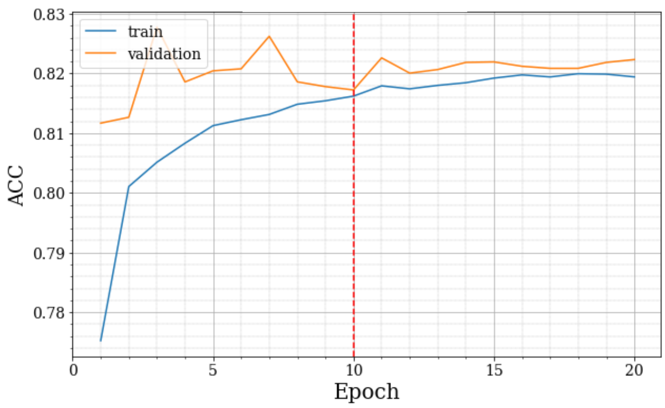
Table 3. Average performance metrics for different approaches using ensemble and hold-out validation. The number at the end of the approach refers to the number of weak learners used.

Approach	ACC	F1-Score	Sensitivity	Specificity	Precision
SentAnalyPtAdaBoost_2	**0.840**	**0.777**	**0.765**	**0.885**	0.790
SentAnalyPtAdaBoost_3	0.840	0.770	0.751	0.878	0.794
SentAnalyPtAdaBoost_4	0.838	0.759	0.733	0.869	0.797
SentAnalyPtAdaBoost_5	0.840	0.764	0.745	0.873	0.793
SentAnalyPtAdaBoost_6	0.838	0.758	0.728	0.869	0.801

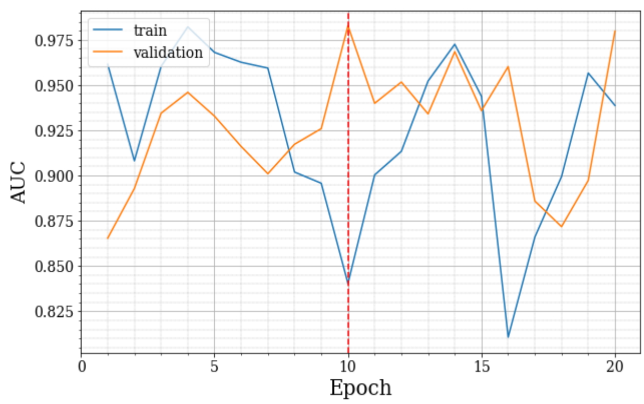
6.2. RoBERTa Models

To perform TL based on the pre-trained RoBERTa model with the complete dataset, it was necessary to change the hyperparameters. These included the batch_size, which was reduced from 128 to 64, to decrease the computational demand for training, and an increase in the learning rate to 1×10^{-4} instead of 2×10^{-5}. This model was denoted as SentAnalysisPtRoBERTa.

Figure 12a shows that there were epochs with a validation ACC higher than the best epoch. That occurrence is often due to the complex and dynamic nature of the training process. Looking at Figure 12b, which refers to the metric monitored (AUC) to have a model with a performance more balanced among the three classes, it is evident that epoch 10 is the one that achieves the highest value, providing a more balanced validation. This conclusion is further supported when examining the loss presented in Figure 13a.

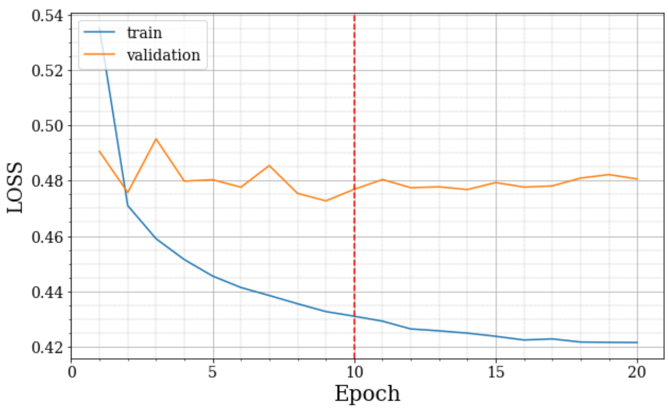


(**a**)

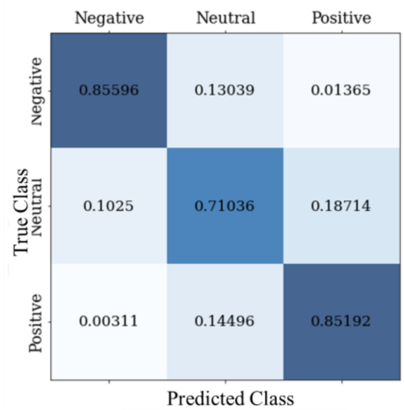


(**b**)

Figure 12. Train history of the performance metrics ACC (**a**), and AUC (**b**), using SentAnalysisPtRoBERTa.



