ISSN 0102-745X


Notas Técnicas

CBPF-NT-001/22
março 2022

# Using Deep Learning Transformer Networks to Identify Symptoms Associated with COVID-19 on Twitter

Vítor Machado, Clécio R. Bom, Kary Ocaña, Rafael Terra e Miriam B.F. Chaves

NIB Núcleo de Informação C&T e Biblioteca

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES

PÁTRIA AMADA BRASIL GOVERNO FEDERAL

dx.doi.org/10.7437/NT2236-7640/2022.01.001
Notas Técnicas, v. 12, n. 1, p. 1–10, 2022

# Using Deep Learning Transformer Networks to Identify Symptoms Associated with COVID-19 on Twitter

*Redes Transformers de Aprendizado Profundo para Identificação de Sintomas Associados à COVID-19 no Twitter*

Vítor Machado [1],* Clécio R. Bom [1],† Kary Ocaña [2],‡ Rafael Terra [2],§ e Miriam B. F. Chaves[2]¶
[1] *Brazilian Center for Physical Research (CBPF/MCTI)*
*Dr. Xavier Sigaud Street, 150, Ed. César Lattes,*
*Urca, Rio de Janeiro, RJ. CEP: 22290-180, Brazil e*
[2] *National Laboratory for Scientific Computing (LNCC/MCTI)*
*Getulio Vargas Avenue, 333,*
*Quitandinha, Petrópolis, RJ. CEP: 25651-076, Brazil*

*Submetido: 17/12/2021* *Aceito: 23/03/2022*

**Abstract:** This study aims to present a methodology to identify, through Twitter posts, predefined symptoms of COVID-19 aided by Deep Learning techniques, namely Transformers Networks. The proposed approach was evaluated on a public Twitter database in Brazilian Portuguese, using user reports of COVID-19 symptoms. We mine the Twitter database, extract phrases with symptoms, compare distributions, and build a database to construct high accuracy Deep Learning networks, which can be used to identify symptoms. We use a cross-validation procedure to evaluate the result's performance. Additionally, we interpret the results using a Local Interpretable Model-Agnostic Explanations (LIME) algorithm. We identified 907 tweets containing one or more of the 14 previously chosen COVID-19 symptoms. The most frequently reported symptoms were a cough (392), headache (154), runny nose (143), fever (124), nausea (106), and diarrhea (105) amongst users who reported at least one symptom. The BERT architecture identified all 14 symptoms reported in Twitter phrases in Portuguese, resulting in identifying each symptom with over 97% accuracy and over 0.95 of AUC-ROC at the test dataset. This project is a step towards a complementary tool to identify symptoms in future automated clinical settings, e.g., medical chatbots, to support faster clinical assessment in Portuguese.

**Keywords:** Deep Learning, Transformer Networks, BERT, COVID-19, Text Classification.

**Resumo:** O presente estudo objetiva apresentar uma metodologia para identificar, em publicações no Twitter, sintomas predefinidos de COVID-19 através de técnicas de Aprendizado Profundo, mais especificamente, redes Transformers. A abordagem proposta foi avaliada em uma base de dados de tweets públicos em português brasileiro usando relatos de sintomas de COVID-19 feitos pelos usuários. Mineramos a base de dados do Twitter para extrair frases com sintomas, comparamos distribuições e montamos uma base de dados para construir redes de Aprendizagem Profunda com alta acurácia que podem ser usadas para identificar a presença de sintomas. Usamos um processo de Validação Cruzada para avaliar a performance dos resultados. Adicionalmente, interpretamos os resultados usando o algoritmo de Local Interpretable Model-Agnostic Explanations (LIME). Identificamos 907 tweets contendo ao menos um dos 14 sintomas predefinidos de COVID-19. Entre os usuários que reportaram algum sintoma, os sintomas reportados mais frequentemente foram tosse (392), dor de cabeça (154), coriza (143), febre (124), enjoo (106) e diarréia (105). A arquitetura BERT identificou todos os 14 sintomas reportados em frases do Twitter em português, identificando cada sintomas com mais de 97% de acurácia e mais de 0.95 de AUC-ROC nos dados de teste. Este projeto é um passo no caminho do desenvolvimento de uma futura ferramenta clínica automatizada auxiliar para identificar sintomas, e.g., *chatbots* médicos, para auxiliar avaliações clínicas mais céleres em português.

**Palavras-chave:** Aprendizagem Profunda, Redes Transformers, BERT, COVID-19, Classificação de Texto.

## 1. INTRODUCTION

The Coronavirus pandemic disease (COVID-19) has caused millions of infections and deaths worldwide[1]. The COVID-19 infection has a long hospitalization time, from 10

*Electronic address: vmachado@cbpf.br
†Electronic address: debom@cbpf.br
‡Electronic address: karyann@lncc.br
§Electronic address: rafaelst@lncc.br
¶Electronic address: mbcm@lncc.br

[1] https://www.who.int/publications/m/item/weekly-operational-update-on-covid-19---8-november-2021

to 13 days [2], and due to its transmission through the air, hospitals became potential risk spots of new infections. Despite that, many people looked for hospitals when they thought they were infected, consequently exposing themselves to the COVID-19 virus in hospitals.

