Journal of Systemics, Cybernetics and Informatics (2022) 20(5), 33-59
https://doi.org/10.54808/JSCI.20.05.33

Text Classification of News Using Transformer-based Models for Portuguese

Isabel N SANTANA[1] , Raphael S OLIVEIRA[1], * Erick G S NASCIMENTO [1,2*]

[1]Manufacturing and Technology Integrated Campus – SENAI CIMATEC
Salvador, Bahia, Brazil
[2] Surrey Institute for People-Centred Artificial Intelligence, Faculty of Engineering and Physical Sciences
University of Surrey, Guildford GU2 7XH, United Kingdom

**Corresponding author: [3]erick.sperandio@fieb.org.br*

Abstract[1]

This work proposes the use of a fine-tuned Transformers-based Natural Language Processing (NLP) model called BERTimbau to generate the word embeddings from texts published in a Brazilian newspaper, to create a robust NLP model to classify news in Portuguese, a task that is costly for humans to perform for big amounts of data. To assess this approach, besides the generation of embeddings by the fine-tuned BERTimbau, a comparative analysis was conducted using the Word2Vec technique. The first step of the work was to rearrange news from nineteen to ten categories to reduce the existence of class imbalance in the corpus, using the K-means and TF-IDF techniques. In the Word2Vec step, the CBOW and Skip-gram architectures were applied. In BERTimbau and Word2Vec steps, the Doc2Vec method was used to represent each news as a unique embedding, generating a document embedding for each news. Metrics accuracy, weighted accuracy, precision, recall, F1-Score, AUC ROC and AUC PRC were applied to evaluate the results. It was noticed that the fine-tuned BERTimbau captured distinctions in the texts of the different categories, showing that the classification model based on this model has a superior performance than the other explored techniques.

***Keywords**: Deep Learning, NLP, BERT, BERTimbau, Transformers, Word2Vec, News classification, Portuguese.*

[1] The author is grateful to Professor Dr. Junia Matos for editing this paper [Manufacturing and Technology Integrated Campus – SENAI CIMATEC, Salvador, Bahia, Brazil (junia.matos@fieb.org.br)].

1. Introduction

Understanding human language has been one of the greatest challenges of Artificial Intelligence. The Natural Language Processing (NLP) area faces big challenges such as the context understanding, sentiment analysis or figurative language interpretation. Recently, NLP has been boosted with deep learning, performing significant advances and transforming the interaction between humans and machines.

Language corpus are sets of text and audio data for a given language, while corpora is the plural of corpus. Language models are usually pre-trained to process large amounts of text. They serve as a general basis for a diverse range of applications, becoming basic models which can be leveraged to solve problems on specific tasks through transfer learning. However, training these foundation models is generally extremely expensive, requiring a huge computational effort. Recently, the emergence of the Transformer-based neural network architecture and the attention mechanism (Vaswani *et al.*, 2017) has decreased the required computational resources to perform such tasks. Transformers have made it feasible to train foundation models on large volumes of text and general scope audio data, making these models the most powerful.

The most part of pre-trained models based on Transformers publicly available was trained in English language, but there are other models in other languages, such as Google's Multilingual BERT, Bidirectional Encoder Representations from Transformers (Devlin *et al.*, 2018), which encompasses 104 languages including Portuguese. However, these models are not trained as a big and representative linguistic corpus of a specific language, opening a gap in the foundation models construction to other languages. In this context, in 2020, the NeuralMind company released a Bert model trained in Brazilian Portuguese named BERTimbau (Souza *et al.*, 2020). This initiative opened new possibilities for research in the field of NLP in Portuguese.

One of the main reasons to use BERT is due to the fact that its self-attention mechanism can learn the semantic relationship between words. Also, the model does not process input words in sequence, opposed to models based in Recurrent Neural Networks (RNN) (Sherstinsky,2020). Furthermore, this model is based on unsupervised learning, which means that they are trained on a corpus with unlabeled data. Thus, this study compares different types of NLP techniques to a specific task of classification in text categories of news written in Brazilian Portuguese from the regional newspaper "A Tribuna" from Vitória/ES, and proposes a methodology for Brazilian Portuguese text classification using BERTimbau as a base, then specializing it as for this case study. News classification tasks become costly when we handle large amounts of data with different textual structures and contexts.

The methodology to the classification demonstrated in this paper was performed through a Machine Learning model fed by numerical vectors that represent the words, named word embeddings (WE) (Collobert *et al*., 2011). The experiment also did a comparative study between WEs made from BERTimbau and those made from an older but widely used NLP technique called Word2Vec (Mikolov *et al.*, 2013). This paper is organized as follows: Section 1 presents the Introduction. Section 2 presents Theoretical Reference. Section 3 presents the Methodology, while Section 4 presents the Results and Discussion, and finally Section 5 presents the Conclusion.

2. Theoretical Reference

One of the biggest challenges in NLP is training language models on large linguistic corpora. Before the advent of WE, a widely used technique was Bag of Words (BoW). It transforms the analyzed corpus into a sparse matrix. The rows of this matrix represent the number of documents to be analyzed, and the columns represent the presence or absence of a given token (word) in the document (Wallach, 2006) . A similar technique, used in this study as a baseline, is TF-IDF (Term-Frequency - Inverse Document Frequency). It calculates the frequency of a term in a document divided by the inverse of its frequency in all other documents in the corpus, thus indicating the importance of a word for a document (Stein & Silva, 2016). Language

models fed with BoW and TF-IDF can perform well on tasks such as language identification, spelling correction, gender classification, and entity name recognition (Russell *et al*., 2010). However, these techniques fail to capture context and semantics of words and add disadvantages such as high sparsity of word representations, since the size of each vector is equal to the number of distinct words in the corpus, but they carry very little relevant information.

In 2011, the first techniques for building word representation in a dense vector space, the Word Embedding (WE) (Collobert *et al*., 2011), appeared. The generation of a WE considers words that are close to the target word, in order to capture the context and semantics. In this way, representations of similar words occupy dense and close vector spaces, and therefore, it becomes possible to calculate the degree of similarity of words through similarity analysis techniques, such as cosine similarity (Sidorov *et al.*, 2014). In 2013, Google presented the unsupervised learning technique Word2Vec (Mikolov *et al*., 2013). For the WE generation, Word2Vec has the CBOW (Continuous Bag of Word) and Skip-gram approaches. CBOW aims to predict a target word from a number of context words, located before and after that word. The number of words is defined in the "window size". Skip-gram, on the other hand, tries to predict a number of context words from an input target word. The CBOW technique has the advantage of being simpler and its training time is shorter than Skip-gram. In addition, the first technique captures syntactic relations between words better than the last one. However, the Skip-gram technique can better capture semantic relationships between words (Souza *et al.*, 2020). An example of a learned relationship between WEs is realized by subtracting two vectors and adding another WE to the result. For example, Paris - France + Italy = Rome (Mikolov *et al.*, 2013, p.9).

The Word2Vec technique builds context-free WE, which means that every word in your vocabulary is represented by the same dense vector regardless of what a word means in a given sentence. So its biggest drawback is that it cannot differentiate words that are homographs. Another weakness is that the Word2Vec-based model cannot create WE from words that are not present in its vocabulary. WE were commonly used as embedding layers in RNNs such as LSTM (Hochreiter &

Schmidhuber, 1997) and GRU (Cho *et al.*, 2014), which require higher computational power to train. Because of their inherently recurrent architecture, they cannot be trained in parallel. As a consequence, this feature reduces the gains in training these networks on supercomputers and other parallel architectures.