(**a**)



(**b**)

Figure 13. Train history of the loss (**a**), and the resulting CM (**b**), using SentAnalysisPtRoBERTa.

The CM shown in Figure 13b presents a good balance between classes, although the intermediate class (neutral) always ended up being disadvantaged. This is one of the characteristics of intermediate classes, as they are consistently subject to a higher classification error and are likely more subjective in the user reviews.

Using hold-out validation, SentAnalysisPtRoBERTa attained an average AUC, ACC, F1-Score, Sensitivity, Specificity, and Precision of 0.933, 0.820, 0.777, 0.806, 0.897, and 0.756, respectively. These results highlight the advantages of employing the RoBERTa model for SA, emphasizing its ability to yield enhanced overall performance, surpassing the SentAnalysisP model. Furthermore, when using boosting subjected to two-fold cross-validation, the average ACC was 0.829 (with a standard deviation of 0.013), while AUC was 0.897 (with a standard deviation of 0.004). When compared to SentAnalyPtAdaBoost_2, it is notorious that using two-fold cross-validation had a lower impact in the SentAnalyPtAdaBoost-RoBERTa_2 model. Furthermore, when using two-fold cross-validation, the average ACC was 0.829 (with a standard deviation of 0.013), while AUC was 0.897 (with a standard deviation of 0.004). When compared to SentAnalysisPt, it is notorious that both models attained a similar performance, but SentAnalysisPtRoBERTa was superior. These results are further supported by the per-class ACC of the SentAnalysisPtRoBERTa at 0.955, 0.834, and 0.867 for the negative, neutral, and positive classes, respectively.

Similarly, to the previous model using BERTimbau, boosting models were developed using RoBERTa, and the same conclusion was reached; that it is preferable to use two classifiers in the ensemble, attaining a performance, using hold-out validation, of 0.828, 0.765, 0.757, 0.880, and 0.773 for the average ACC, F1-Score, Sensitivity, Specificity, and Precision, respectively. This boosted model was named SentAnalyPtAdaBoostRoBERTa_2.

The tests in this chapter confirm that the RoBERTa architecture yields better results individually compared to the standard BERT-based architecture in terms of ACC. But when using boosting, the SentAnalyPtAdaBoost_2 was superior. Furthermore, this study also aims to implement an efficient model on edge computing platforms through an ensemble approach, which reveals the limitation of the RoBERTa architecture due to concerns about the final model's size.

6.3. Comparative Analysis

Regarding the AUC metric, the analysis focussed on SentAnalysisPt and SentAnalysisPtRoberta models. Both exhibit AUC values exceeding 0.92, indicating exceptional performance. High ACC is generally desirable, but it may not be the only metric to consider, especially for imbalanced datasets. All models have relatively high ACC values, with booting-based ones attaining the highest values. However, ACC alone might not be sufficient for model selection. In regard to the F1-Score, all models attained a good performance, signifying a commendable equilibrium between precision and sensitivity.

Based on the studied metrics, SentAnalyPtAdaBoost_2 was found to be the best model. Table 4 provides a comparative analysis with well-known state-of-the-art works performing SA in comparison to this work. Despite utilizing different databases, this comparative analysis allows for an initial examination, revealing that the conducted work has achieved significantly superior performance. This outcome further underscores the significance of the work carried out.

Table 4. Comparative analysis with state-of-the-art works that performed SA with transformer-based models.