Providing diagnosis or specific health information directly to patients using chatbots is a way to reduce the exposure of people in hospitals and reduce crowds and contact with COVID-19 infected people, improving health professionals' quality of work and worktime [1, 2]. In addition, using a chatbot to collect the anamnesis information raises a new possibility for health professionals to process data information before direct contact with patients.

Early studies focused on identifying the symptoms experienced by patients infected by the virus, mainly those hospitalized or who received clinical care [3]. However, many infected people only experience mild symptoms or are asymptomatic and do not seek clinical care, although the specific portion of asymptomatic carriers is unknown [4, 5].

Transformer Network [6, 7] approaches are used in several applications for speech and text, such as text classification [8], translation [7], and text summarization [9]. Furthermore, several studies adapting Deep Learning for analyzing COVID-19 reports have been reported in [10–12].

As a result, we explored the use of social media, specifically Twitter, to study symptoms reported by people testing positive for COVID-19 or even those with symptoms but without a test result.

Our primary goals were to (i) verify that users report their experiences with COVID-19, including their positive test results and symptoms, and (ii) compare the distribution of self-reported symptoms with those reported in studies conducted in clinical settings in Brazilian Twitter COVID-19 reports, developing a deep learning network able to identify COVID-19's symptoms in text. Our secondary objectives were to (i) create a COVID-19 symptom database that captures tweets in which users express symptoms, developing a systematic workflow for automated symptom detection and processing, and (ii) collect a cohort of COVID-19-positive Twitter users' self-reported Portuguese information in the Brazilian media.

The remainder of this article is organized as follows. Section 2 presents some related works. Section 3 presents a background with popular solutions provided by Deep Learning for Natural Language Processing. Section 4 presents the COVID-19 Database construction. Section 5 introduces Machine Learning (ML) models used to analyze the COVID-19 database. Section 6 describes our results and and analyses. Finally, section 7 provides some concluding remarks and our perspectives for future works.

## 2. RELATED WORK

Sarker *et al.*, 2021 [13], analyze the spectrum of COVID-19 symptoms self-reported by users from Twitter, aiming to complement symptoms identified in clinical settings. This work was the first study that focused on extracting COVID-19 symptoms from public social media to the best of our knowledge. This article assists the research community, and it is part of a more extensive, maintained data resource — a social media COVID-19 Data Bundle[3].

Kumar *et al.*, 2021 [14], describe approaches for classifying tweets containing COVID-19 symptoms in three classes (self-reports, non-personal reports & literature/news mentions) in the Social Media Mining for Health Applications (SMM4H) shared tasks in 2021. BERT and XLNet were implemented for this text classification task. The best result was achieved by the XLNet approach with an F1 score of 0.94, a precision of 0.9448, and recall of 0.94448. Their results are slightly better than the average score, *i.e.*, an F1 score of 0.93, a precision of 0.93235, and recall of 0.93235, suggesting that this problem has several high-performing solutions.

According to [15], most internet users would be receptive to using health chatbots, although hesitancy regarding this technology likely compromises engagement. In addition, patients' perspectives, motivation, and capabilities need to be taken into account and listened to when developing health chatbots to improve their use frequency and quality.

Valdes *et al.*, 2021 [16], describe an automatic classification of Twitter posts related to COVID-19. It was oriented towards solving a binary classification problem, identifying self-reporting tweets of potential cases of COVID-19, and classifying tweets containing COVID-19 symptoms using models based on bidirectional encoder representations from Transformers (BERT). Based on the results obtained, the authors concluded that a model trained with quality domain-specific data (CT-BERT) could outperform a model trained with a much more significant amount of general (general purpose/non-specific) domain data (BERT).

In [17], Luo *et al.* describe a system that aims to automatically distinguish English tweets that self-report potential cases of COVID-19 from those that do not. It proposed a model ensemble that leverages pre-trained representations from COVID-Twitter-BERT, RoBERTa, and Twitter-RoBERTa. The model obtained F1 scores of 76% on the test set in the evaluation phase and 77.5% in the post-evaluation phase.

The study in [18] aimed to detect and characterize user-generated conversations that could be associated with COVID-19-related symptoms, as well as experiences with access to testing, and mentions of disease recovery using an unsupervised machine learning approach.

Overall, in the present study, we used Transformer Networks in a supervised machine learning framework to characterize self-reporting symptoms and mentions of recovery related to COVID-19 from Brazilian Portuguese Twitter social media. Many users reported symptoms related to COVID-19, but they could not get tested to confirm their concerns. Future studies should continue to explore the utility of infoveillance approaches to estimate COVID-19 disease severity, extending research to more text and audio databases

[2] https://www.cdc.gov/coronavirus/2019-ncov/hcp/clinical-guidance-management-patients.html

[3] (https://sarkerlab.org/covid_sm_data_bundle/)

and more refined ML models.