At that same time, the concept of attention mechanism (Vaswani *et al.*, 2017) and the "Sequence-to-Sequence" approach for solving some problems, such as translation between languages, surged. In 2018, BERT emerged as a model based on the neural network architecture called Transformers, which promoted a major revolution in the field of NLP. The Transformers architecture is shown in Figure 1.

The Transformer has an Encoder-Decoder architecture. The model receives as input a sequence of words, which will be encoded in WE, and later they will be decoded in words as outputs, for a given NLP task (e.g. text translation). The original proposed architecture is composed of a stack of six encoders and six decoders that are identical, but do not share the same weights. Each encoder has a MultiHead Self-Attection layer, and a feed-forward neural network. The Positional Encoding is present at the input of the model, being responsible for assigning the position of each word in a sentence, thus preserving the order of the words (Vaswani *et al.*, 2017).

The Self-Attention layer is responsible for generating a WE for each word. The WE of a word is based on the weighted sum of all the other words in the sequence, where those that are important to the target word will receive greater importance. The idea of this mechanism is to flag that the meaning of the target word can be better explained by looking at the other words in the sequence, storing contextual information in each representation. The feed-forward takes as input the output produced by the attention mechanism and sends it to the next encoder. The structure of the decoder is similar to the encoder structure, but with the addition of an intermediate attention layer that signals the decoder to "pay attention" to the most relevant words in the sentence. Despite the existence of other predecessor models based on Transformers, BERT was the first to capture the context to the left and right of the target word in a bidirectional way (Delvin *et al.*, 2018). This makes it easier for the model to find relationships among words.

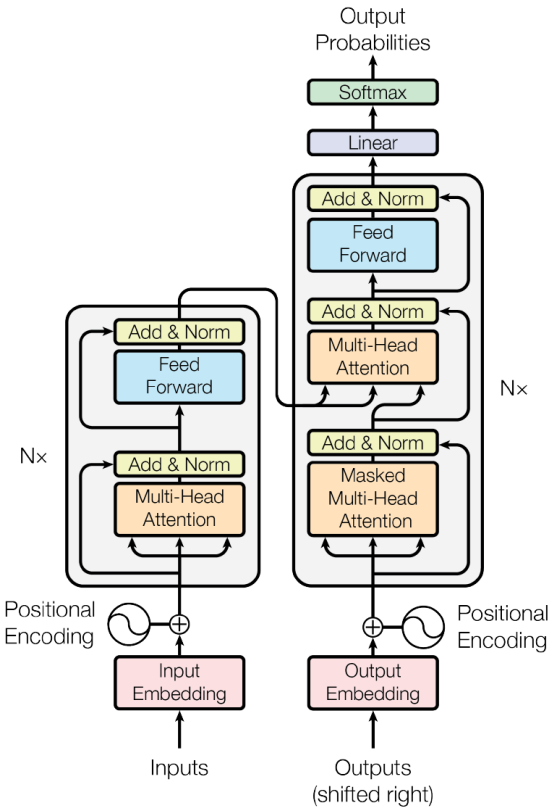

Figure 1: Transformer Architecture Model.

Sources: (Devlin *et al.*, 2018)

This new WE building process, unlike RNN architectures, does not depend on the sequence of words. So multiple sentences can be processed in parallel. Such independence made it possible to train these models on large corpora, making these models very powerful. As a consequence, we are now able to work with these large pre-trained models taking advantage of transfer learning in any NLP task. Regarding BERT, this model has a stack of twelve encoders and twelve attention mechanisms for the Base version and 24 encoders and sixteen attention mechanisms for the Large version. The feed-forward layers are 768 for BERT-Base and 1024 for BERT-Large. In addition, they have 110 and 355 million parameters, respectively. BERT was trained on Wikipedia and the Book Corpus (Devlin *et al.*, 2018). BERTimbau is the first BERT model that was trained on the Brazilian Portuguese BrWaC corpus, the Brazilian Web as Corpus (Souza *et al.*, 2020). Its pre-trained model has Large and Base versions.

Transformer-based models can be applied to help several everyday problems. In journalism, the large volume of information that a journalist is exposed to, can turn one's work of collecting and classifying content into an unfeasible activity. So, artificial intelligence can serve as a supporting tool for these professionals' activities. Therefore, this study proposed a model trained with WE that can classify news by its context, in categories defined to fit the problem presented. In summary, the methodology can be applied to any set of texts, with any categories that exist according to the purpose of classification.

The Gradient Boosting Decision tree model (GBDT) was used to perform the news classification in all stages of the research. GBDT is an ensemble boosting approach that combines other models of decision trees trained in sequence. In each iteration, the model uses the errors obtained by each previous decision tree to adjust the next one, aiming at improving the final results. (Ke *et al*., 2017).

In the first stage of this research, described in the Methodology section, the K-means algorithm (MacQueen, 1967) was used to rearrange news into a smaller number of categories. K-means is an unsupervised learning algorithm. It aims to separate data into clusters according to their similarities. K refers to the desired number of clusters.

3. Methodology

The study was implemented following steps illustrated in Figure 2. It was developed with the Python programming language (version 3.7.11). The K-means algorithm from Scikit Learn was used to plot the clusters. WE for Word2Vec were built with the open source library Gensim (version 3.6.0). In the fine-tuning step of BERT, the pre-trained BERTimbau model from the BERT-base version was used. The open source PyTorch library was used for GPU usage. BERTimbau was manipulated with Transformers Hugging Faces library (version 4.9.0), from the open-source community of pre-trained Deep Learning models "Hugging's Face" (Wolf *et al*., 2019). The model used for news classification was the XGBClassifier, from the XGBoost library (version 0.90), which uses Gradient Boosting technique.

The data analysis was performed on a collection of news from the newspaper "A Tribuna", collected between the years 2004 and 2007. The database was labeled according to newspaper sections, with a total of 42,123 samples and is distributed in nineteen unbalanced categories. About 77% of data is concentrated in six categories, being "Economy" the largest category with 6,557 samples and the smallest category, "Everything to do", with only 30 samples.

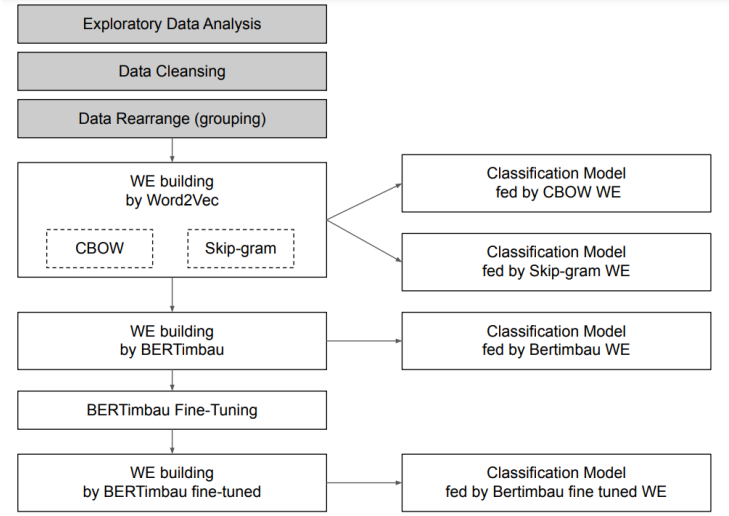

Figure 2: Steps of the study

To explore news texts, histograms of word amounts were generated by class and also for the whole database (Figure 3).