Model Name	Language	ACC	F1	Sensitivity	Specificity
NB3 [32]	Portuguese	0.65	0.60	-	-
Self-AT_LSTM [33]	English	0.67	0.68	0.68	-
SVM [29]	Portuguese	0.81	0.83	-	-
LogReg3 [34]	English	0.77	0.76	0.77	-
PTT5 [28]	Portuguese	0.82	0.82	0.82	
BERTimbau [22]	Portuguese	0.80	0.77	-	0.78
BERT [21]	English	0.79	-	-	-
RoBERTa [21]	English	0.78	-	-	-
This work (SentAnalyPtAdaBoost_2)	Portuguese	0.84	0.78	0.77	0.79

It is noteworthy that upon comparing the ensemble models outlined in [21] with those developed within the scope of this work, a remarkable similarity emerges. Specifically, when employing an ensemble with the RoBERTa model, there appears to be a consistent decrease in ACC compared to the BERT model. This observation may suggest a potential challenge in effectively forming an ensemble with this model.

6.4. Inference on Edge

The edge computing analysis compared the performance of the developed models considering the inference two edge platform, Jetson Nano and Raspberry Pi, against the benchmark GPU, the RTX 3090 platform, simulating a cloud computing system and providing insights into the feasibility of implementing complex models on edge computing platforms. Table 5 presents the average elapsed times for classifying the three classes in a total of nine reviews of varying sizes.

Table 5. Average inference time per review of developed models.

Model		Average Platform Time (Seconds)		
Architecture	Parameters	Jetson Nano	Raspberry Pi	RTX 3090
SentAnalysisPt	110 M	2.720	4.193	2.107
SentAnalyPtAdaBoost_2	220 M	74.070	15.593	2.390
SentAnalysisPtBertLarge	335 M	40.427	15.393	0.597
SentAnalysisPtRoBERTa	355 M	4.033	4.330	2.107
SentAnalyPtAdaBoostRoBERTa_2	710 M	1295.867	188.567	3.683

Based on Table 5, a comprehensive analysis of computational efficiency across various models and platforms, exploring the model complexity, comparing hardware platforms, examining performance relative to model intricacy, assessing the impact of hardware, and considering energy efficiency is conducted.

SentAnalysisPt and SentAnalysisPtRoBERTa exhibit relatively lower parameters, while SentAnalyPtAdaBoost_2 and SentAnalyPtAdaBoostRoBERTa_2 are notably parameter rich. Significantly divergent computational performances emerge when scrutinizing the platforms. The RTX 3090 stands out as the most powerful, followed by the Jetson Nano and Raspberry Pi.

A closer look at the models' average times per platform, shown in Figure 14, reveals that computational efficiency extends beyond mere model complexity. Surprisingly, SentAnalysisPtBertLarge, with fewer parameters than SentAnalyPtAdaBoost_2 and SentAnalyPtAdaBoostRoBERTa_2, demonstrates notably swifter performance. This highlights the pivotal role of model optimization and design in computational efficiency.