## 3. METHODOLOGY

### 3.1. Deep Learning

Deep learning (DL) is a subfield of Machine Learning with algorithms inspired by the brain's structure and pattern recognition. It uses many stacks of "*neurons*" with linear and non-linear transformations to process information. The word "*deep*" in "*deep learning*" is due to the high depth that the networks can achieve [19].

DL techniques are a valuable tool for natural language processing (NLP). In particular, recurrent neural networks (RNN) were the most successful approach until a few years ago, reaching over 95% accuracy in classification datasets such as 20News e Fudan, though they performed with less than 50% in other datasets such as ACL and SST [20].

The main characteristic of RNNs is that they pass information about their current state on to the following calculation at each step. This way, information about the previous steps can be used for output predictions. Despite that, this sequential flow of information creates a bottleneck for speeding up the calculations, since dependence on previous steps makes parallelization hard [19].

In the current state of the art, other NLP tasks based on DL techniques include translation [7, 21], text summarization [9], and sentence similarity [22].

### 3.2. Transformers

The Transformer framework is a DL method introduced in 2017 for NLP tasks. It appeared as a state of the art technique for translation task on WMT 2014 English-to-German (EN-DE) and English-to-French (EN-FR), with BLEU scores of 28.4 (EN-DE) and 41.8 (EN-FR). The main idea behind Transformers is the use of a learnable attention mechanism to choose which parts of the sentence are more relevant for the meaning of each word [7].

A sentence needs to be tokenized before being processed by the attention mechanism. In other words, the sentence is split into words or sub-words, and each word (or sub-word) is substituted by a corresponding token (a number). Each of these tokens has a different embedding representation, a numerical vector of some fixed dimension to represent the token.

The Transformer's attention process happens in the so-called multi-head attention layer, formed by several self-attention sublayers. The objective of this layer is to transform each embedding representation into a new embedding that is a linear combination of all the sentence's embeddings.

In each self-attention layer, also called a *head*, the input embeddings are multiplied by three matrices with learnable weights named $W_Q$, $W_K$, and $W_V$, resulting in three vectors $q_i$, $k_i$, and $v_i$, respectively, for each input token. These vectors are known as the queries, the keys, and the values, respectively. For example, the attention mechanism for $Token1$ involves multiplying $q_1$, the query vector of $Token1$, by each of the $k_i$ vectors, the key vectors of each token, resulting in a score for each token. These scores are regularized, divided by 8 in our example, and fed to a softmax function, resulting in a number between 0 and 1 for each of them. Then, each of these numbers is multiplied by its token value vector $v_i$. The sum of these weighted value vectors is the $z_1$ vector, the output of $Token1$. Then, the $z_1$ vectors of each head are concatenated together and multiplied by $W_O$, giving the output of $Token1$, a new embedding for it. The calculations inside one self-attention layer can be visualized in Figure 1. The idea behind this process is simple: the closer the key and the query representations are, the higher the score and the more attention is paid to it.

Particularly, Transformers are recommended for our task since this method uses attention mechanisms on the context of the whole sentence and has achieved state-of-the-art results in text classification tasks [23, 24].

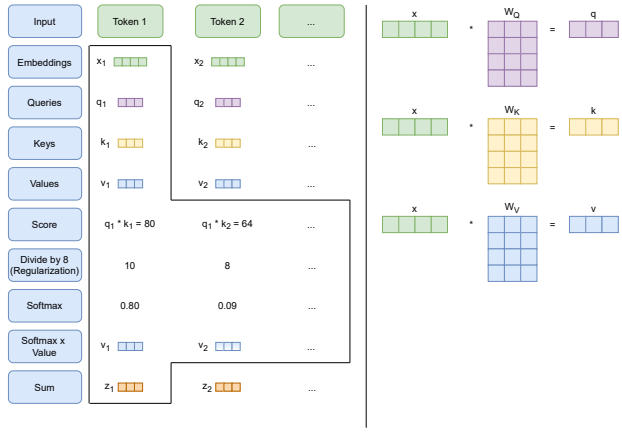


Figure 1: Left: example of how to calculate the forward of the first token in a self-attention layer. Right: scheme showing the origin of the $q_j$, $k_j$ and $v_j$ vectors.

### 3.3. BERT

BERT is a Transformer network pre-trained using the BooksCorpus and English Wikipedia [6]. BERT-based models have been achieving state of the art results in multiple classification tasks [8, 25, 26].

BERT is an encoder network, meaning it is based on several stacks of encoder layers. Each encoder used comprises a multi-head attention layer, as described in the previous section, and a Position-Wise Feed Forward Network (PWFFN). Skip connections, dropouts (with p=0.1), and layer normalization in the encoder are standard practices for training deep neural networks more effectively.