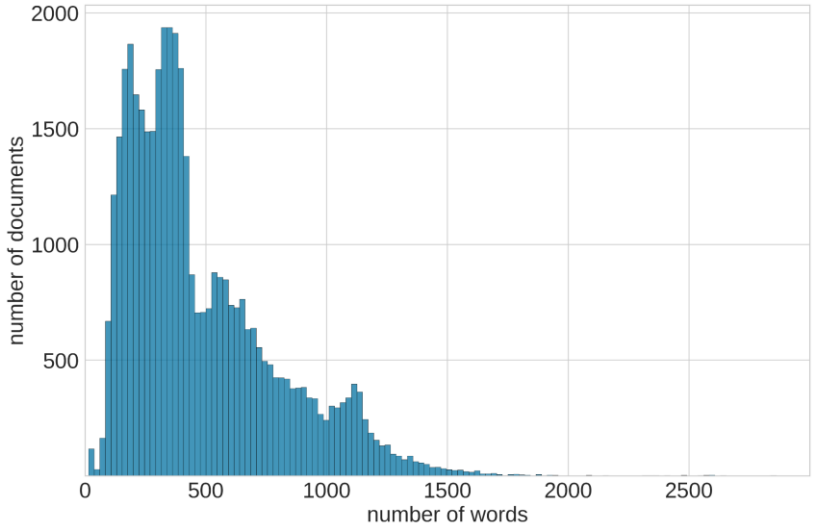

Figure 3: Histogram that shows the number of words per document

In the overall histogram, it was noticed that most texts have between 100 and 450 words, but there is a significant amount of texts with 500 to 1000 words. The average is 485 words, with the smallest text containing 14 words and the longest 5,523 words. In order to group those news in more general categories, the first step was to generate word clouds for each category to know the most cited words and try to extract some information about the subject and context of the existing categories. Before clouds were generated, the text was cleaned by removing very frequent words, HTML tags, numbers, accentuation, punctuation, prepositions and articles. By observing those word clouds, it was possible to clearly identify the subject matter in some classes, while the distinction was not so clear in others.

To reduce the unbalance of the dataset and facilitate the regrouping process of the minority categories, the K-means and TF-IDF techniques were used in order to identify whether such categories could be incorporated into larger ones or simply be discarded. The definition of the initial number of clusters was based on the Elbow method (Kodinariya & Makwana, 2013) using the Inertia metric. The analysis resulted in six clusters. An aspect also taken into consideration was the fact that almost 80% of instances were distributed in only six of nineteen existing categories. After clusters were defined, heat maps were plotted which indicated the similarity of various categories with six major clusters. In this way, starting nineteen classes were rearranged into only ten.

The next step was to build WE using Word2Vec with CBOW and Skip-gram architectures. The goal was to feed a classification model, which served as a baseline for comparing classification results with WE built by Transformers in following steps. The Word2Vec model from the Gensim API[2] was used to generate the WE of the words. The size of the output WE vector was 768 to be equivalent to the one built by BERT, which also produces an embedding of the same size. The window-size chosen for Skip-gram and CBOW was five words.

[2] API: Application Programming Interface

After WEs were generated, each news text had the amount of WE equivalent to the amount of words. Then, the Doc2Vec method was used to create only one WE per text. For each news story, a single WE was calculated by averaging the WE derived from each text. To train the classification model, the doc2vecs were divided into train and test in the proportion 80% and 20% respectively. The model chosen to be the baseline was Word2Vec with CBOW. This architecture was chosen for having the fastest processing, given the size of the dataset used in this study.

In order to discover the power of Transformers in capturing the news context, for this step, the pre-trained BERTimbau Base model was used, with twelve feed-foward layers of dimension equal to 768 and twelve attention mechanisms. The BERT requires a specific input format to be fed. Mandatorily, word sequences must have a fixed size of 512 words. Using BERT Tokenizer, words were transformed into tokens, which represent BERT's vocabulary words. The sentences were marked with special tokens [CLS], indicating the beginning of the sentences, and [SEP] delimiting the end. In fact, BERT's Tokenizer transforms subwords into tokens. This approach enables BERT to generate a representation for an unknown word. Next, the BERTimbau model was fed with a list of integers that indicate the position of each token in the BERTimbau vocabulary, the "input ids", to obtain WEs of the sentences. As output of each sentence, 510 WE were generated (ignoring the first and last tokens of markup) with dimension equal to 768. Then, the average of these WEs was calculated to reduce it to just one per sentence. To train the BERT classification model, the doc2vecs were divided into train and test in the proportion 80% and 20% respectively.

The next step was to perform fine-tuning of BERTimbau, which is the process of training the pre-trained model on a specific dataset. The BERTimbau (Bert-Base) model fine-tuning technique chosen was to train all layers of the model with the masked-language modeling (MLM) strategy. This approach consists of providing a sentence to the model with some masked input tokens and the output should be the same complete sentence. The model's attempts to guess tokens makes BERT better understand the usage of words in a specific context. Thus, at the end of training, the

weights of the layers of its entire architecture are updated and adjusted according to the "A Tribuna" dataset.

The first step for the fine-tuning was to transform words from the dataset into tokens. Then, the new words from “A Tribuna” dataset were added to the BERTimbau vocabulary. Next, those texts were transformed into a sequence of 512 tokens. Then, a proportion of 15% of tokens to be masked was defined, so that the model would try to guess their values during training, as suggested in (Devlin *et al.*, 2018). Finally, the data was split into train and test at the ratio of 90% and 10%, respectively. For model learning, the new data was displayed to BERTimbau. Each sample in the training set is analyzed and the error is computed, so that the weights are adjusted to minimize the overall error according to a loss function. Once all samples were analyzed, an epoch has been finished. The final neural network is a set of weights fine-tuned to that specific purpose, according to the desired result. During the training, four epochs were required for training, with a learning rate equals to 2e-5. A new classification of the dataset was performed using the fine-tuned BERTimbau. The result obtained was compared with the original BERTimbau classification.

3.1. Performance Evaluation

The indicators and metrics used to analyze the results were as follows:

i. Confusion Matrix (CM) - enables a more detailed observation of the model's hits and misses in relation to the expected result. e. Measures of the quality of classification are built from a confusion matrix which records correctly and incorrectly recognized examples for each class (as cited in Sokolova *et al.*, 2006, p.1015).

ii. Accuracy e weighted Accuracy - these metrics that report the model's total hits. The weighted accuracy is more sensitive than the accuracy, especially when dealing with unbalanced datasets. The higher the better.

iii. Precision - indicates how correctly the model actually classified instances as true positive correctly. The higher the better.

iv. Recall - indicates the percentage of true positives that the model got right in relation to the existing total. The higher the better.

v. F1-Score - also known as F-measure is a harmonic mean of precision and revocation.

vi. 6. Area Under the Curve - Receiver Operating Characteristic Curve (AUC ROC) - the ROC curve is presented on a plane and is constructed by observations of the performance of a binary classifier on the ratio of True Positive Rate and False Positive Rate. The coordinate (0,1) in the ROC space indicates the non-existence of false negatives. The larger the area under the ROC curve, the better.

vii. Area Under the Curve Precision Recall Curve (AUC PRC) - the PRC curve is a metric best suited for use on unbalanced bases. Analogous to the ROC curve, it is obtained by observations of the performance of a binary classifier on the ratio of Accuracy and Revocation. The larger the area under the ROC curve, the better (Davis & Goadrich, 2006).

4. Results and Discussion

The overall results obtained in classification with Word2Vec, using CBOW and Skip-gram are illustrated in Table 1. Regarding Word2Vec, the best results were obtained with the Skip-gram approach. The general accuracy obtained was only 57.8%. The weighted accuracy was 57%. The precision was 55%.