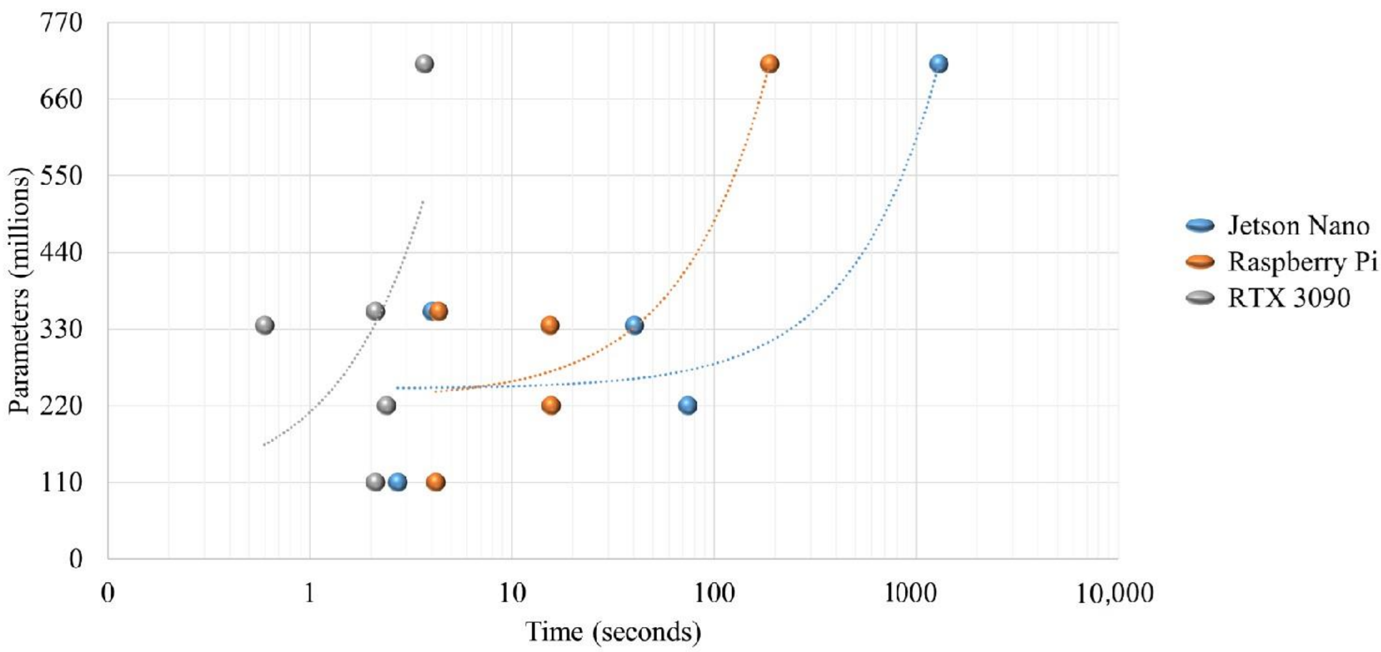


Figure 14. Performance to complexity ratio in terms of parameters and time of the examined hardware platforms. The dash lines indicate the interpolation of the points.

The influence of hardware on computational efficiency is obvious. The RTX 3090 consistently outperforms the Jetson Nano and Raspberry Pi in several orders of magnitude for the larger models. The Jetson Nano and Raspberry Pi, while less powerful in computational terms, manage well with smaller models; however, Raspberry Pi substantially outperformed the Jetson Nano in the larger models, due to RAM memory availability.

Choosing the appropriate model and platform hinges on the specific demands of the task at hand. In resource-limited environments, where computational power and energy efficiency are paramount, opting for SentAnalysisPt on the Jetson Nano or Raspberry Pi may prove advantageous. In scenarios like cloud computing, where computational prowess is paramount, the RTX 3090 paired with a larger model, such as SentAnalyPtAdaBoost_2, is likely the best option. This analysis is clearly highlighted in Figure 14, where the performance to complexity ratio shows a clear trendline for the hardware platforms. It becomes evident that, on a global scale, as the complexity of the developed model and architecture increases, in this case, the application of an ensemble system with boosting and weighted voting, the platforms face significant challenges.

7. Conclusions

The surge of NLP in sentiment classification has driven substantial progress, particularly in tackling the scarcity of sentiment classification models in Portuguese. This study navigated SA complexities using TL with transformer models, initially trained diversely and then finetuned for review language nuances. Our SA analysis of Portuguese restaurant reviews, deploying BERT and RoBERTa on edge devices, showed commendable speed–accuracy balance for real-time use, achieving an impressive 0.84 accuracy, surpassing state-of-the-art models such as PTT5 (0.82) and BERTimbau (0.8) (Table 4). The model's performance in terms of F1, sensitivity, and specificity further solidifies its effectiveness, outperforming existing models in various aspects (Table 4).

While boosting-based models improved performance, model size considerations arose, impacting edge computing feasibility. Authentic restaurant reviews added complexity, addressing diverse styles and irony. This work achieved its primary goal, providing robust SA models for the Portuguese language, applicable on edge devices, and expanding to real-world SA applications. Additionally, the average inference time per review for our developed models, including SentAnalyPtAdaBoost_2, offer valuable insights for real-time applications on platforms such as Jetson Nano and Raspberry Pi.

This study acknowledges the limitations in data size, language specificity, and device variability, prompting opportunities for future enhancement. Access to larger, diverse datasets could boost model generalization. Future work explores neural architecture search for model architecture optimization, with the potential to achieve even higher accuracies. Acknowledging our challenges, we look to expand into restaurant brand monitoring and business decision making. Continuous exploration and refinement are vital. This work can also be replicated for other languages by using the provided source code.