The PWFFN is a dense network that processes each embedding independently instead of processing the sequence of embeddings as a vector. It comprises a Dense (Linear) sublayer with an output dimensionality of 3072 and another with an output dimensionality of 768. The stack of encoders is called the body of the model. BERT-Base, the body of the model used in the present work, comprises 12 Encoder layers, summing up to 110 million parameters. A BERT-Base network is shown in Figure 2.

After all the encoder layers, the network ends with a dense layer with output dimensionality of 768 and tanh as the activation function, a dropout layer of probability of 0.1, and a final dense layer with output dimensionality of 14 (*i.e.*, the number of symptoms to be identified in our case of study). This is called the head of the model.

BERTimbau is a BERT model body pre-trained on brWaC (Brazilian Web as Corpus), the most extensive open Brazilian Portuguese corpus when BERTimbau was released [27]. It reached state-of-the-art results in the ASSIN2 dataset, with 0.52 in Pearson's correlation for Sentence Textual Similarity and an F1-score of 90.0 for Recognizing Textual Entailment. It also reached state-of-the-art results in the MiniHAREM dataset with an F1-score of 83.7 (5 classes) and 78.5 (10 classes) for Named Entity Recognition. The previous state-of-the-art for all these datasets was BERT Multilingual; a BERT model pre-trained on 104 languages with the largest Wikipedia, including Portuguese. These results suggest that a BERT network trained exclusively in Portuguese can outperform a multilingual BERT.

Other models tested on our dataset include BERT Base Uncased, BERT Multilingual, and GPT-2. However, they were all overcome by BERTimbau's results, including accuracy, precision, and AUC-ROC, motivating the use of BERTimbau in the dataset described in section 4.

### 3.4. Cross Validation

Before the cross validation framework, the data was shuffled and split into two groups: training and testing (20% of the data).

The cross validation framework is commonly used among data scientists due to the stability of results and the precision for error evaluation. It was performed with 5 (five) folds on the training group. In other words, the training group was shuffled and split among 5 (five) groups and were created 5 (five) independent models with the same architecture and initial weights. Each model used a different set for validation, using all the other ones for training, as shown in Figure 3.

Finally, all the 5 (five) different trained models evaluated the testing group, since it is a set of data never processed by the networks trained. It was used with the median and the interval between 15.87% and 84.13% percentiles to evaluate all the metrics in the test dataset, as described in section 6.

### 3.5. Local Interpretable Model-Agnostic Explanations (LIME)

The Local Interpretable Model-Agnostic Explanations (LIME) framework was used as a strategy for interpreting individual decisions of the model instead of trying to find a global reasoning for the model. LIME does that by running the model several times, showing different parts of the same sentence in the text data. Thereat, LIME can identify the most important words used by the model for classification and returns an estimation of the faithfulness about explanations of each sentence [28].

LIME supports finding biases in the classification based on marked words and the calculated faithfulness, which allows improvements in the model adequacy to the precise information of the text (in our case, the COVID symptom Tweets).

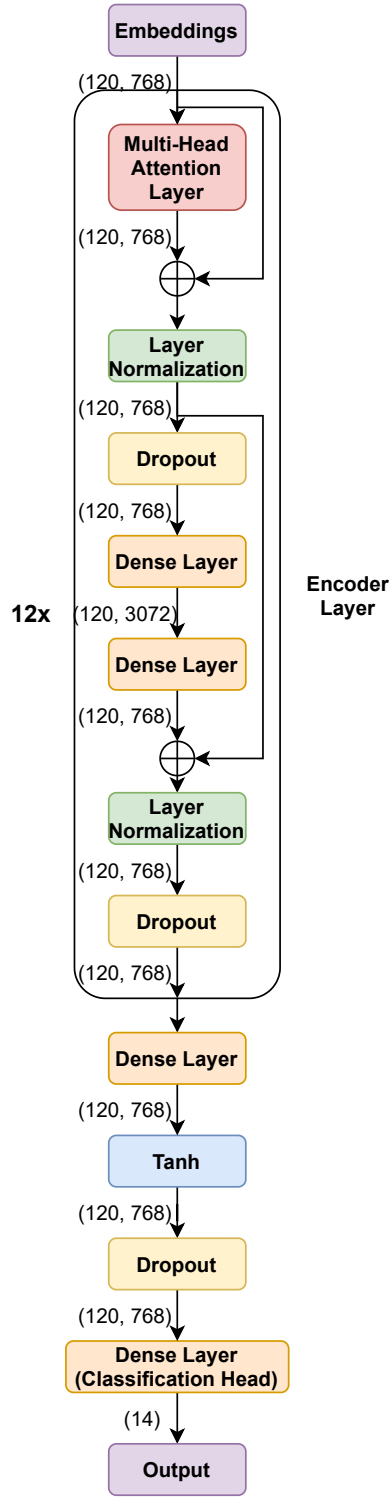


Figure 2: BERT for a classification task with 14 classes. The numbers after each layer are the shape of the output of the layer.