Table 1: Comparative table of obtained results

Word2Vec		
Metric	**CBOW**	**Skip-gram**
Accuracy	0.557	0.578
W. Accuracy	0.550	0.570
Precision	0.531	0.554
Recall	0.485	0.511
F1-Score	0.496	0.524
AUC ROC	0.878	0.889
AUC PRC	0.528	0.552

Figures 4 and 5 illustrate the classification reports for CBOW and Skip-gram with metrics Accuracy, Revocation, and F1-Score separated by categories.

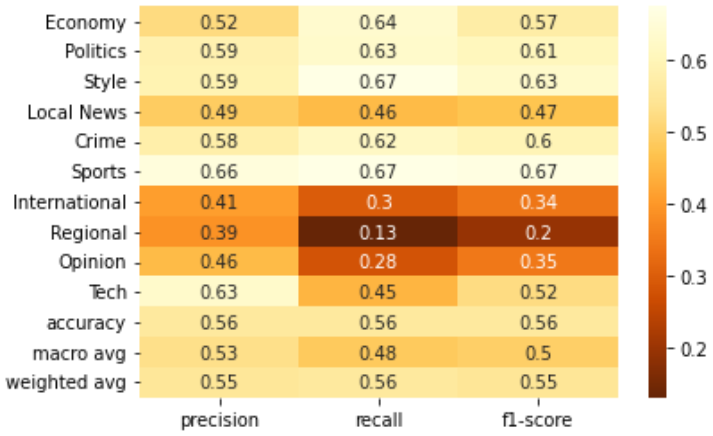

Figure 4: Metrics of classification performed by the model fed with Word2Vec CBOW WE

In Figure 5 where the best results are displayed, it is possible to highlight "Sports" and “Tech” as categories that the model was most accurate with 67%. The "Regional" category obtained the lowest recall result, with a value of only 19%. The best F1-Score result was also for "Sports" with 68%. The model performed below 50% in all metrics for the "Local News", "International", "Opinion" and "Regional" categories.

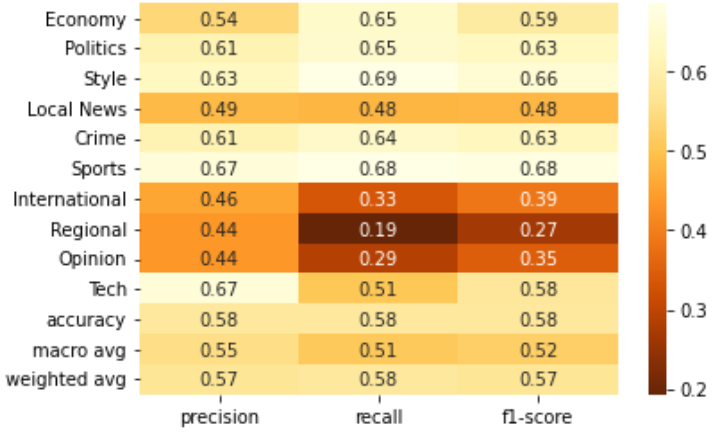

Figure 5: Metrics of classification performed by the model fed with Word2Vec Skip-gram WE

Analyzing confusion matrices of Figures 6 and 7, it can be seen that the model fails in classifying some categories, especially “International”, “Local News”, “Regional” and “Opinion”. This is an indication that the WE of words, for being the same

regardless of the context, are not efficient to be applied in scenarios with extensive texts and diverse contexts.

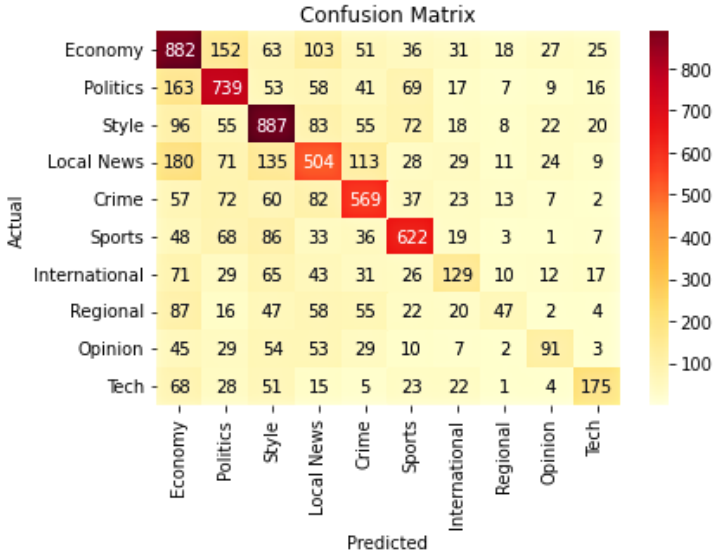

Figure 6: Confusion Matrix of classification model fed by Word2Vec CBOW WE

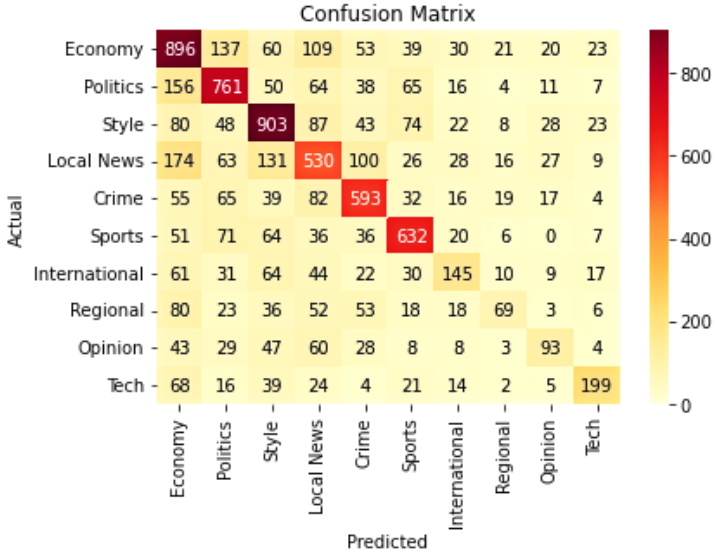

Figure 7: Confusion matrix of classification model fed by Word2Vec Skip-gram WE

For a better comparison of the model's performance in classifying categories, AUC ROC and PRC plots were generated. In both approaches, ROC curves shown in Figures 8 and 9 were very close with values of 88% and 89%.