Author Contributions: Conceptualization, A.B., D.P., M.S., F.M., S.S.M. and F.M.-D.; methodology, A.B., D.P., F.M., S.S.M. and F.M.-D.; software, A.B. and D.P.; validation, D.P., M.S., F.M., S.S.M. and F.M.-D.; formal analysis, A.B., D.P., M.S., F.M. and S.S.M.; investigation, A.B., D.P., M.S., F.M., S.S.M. and F.M.-D.; resources, F.M., S.S.M. and F.M.-D.; data curation, A.B., D.P. and M.S.; writing—original draft preparation, A.B.; writing—review and editing, D.P., M.S., F.M., S.S.M. and F.M.-D.; visualization, A.B.; supervision, F.M., S.S.M. and F.M.-D.; project administration, F.M.-D.; funding acquisition, F.M., S.S.M. and F.M.-D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by ARDITI—Agência Regional para o Desenvolvimento da Investigação, Tecnologia e Inovação under the scope of the project M1420-01-0247—RRSO—Restaurant Review Sentiment Output, co-financed by the Madeira FEDER-000055 Program—European Social Fund. It was also funded by LARSyS (Project—UIDB/50009/2020, DOI: 10.54499/UIDB/50009/2020 (https://doi.org/10.54499/UIDB/50009/2020)), Portuguese Foundation for Science and Technology (FCT) for support through Projeto Estratégico LA 9—UIDB/50009/2020, ARDITI—Agência Regional para o Desenvolvimento da Investigação, Tecnologia e Inovação under the scope of the project M1420-09-5369-FSE-000002—Post-Doctoral Fellowship, co-financed by the Madeira 14–20 Program—European Social Fund.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4171–4186.
2. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
3. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models Are Few-Shot Learners. In Proceedings of the Advances in Neural Information Processing Systems; Curran Associates, Inc.: Red Hook, NY, USA, 2020; Volume 33, pp. 1877–1901.
4. What Is Deep Learning? | How It Works, Techniques & Applications. Available online: https://www.mathworks.com/discovery/deep-learning.html (accessed on 9 June 2023).
5. Mathew, A.; Arul, A.; Sivakumari, S. Deep Learning Techniques: An Overview. In *Advanced Machine Learning Technologies and Applications*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 599–608. ISBN 9789811533822.
6. Adnan, M.; Sarno, R.; Sungkono, K.R. Sentiment Analysis of Restaurant Review with Classification Approach in the Decision Tree-J48 Algorithm. In Proceedings of the 2019 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia, 21–22 September 2019; pp. 121–126.
7. Zahoor, K.; Bawany, N.Z.; Hamid, S. Sentiment Analysis and Classification of Restaurant Reviews Using Machine Learning. In Proceedings of the 2020 21st International Arab Conference on Information Technology (ACIT), Giza, Egypt, 28–30 November 2020; pp. 1–6.
8. Patil, D.R.; Shukla, D.; Kumar, A.; Rajanak, Y.; Pratap Singh, D.Y. Machine Learning for Sentiment Analysis and Classification of Restaurant Reviews. In Proceedings of the 2022 3rd International Conference on Computing, Analytics and Networks (ICAN), Punjab, India, 18–19 November 2022; pp. 1–5.
9. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. *Comput. Sci.* **2018**. *preprint*.
10. Liu, Z. Effective Transfer Learning for Low-Resource Natural Language Understanding. *arXiv* **2022**, arXiv:2208.09180.
11. Iman, M.; Arabnia, H.R.; Rasheed, K. A Review of Deep Transfer Learning and Recent Advancements. *Technologies* **2023**, *11*, 40. [CrossRef]
12. Wong, W.; Koh, Y.S.; Dobbie, G. Using Flexible Memories to Reduce Catastrophic Forgetting. In *Advances in Knowledge Discovery and Data Mining*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 219–230. ISBN 978-3-031-33376-7.
13. Dietterich, T.G. Ensemble Methods in Machine Learning. In Proceedings of the Multiple Classifier Systems; Springer: Berlin/Heidelberg, Germany, 2000; pp. 1–15.
14. Freund, Y.; Schapire, R.E. Experiments with a New Boosting Algorithm. *Int. Conf. Mach. Learn.* **2001**. [CrossRef]
15. Ferreira, A.J.; Figueiredo, M.A.T. Boosting Algorithms: A Review of Methods, Theory, and Applications. In *Ensemble Machine Learning*; Zhang, C., Ma, Y., Eds.; Springer: Boston, MA, USA, 2012; pp. 35–85. ISBN 978-1-4419-9325-0.
16. Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
17. Freund, Y.; Schapire, R.E. Game Theory, on-Line Prediction and Boosting. In Proceedings of the Ninth Annual Conference on Computational Learning Theory—COLT '96, Desenzano del Garda, Italy, 28 June–1 July 1996; ACM Press: New York, NY, USA, 1996; pp. 325–332.
18. Zhu, J.; Rosset, S.; Zou, H.; Hastie, T. Multi-Class AdaBoost. *Stat. Interface* **2006**, *2*, 349–360. [CrossRef]
19. EnsembleVoteClassifier: A Majority Voting Classifier—Mlxtend. Available online: http://rasbt.github.io/mlxtend/user_guide/classifier/EnsembleVoteClassifier/ (accessed on 5 June 2023).
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In *Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates, Inc.: West Chester, OH, USA, 2017; Volume 30.
21. Moura, A.; Lima, P.; Mendonça, F.; Mostafa, S.S.; Morgado-Dias, F. On the Use of Transformer-Based Models for Intent Detection Using Clustering Algorithms. *Appl. Sci.* **2023**, *13*, 5178. [CrossRef]
22. Lopes, É.P.; Freitas, L.; Gomes, G.; Lemos, G.; Hammes, L.O.; Corrêa, U.B. Exploring BERT for Aspect-Based Sentiment Analysis in Portuguese Language. *Int. FLAIRS Conf. Proc.* **2022**, *35*. [CrossRef]
23. Souza, F.; Nogueira, R.; Lotufo, R. BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In *Proceedings of the Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Part I, Rio Grande, Brazil, 20–23 October 2020*; Cerri, R., Prati, R.C., Eds.; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12319, pp. 403–417.
24. Souza, F.D.; de Oliveira e Souza, F.J.B. Embedding Generation for Text Classification of Brazilian Portuguese User Reviews: From Bag-of-Words to Transformers. *Neural Comput. Appl.* **2022**, *35*, 9393–9406. [CrossRef]
25. Izsak, P.; Berchansky, M.; Levy, O. How to Train BERT with an Academic Budget. *arXiv* **2021**, arXiv:2104.07705.