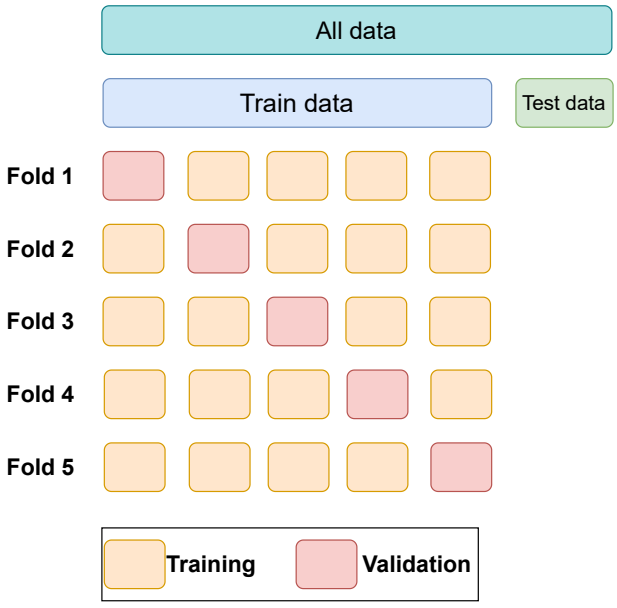


Figure 3: Cross-validation scheme.

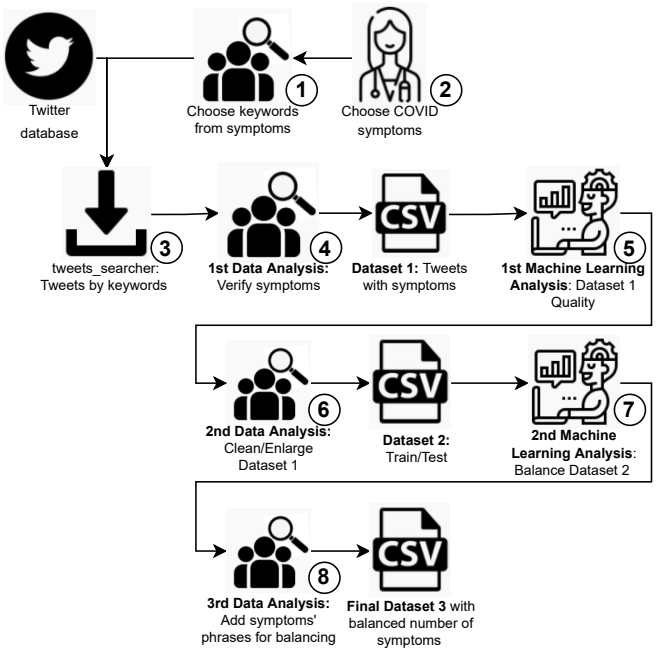


Figure 4: Workflow conceptual view for the database construction.

## 4. COVID-19 DATABASE CONSTRUCTION

In this project, we perform data analysis and topic modeling on the tweets about COVID-19 in Brazil from 1 February 2021 to 31 April 2021. A scientific workflow was modeled for extracting and processing tweets to be analyzed with deep learning. It aims to provide a comprehensive COVID-19 Twitter data set containing all tweets of high quality.

The workflow is formed by 8 (eight) main activities, with each activity representing an algorithm, software, or data analysis (Figure 4). Activities 1 and 2 are performed for data acquisition by choosing COVID symptoms and keywords. Activity 3 executes TweetSearcher for extracting COVID data information from Twitter based on the COVID symptoms and keywords from previous activities. Activities 4, 6, and 8 validate results by analytic data specialists. Activities 5 and 7 execute AI tools using deep learning, NLP, or neural networks consuming results from activities 4 and 6. Finally, activity 8 validates results and generates a final Dataset in CSV format.

The workflow execution is conducted by human-machine interaction between data analytic, health, and deep learning specialists. Also, the workflow can be repeatedly executed until it reaches a consensus data quality. More details about activities are presented in the next sections.

### 4.1. Data Selection: COVID-19 Keywords and Symptoms

The set of keywords related to the coronavirus is formed by: *covid*, *corona*, *c19*, including equivalents hashtag (*e.g.*, *#covid19*). As those keywords retrieved vast numbers of tweets using TweetSearcher, a filter was included to keep tweets containing at least one of the chosen symptoms. In addition, a second filter was included to remove retweets and images.

The set of chosen symptoms are presented in Portuguese in Table I. They are: calafrio *(chill)*, congestão nasal *(nasal congestion)*, coriza *(runny nose)*, diarreia *(diarrhea)*, dor de cabeça *(headache)*, dor muscular *(muscle pain)*, dor de garganta *(sore throat)*, febre *(fever)*, mal estar *(feeling sick)*, perda de olfato *(loss of smell)*, perda de paladar *(loss of taste)*, sonolência *(drowsiness)*, tosse *(cough)*, enjoo *(nausea)*, cansanço *(tiredness)*.