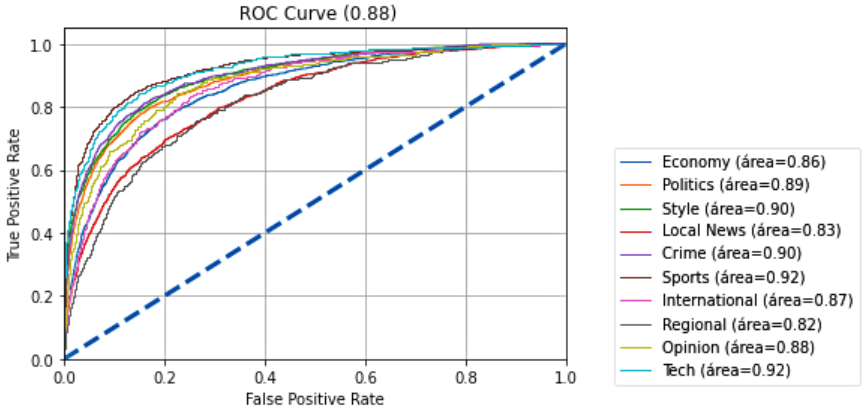

Figure 8: Word2Vec CBOW ROC Curve

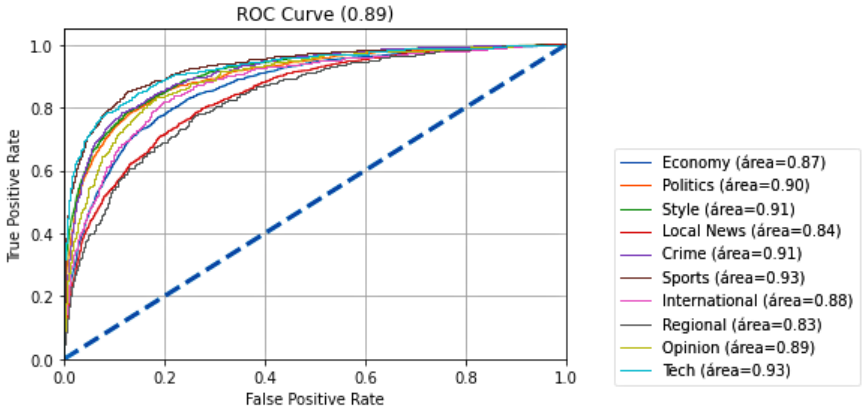

Figure 9: Word2Vec Skip-gram ROC Curve

The AUC PRC result for CBOW and Skip-Gram was only 53%, Figure 10, and 55%, Figure 11, respectively. Analyzing these charts, it is noticeable the existence of overlapping curves for some thresholds, indicating that the model is not discerning well these existing categories. The "Regional" category obtained the curve with the smallest area in both approaches, highlighting the CBOW approach which obtained a value of only 22%.

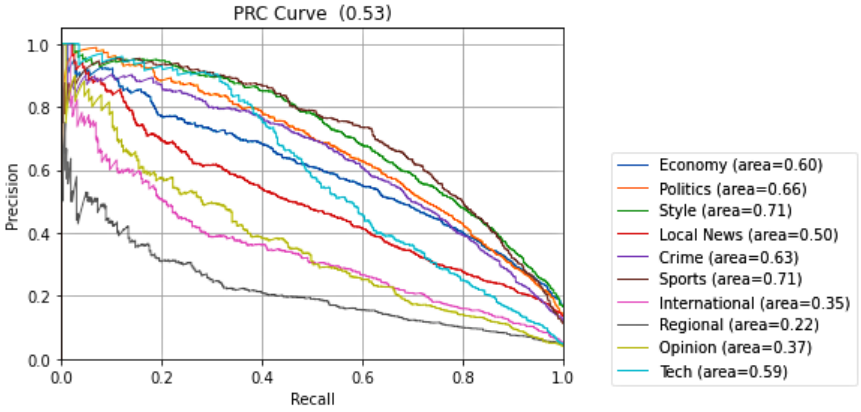

Figure 10: Word2Vec CBOW PRC Curve

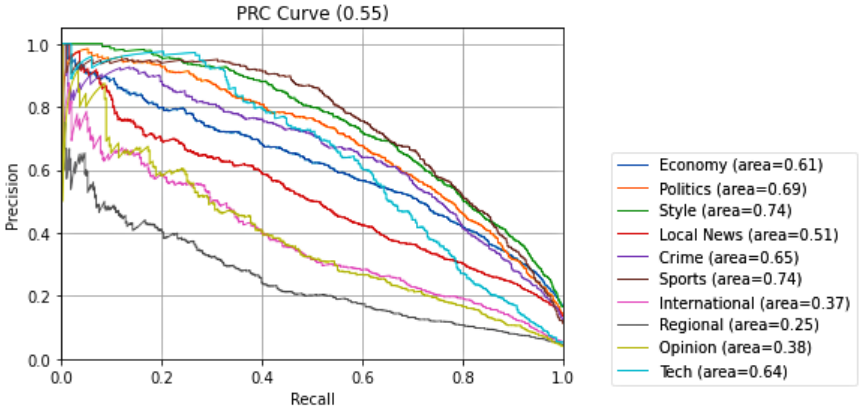

Figure 11: Word2Vec Skip-gram PRC Curve

Finally, two plots are shown with clusters resulting from the classification using CBOW in Figure 12 and Skip-gram in Figure 13. In both plots, although it is possible to see most of these groups, they are not clearly separated.

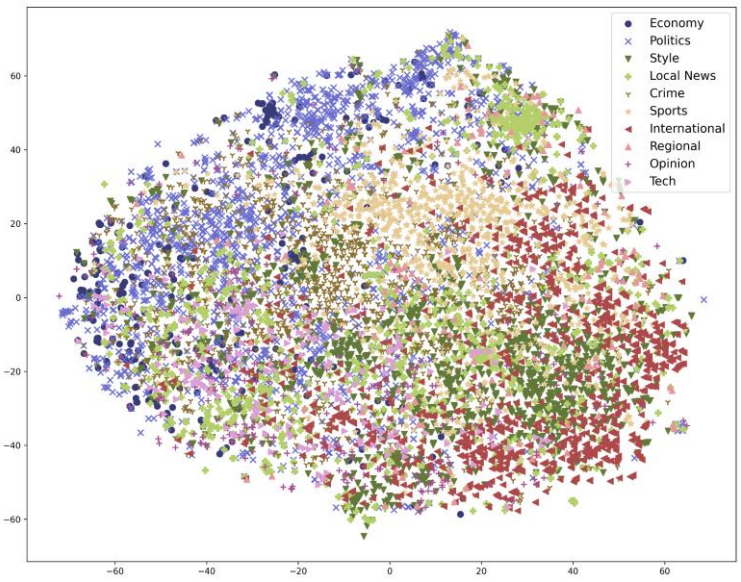

Figure 12: Resulting clusters from Word2Vec CBOW

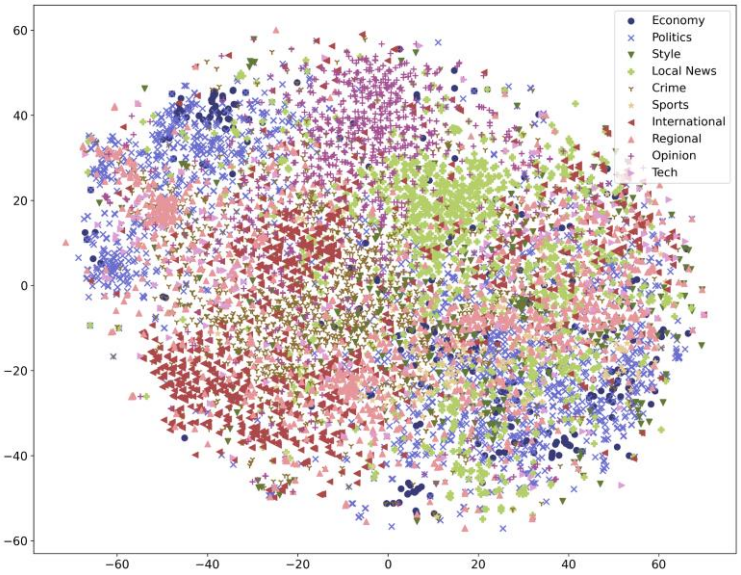

Figure 13: Resulting clusters from Word2Vec Skip-gram

Analyzing metrics of the original BERTimbau, there was an increase of about 25 percentage points in metrics of accuracy, precision, revocation, and F1-score. According to metrics shown in the "Original" column of Table 2, the results achieved with WEs from the original BERTimbau were significantly better than those from Word2Vec.