26. Mosbach, M.; Andriushchenko, M.; Klakow, D. On the Stability of Fine-Tuning BERT: Misconceptions, Explanations, and Strong Baselines. *arXiv* **2021**, arXiv:2006.04884.
27. Kuncheva, L.I. *Combining Pattern Classifiers: Methods and Algorithms*; John Wiley & Sons: Hoboken, NJ, USA, 2014; ISBN 978-1-118-31523-1.
28. Gomes, J.R.S.; Garcia, E.A.S.; Junior, A.F.B.; Rodrigues, R.C.; Silva, D.F.C.; Maia, D.F.; Filho, A.R.G. Deep Learning Brasil at ABSAPT 2022: Portuguese Transformer Ensemble Approaches. *CEUR-WS* **2022**.
29. dos Santos, F.L.; Ladeira, M. The Role of Text Pre-Processing in Opinion Mining on a Social Media Language Dataset. In *Proceedings of the 2014 Brazilian Conference on Intelligent Systems, Sao Paulo, Brazil, 19–23 October 2014*; IEEE: New York, NY, USA, 2014; pp. 50–54.
30. Neuralmind (NeuralMind Inteligência Artificial). Available online: https://huggingface.co/neuralmind (accessed on 24 July 2023).
31. Thegoodfellas/Tgf-Xlm-Roberta-Base-Pt-Br · Hugging Face. Available online: https://huggingface.co/thegoodfellas/tgf-xlm-roberta-base-pt-br (accessed on 25 July 2023).
32. Brum, H.B.; das Gracas Volpe Nunes, M. Building a Sentiment Corpus of Tweets in Brazilian Portuguese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 December 2017*; European Language Resources Association (ELRA): Paris, France, 2017; pp. 4167–4172.
33. Xia, H.; Ding, C.; Liu, Y. Sentiment Analysis Model Based on Self-Attention and Character-Level Embedding. *IEEE Access* **2020**, *8*, 184614–184620. [CrossRef]
34. Tusar, M.T.H.K.; Islam, M.T. A Comparative Study of Sentiment Analysis Using NLP and Different Machine Learning Techniques on US Airline Twitter Data. *arXiv* **2021**, arXiv:2110.00859.