### 4.2. Twitter Acquisition: TweetSearcher

The third activity executes TweetSearcher to extract tweets from Twitter posts, which are stored in our local COVID database. TweetSearcher is a Python script that communicates with the Twitter API through the Tweepy library[4]. The script performs a cross-search between symptoms and keywords related to COVID-19, as shown in Figure 5. The Twitter API can filter no related words using blocklists and even advanced query parameters. We filtered retweets and tweets with images. Finally, the script creates a file using the JavaScript Object Notation Lines (JSONL) format to store the obtained tweets in a Comma Separated Values (CSV) format. TweetSearcher is available in the GitHub repository [5].

### 4.3. First Data Analysis: Verifying COVID-19 Symptoms

At least three data analysts manually reviewed tweets to identify the true self-reports. Multiple filter layers (Activities

[4] (https://www.tweepy.org/)
[5] https://github.com/rafaelstjf/tweets_searcher

| Symptoms in English | Symptoms in Portuguese | No. of Symptoms in Dataset 1 | No. of Symptoms in Dataset 3 |
|---|---|---|---|
| cough | *tosse* | 366 | 392 |
| nausea | *enjoo* | 59 | 106 |
| runny nose | *coriza* | 57 | 143 |
| headache | *dor de cabeça* | 56 | 154 |
| diarrhea | *diarreia* | 35 | 105 |
| muscle pain | *dor muscular* | 27 | 116 |
| fever | *febre* | 26 | 124 |
| loss of taste | *perda de paladar* | 18 | 105 |
| tiredness | *cansaço* | 17 | 116 |
| sore throat | *dor de garganta* | 15 | 136 |
| chill | *calafrio* | 13 | 102 |
| loss of smell | *perda de olfato* | 13 | 114 |
| nasal congestion | *congestão nasal* | 8 | 139 |
| drowsiness | *sonolência* | 7 | 101 |
| feeling sick | *mal estar* | 1 | - |
| **Total** | | **735 (in 556 tweets)** | **1953 (in 907 tweets)** |

Table I: Number of symptoms presented in the Dataset 1 (Initial) and Dataset 3 (Final).

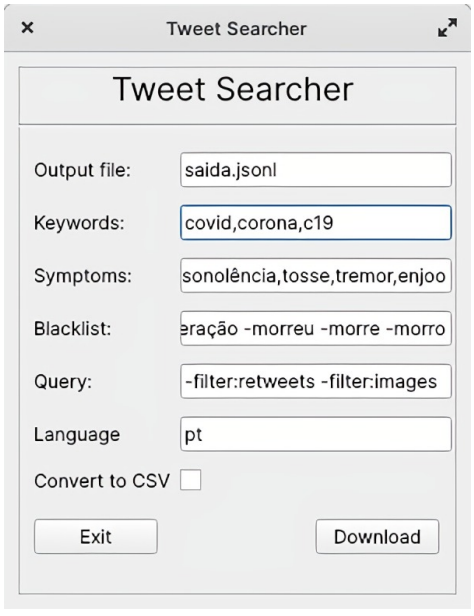


Figure 5: TweetSearcher Interface

4, 6, and 8) gave us a tractable set of potential COVID-19-positive symptoms (a few hundred) reported by users.

Activity 4 verifies the presence of COVID-19 symptoms in tweets. We further removed users from our COVID-19 database if:

1. Their tweet reports were considered to be fake;
2. Were retweeted from other users;
3. Users affirm that their tests are negative despite their initial beliefs based on symptoms that led to the belief about contracting the virus.

For instance, tweets such as **COVID-19: Brasil chega a 25,4 milhões de vacinados** (*COVID-19: Brazil reaches 25.4 million vaccinated*) are discarded due to the fact that COVID-19 symptoms are not in tweet posts.

Finally, to discover users who self-reported positive COVID-19 tests, we categorized tweets in four sets based on the presence of COVID-19 symptoms:

1. Tweets with no symptoms;
2. Tweets with symptoms and a confirmed positive test (this is our COVID-19-positive set);
3. Tweets with symptoms and a confirmed negative test;
4. Tweets with symptoms and no test.

### 4.4. Second Data Analysis: Verifying Symptoms in Tweets

Activity 6 identifies and excludes tweets (produced by ML in Activity 5) that present unusual or meandering words derived from symptoms. For instance, the use of the Portuguese word **sonoridade**, which means *sonority*, was excluded since it has no relationship with the COVID symptom **sonolência** (*drowsiness*). Note that both words sono-ridade and sono-lência have the same root, with different meanings. However, not all the words with this problem were removed, as the results' analysis will clarify.