Table 2: Comparative table of the obtained results

	BERTimbau	
Metrics	**Original**	**Fine-tuned**
Accuracy	0.811	0.877
W. Accuracy	0.810	0.880
Precision	0.799	0.873
Recall	0.774	0.853
F1-Score	0.784	0.862
AUC ROC	0.967	0.982
AUC PRC	0.830	0.897

In Figure 14, metrics by category can be analyzed and it can be seen that results are more balanced. The lowest accuracy is 65% for the "Regional" category and the highest is "Sports", which reached a value of 94%. A highlight is the improved recall for categories "International", "Opinion" and "Regional", which had values two times higher than those obtained with the Word2Vec technique. The best F1-score result is for the "Sports" category with 93% and soon after, for "Style" with 89%.

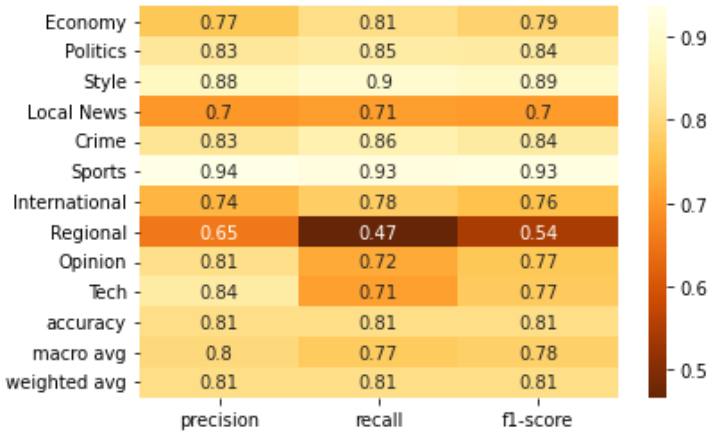

Figure 14: Metrics of classification performed by the model fed with BERTimbau WE

The Confusion Matrix, in Figure 15, demonstrates how the model has a good accuracy in predicting the categories. However, the "Regional" category, similarly

as in the previous model, continues to be the one with the lowest number of correct predictions.

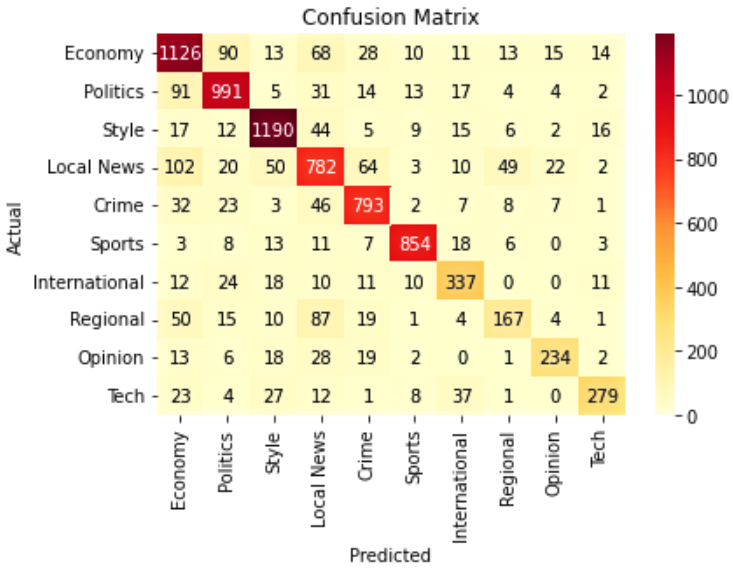

Figure 15: Confusion matrix of classification model fed by BERTimbau WE

The BERTimbau ROC curve, illustrated in Figure 16, compared to the Word2Vec - Skip-gram ROC curve, increased from 89% to 97%. The AUC of all categories obtained a percentage greater than 94%. The category "Sports" achieved the highest AUC with a value of 99%.

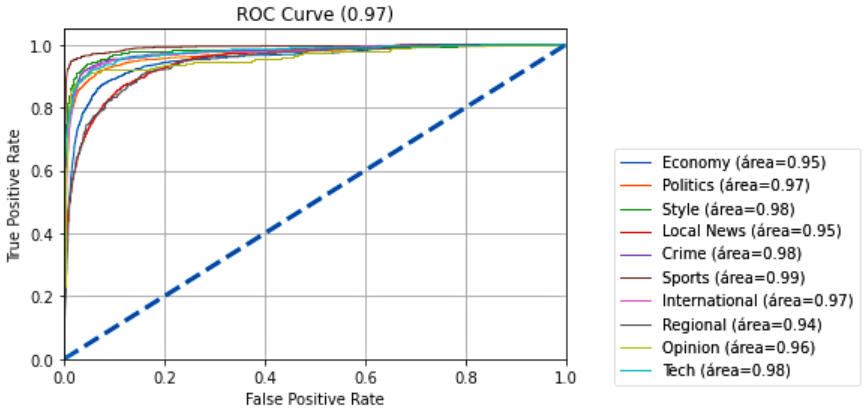

Figure 16: BERTimbau ROC Curve

A relevant point was the increase in the AUC PRC which increased from 55%, Figure 10, to 83%, Figure 17, which demonstrated that BERTimbau was able to capture

nuances of different subjects, performing well in predicting even for minority categories.

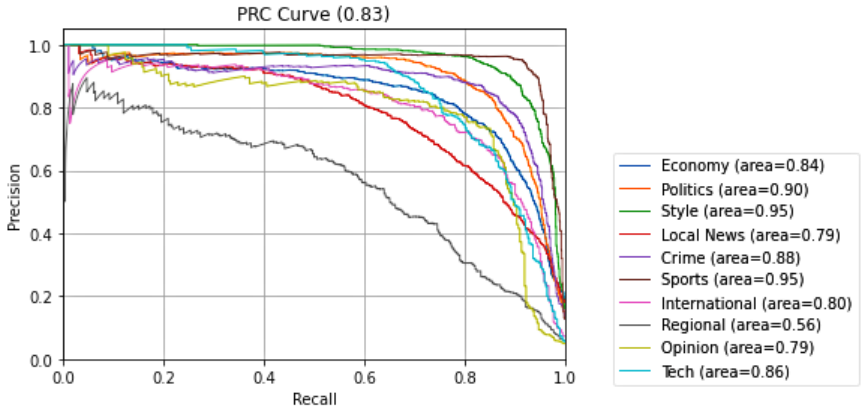

Figure 17: BERTimbau PRC Curve

The K-means output illustrated in Figure 18 shows a better distinction of clusters, however there is still not a good distinction between the “Regional”, “Local News” and "International" clusters.

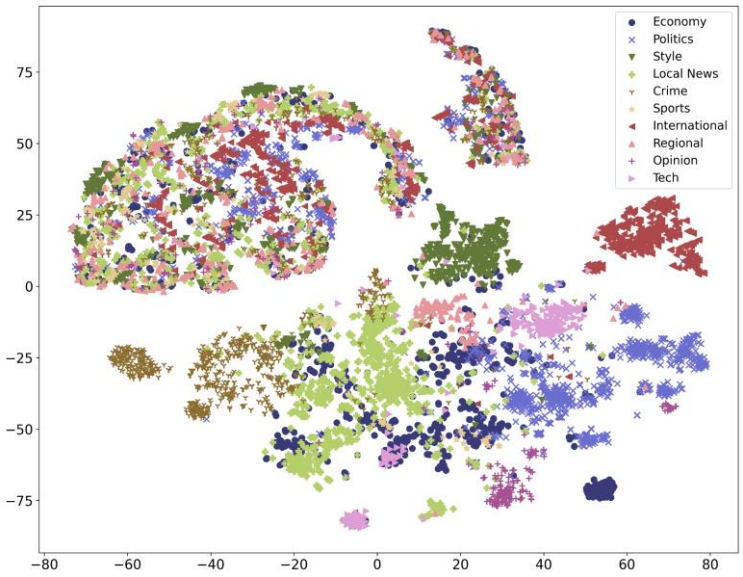

Figure 18: Resulting clusters from BERTimbau

The next step was to perform the BERTimbau fine-tuning. Perplexity (Faleiros & Lopes, 2016) was the metric used to evaluate the fine tuning of the model . It tries to show how likely the model is to be confused when choosing a word. So the lower

the Perplexity, the better the model. The value obtained at the end of the BERTimbau training was approximately 4.21. After that, the trained model was used to generate new WE for the new dataset classification. With the new WE created, a better performance than the original BERTimbau was obtained. All results of the applied metrics had a relevant increase, demonstrating that the fine-tuned BERTimbau absorbed characteristics of the texts of database context. Table 2 (fine-tuned column) also shows results of the overall metrics obtained. The accuracy of the model reached approximately 87.7%. The weighted accuracy was 88%. The precision was approximately 87%, highlighting the 95% precision achieved for "Sports" (Figure 19). Another interesting result was the 82% precision for the "Regional" category, a considerable improvement over the 65% result obtained by the original BERTimbau. The highest recall was also for the "Sports" and the lowest was 68% for the "Regional" category. As in the original BERTimbau, the best F1-Score result was from the "Sports". There was an improvement over the predecessor from 93% to 96%.

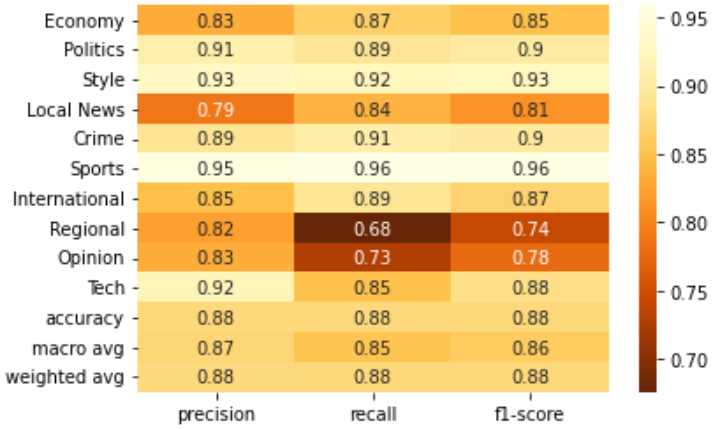

Figure 19: Metrics of classification performed by the model fed with fine tuned BERTimbau WE

When comparing the confusion matrix in Figure 20 with the others, it is noticeable that the accuracy performance of the classifier fed with the WE built by the fine-tuned model is superior.

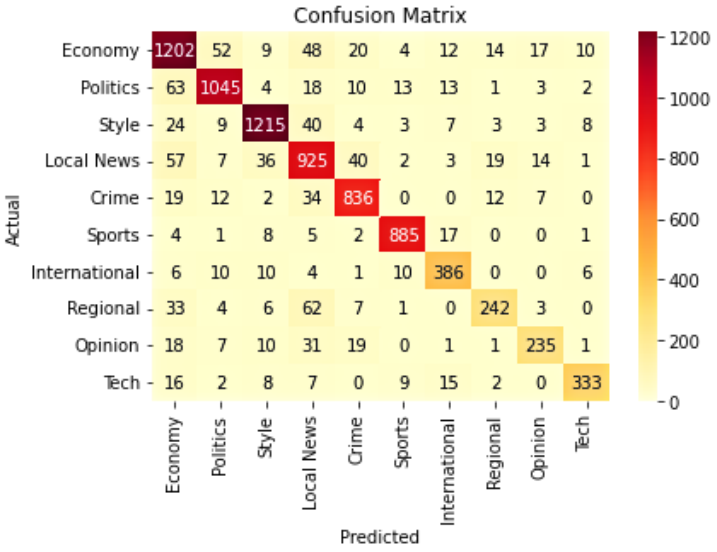

Figure 20: Confusion matrix of classification model fed by fine-tuned BERTimbau WE

The ROC curve (Figure 21) increased from 97% to 98%. The AUC of all categories obtained a percentage greater than 97%. The "Sports" category obtained an AUC of 100%.

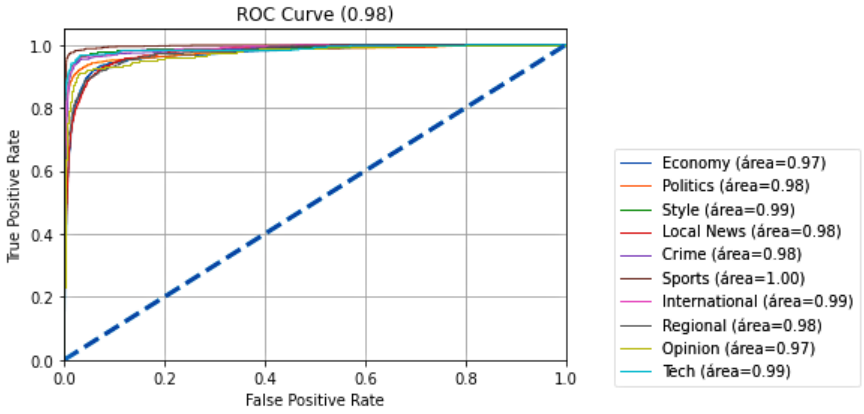

Figure 21: Fine-tuned BERTimbau ROC Curve.

The AUC PRC, illustrated in Figure 22, obtained a value of 89.7%. Those categories with the smallest area were "Opinion" with 79% and "Regional" with 80%. The "Regional" category AUC PRC was only 56% using the original BERTimbau.

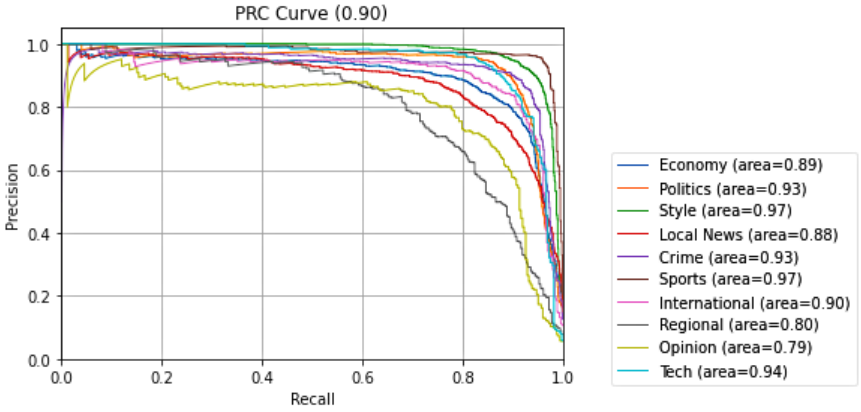

Figure 22: Fine-tuned BERTimbau PRC Curve.

The plot shown in Figure 23, clearly demonstrates ten clusters better defined, and it is evident how much better the fine-tuned BERTimbau model performed compared to others models developed in this study.