### 4.5. Second Data Analysis: Balancing Frequency of Symptoms in Tweets

During activity 8, more tweets, and consequently more symptoms, were added from Twitter until the frequency distribution of each symptom in the database was established. Finally, our initial Dataset (1) is comprised of 556 tweets containing 735 symptoms, which was increased in the final Dataset (3) composed of 907 tweets containing 1.953 symptoms, as presented in Table II.

Since there was only one example of the symptom *feeling sick*, it was excluded from our analyses. Activities 6 and 8 were repeatedly executed until the frequency of all symptoms was balanced, and ML analyses were executed with accuracy.

| | Dataset 1 | Dataset 3 |
|---|---|---|
| **Tweets with symptoms** | 556 | 907 |
| **Tweets without symptoms** | 3419 | 3414 |
| **Total** | 3975 | 4321 |

Table II: Comparison between the data distributions of the initial and the final datasets.

## 5. MACHINE LEARNING PROCESS TO ANALYZE THE COVID-19 DATABASE

The BERT network was used to identify symptoms in sentences from the Dataset (3) obtained from the previous section. Figure 6 presents a schematic describing the steps to execute the BERT network. The steps are described as follows.

Activity 1 aims to clean data using regular expressions (RegEx). Sentences collected by Twitter's API must be processed to clean and extract useful information. For instance, links, usernames (starting with "@"), emojis, hashtags, and "*RT*" were removed due to the fact that this type of text is not expected to appear in real-life anamnesis sentences. Abbreviations such as "*vc*" instead of "*você*" were not removed due to the transformer networks' ability to process those features, maintaining a grammatical coherence. Misspelling is present, but it was not removed as it appears in authentic contexts.

During activity 2, BERTimbau's tokenizer was used to tokenize each sentence. This process separates each word in the sentence and checks its presence in the dictionary of the tokenizer. If the word is contained in the dictionary, it is substituted by a corresponding token (a number). Otherwise, the word is split into sub-words, which are substituted by their corresponding tokens. There are also special tokens to indicate the beginning and the end of tweets. The final list of tokens must have 120 tokens. If it doesn't, the list is completed with *zero* tokens until a list with 120 tokens represents each sentence. In other words, the list is padded to 120 tokens. Although BERT can handle up to 512 tokens *per* data, we limited it to 120 tokens due to the memory size of our GPUs. Other than that, as tweets have 280 characters as a limit, the 120 tokens used are enough to represent all the tweets.

Activity 3 is represented by Token Embedding and Positional Embedding. Embedding is the process of representing some information using a vector. If the information is a token, it is called Token Embedding (TE); if it is a position, it is called Positional Embedding (PE). Each token has a different embedding representation, provided by BERTimbau, *i.e.*, each token is mapped to a pre-trained vector of 512 dimensions to represent the "*meaning*" of the token. The TE is added to a PE to indicate the position of each token on the tweet. PE represents the information about the position of each token, allowing BERT to differentiate the distance between tokens in the sentence.

After the Embedding process, the data is split by Cross Validation and fed into the BERT network (activity 4). The output is a vector with the dimensionality of the number of symptoms. Results are processed by a sigmoid function (activity 5), returning a number between 0 and 1, which can be interpreted as the probability of the presence of one specific

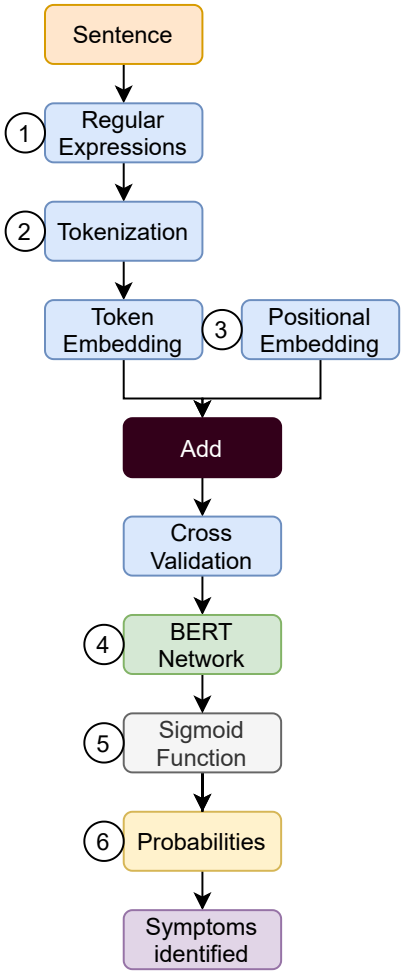


Figure 6: Diagram describing the steps every sentence of the dataset 3 was submitted to to identify symptoms.

symptom in a sentence (activity 6). It is necessary to choose a minimum probability to consider the symptom present; that is called the threshold. The threshold chosen was probability $>= 0.5$. Finally, the process generates a list of symptoms identified in each sentence.