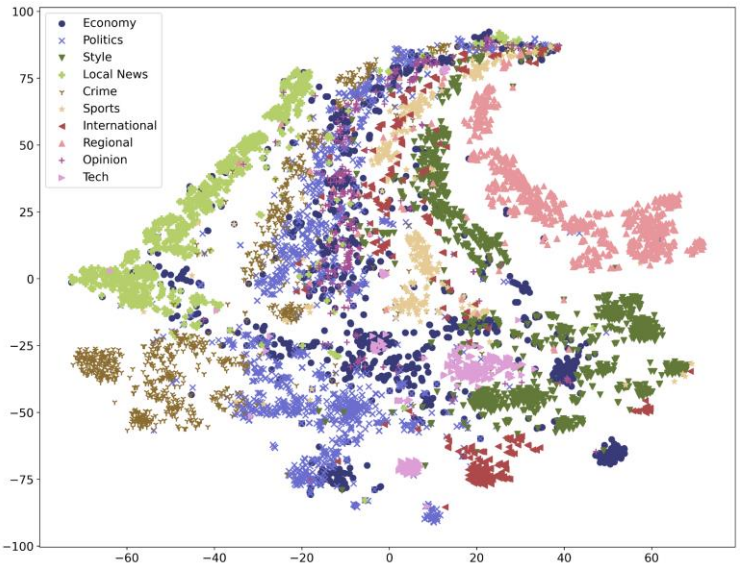

Figure 23: Resulting clusters from fine-tuned Bertimbau

When observing the obtained results, it can be seen that those percentages achieved with techniques that used BERTimbau were considerably higher. The main reason for this is because the WE built by BERT are able to capture context of words, unlike the WE built with Word2Vec, which result in context-free representations of words. This difference is evidenced when analyzing results of PRC curves for each category

of news. Those results for Word2Vec show curve areas below 50%, indicating the underfit of the model predictions for minority categories. However, when analyzing the fine-tuned BERTimbau chart (Figure 22) it is noticeable that those curve areas for all categories are considerably larger, some of them reaching a value up to three times larger. This fact demonstrates that the last model was able to capture semantic particularities of those ten categories.

Proceeding with the analysis of results, it is perceived that when the fine-tuning of BERTimbau was performed, the model specialized itself in the language of the explored dataset. Such circumstance is evidenced in the PRC curve of the "Regional" category. In the original BERTimbau model, its area was 56%. With the fine-tuned BERTimbau, its result increases to 80%. Overall, after the fine tuning, there was a significant improvement in classification for all categories.

5. Conclusions

By performing the comparative study of WE building techniques, it is evident that the dense vector-based representations of words generated by those models based on the Transformers architecture are far superior in relation to their predecessors. The representations generated through classic BoW technique, besides not capturing context, have the problem of the high dimensionality of vectors defined by the size of vocabulary, making the training computationally costly for large corpora. Word2Vec generates a WE for each word in your vocabulary. This feature makes this representation context-free . However, trying to reduce all contexts of a word into a single vector representation did not prove to be a very efficient method. This limitation was perceived when analyzing those results. It was noticed that the model confused categories such as "Politics" and "Economy" whose contexts are distinct but share a similar vocabulary.

BERT, on the other hand, generates numerous representations of each word, depending on the context in which it is presented. This makes its WE context-dependent, so it is possible to capture semantic nuances of different texts. When

applying the original BERTimbau, a considerable improvement in the performance of the news classification task was noticed. After fine-tuning, there is an increase in the classification performance, especially when analyzing the result of PRC curves for each category. During the fine-tuning process, a difficulty encountered was related to the fine-tuned BERTimbau Tokenizer. The token building time of new words coming from “A Tribuna” dataset was much higher than the building time of a token from the main vocabulary. So, it became necessary to use the original BERTimbau Tokenizer to generate those new tokens. For future comparative studies, it is suggested to improve Doc2Vec generation technique in order to choose the one that best preserves news features. Another interesting suggestion is to use the BERTimbau-large version and other models based on Transformers architecture. Explainable AI can also be applied to better understand the performance of attention mechanisms.

Transformer-based language models have taken NLP to a level of excellence when it comes to human language interpretation. Several advances have been noticed in information seeking, speech recognition, Text-to-Speech and dialog systems. Models like BERT and its successors, such as RoBERTa (Liu *et al*., 2019) and GPT-3 (Brown *et al.*, 2020) and its predecessors, GPT-1 and GPT-2, from Open AI, are being used by industry, commerce, health care, justice, but little has been produced for the Portuguese language. One of the main reasons is the little investment to build these models using large datasets in Portuguese.

The creation and specialization of language models for Portuguese, such as the fine-tuned BERTimbau, has captured language regionalisms, increasing its vocabulary and improving classification of news. Journalists may benefit when they need to categorize their large collections into more general subjects. They may also have greater support in searching for topics of interest to them, better filtering the huge amount of unstructured information they are exposed to. The model can also be used to improve recommendation systems according to the type of information consumed by readers. It could also improve aggregation systems, by collecting several different textual sources, linking news by similar contexts. Finally, beyond famous virtual assistants, Transformers-based models can contribute a lot to the improvement of people's daily lives.

6. Acknowledgements

The authors thank the Reference Center on Artificial Intelligence and Supercomputing Center for Industrial Innovation, both from SENAI CIMATEC, for the scientific, technical and computational resources support, as well as the NVIDIA/CIMATEC AI Joint Lab for the technical support. We also thank A Tribuna Newspaper that kindly provided us with their data. We would like to express our deeply felt gratitude to Professor Dr. Júnia Matos and Professor Dr. Bruno Menezes for the comprehensive and detailed peer-review of this document. Finally, we would like to thank Dr. Poliana Ramos Braga Santos for performing the proof reading of this paper as the beta-reader.

References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. Journal of machine learning research, 12(ARTICLE), 2493-2537.

Davis, J., & Goadrich, M. (2006, June). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning* (pp. 233-240).

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

Faleiros, T., & Lopes, A. A. (2016). Modelos probabilísticos de tópicos: desvendando o Latent Dirichlet Allocation (In Portuguese). *Universidade de São Paulo.*

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.

Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of Cluster in K-Means Clustering. *International Journal*, *1*(6), 90-95.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A Robustly Optimized BERT Pretraining Approach.. *arXiv preprint arXiv:1907.11692*.

MacQueen, J. (1967). Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability* (pp. 281-297).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Russell, S. J., Norvig, P., & Davis, E. (2010). *Artificial intelligence: A modern approach.* Prentice Hall.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, *404*, 132306.

Sidorov, G., Gelbukh, A., Gómez-Adorno, H., & Pinto, D. (2014). Soft similarity and soft cosine measure: Similarity of features in vector space model. Computación y Sistemas, 18(3), 491-504.

Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006, December). Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In *Australasian joint conference on artificial intelligence* (pp. 1015-1021). Springer, Berlin, Heidelberg.

Souza, F., Nogueira, R., & Lotufo, R. (2020, October). BERTimbau: pretrained BERT models for Brazilian Portuguese. In *Brazilian conference on intelligent systems* (pp. 403-417). Springer, Cham.

Stein, R. A., & Silva, A. D. B. (2016). Análise assintótica de algoritmo para geração de matriz termo-documento contendo TF-IDF.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., & Gomez, A. N. Kaiser,\Lukasz, & Polosukhin, I.(2017). *Attention is all you need*. Advances in neural information processing systems.

Wallach, H. M. (2006, June). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning* (pp. 977-984).

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020, October). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations* (pp. 38-45).