## 6. RESULTS AND ANALYZES

### 6.1. Model Training

The model training converges, since the training and validation losses decrease to a plateau, as presented in Figure 7. The median of the validation loss across the folds was used to select the best epoch to use on the test dataset.

### 6.2. Model performance

All the described results were evaluated in the test dataset generated by the cross validation framework applied on dataset (3), as described in Section 4. The network reaches high AUC-ROC and precision across all the symptoms in the test dataset, as observed in Figure 8. These results indicate

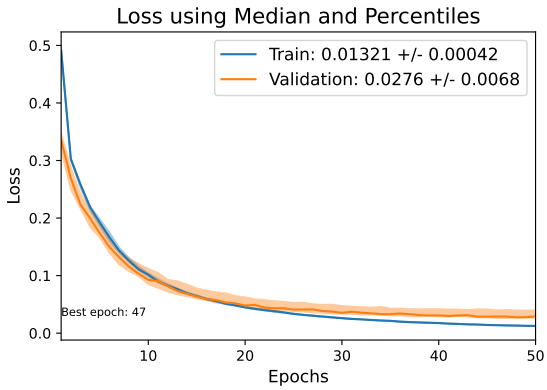


Figure 7: Final training and validation losses for BERTimbau model.

| | AUC-ROC | Precision |
|---|---|---|
| Chill | 0.99388 ± 0.00304 | 0.944 ± 0.010 |
| Nasal congestion | 0.99755 ± 0.00084 | 0.927 ± 0.007 |
| Runny nose | 0.99992 ± 0.00003 | 0.950 ± 0.005 |
| Diarrhea | 0.99924 ± 0.00065 | 0.905 ± .008 |
| Headache | 0.99871 ± 0.00040 | 0.910 ± 0.014 |
| Muscle pain | 0.99989 ± 0.00003 | 0.964 ± 0.024 |
| Sore throat | 0.99710 ± 0.00029 | 0.862 ± 0.011 |
| Fever | 0.99781 ± 0.00049 | 0.969 ± 0.001 |
| Loss of smell | 0.99791 ± 0.00032 | 0.842 ± 0.020 |
| Loss of taste | 0.99982 ± 0.00003 | 0.882 ± 0.030 |
| Drowsiness | 0.95886 ± 0.00019 | 0.947 ± 0.010 |
| Cough | 0.99821 ± 0.00012 | 0.951 ± 0.002 |
| Nausea | 0.99916 ± 0.00034 | 0.913 ± 0.007 |
| Tiredness | 0.99988 ± 0.00017 | 0.960 ± 0.001 |

Table III: Median and standard deviation of AUC-ROC and Precision by symptom. All the metrics were measured for the test dataset.

that the model is highly efficient in differentiating the presence of symptoms in tweets and generates few false positives. Furthermore, Table III shows that the results across the folds of the cross validation are highly stable for all the symptoms.

Since some symptoms can be reported by patients using several different synonyms, slightly lower values of precision or accuracy of results are expected, *e.g.*, *nasal congestion*, *headache*, or *sore throat*. The network learned the most common ways of describing it, but a few rare examples were not generalized, leading to a slightly lower precision. The main problem refers to patients using different words to express their feelings. These differences can be regional or even dependant on age, reinforcing the necessity to collect the most complete and diverse dataset possible for training.

A different problem happened with *loss of smell* and *loss of taste*: very few sentences had one of these symptoms without the presence of the other one. The network created a correlation between the symptoms, leading to a decrease in both precisions.

The symptom *drowsiness* had the lowest AUC-ROC, probably due to the presence in the dataset of words that are similarly written but have very different meanings, such as *sonoro (sonorous)* and *sonolência (drowsiness)*. As its precision can show us, the network was able to identify the actual cases of *drowsiness*, but the AUC-ROC indicates that the assurance of the final result could still be higher. Although that problem could be minimized by creating a more specialized dictionary for the task, the problem probably exists because the dataset used for training is quite different from the context of the imagined application. In fact, in an anamnesis context, the presence of words that are so different to the medical context isn't expected.

Likewise, the emergence of similar words with different meanings should be expected in the context of audio recordings of human language. Furthermore, similar-sounding words will likely confuse a network attempting to identify them. Our research group is currently developing an audio anamnesis network in Portuguese and tackling these challenges.

### 6.3. LIME results

Using LIME to analyze wrong predictions can be a source of future improvement, as observed in Figure 9. The examples presented in the figure suggest that the model has biases inherited from the data. In (a), despite the presence of the word *entupido*, which LIME tags as highly important, the model fails to return the presence of the symptom *congestão nasal*. That is probably a consequence of a lack of more examples connecting that word and symptom in our dataset.

In (b), the problem is different since the model was not trained with many sentences that described the absence of symptoms. Therefore, all the words describing symptoms are used to indicate the presence of the condition, ignoring situations that indicate its absence. Considering the objective of this network, it is a fundamental limitation. A possible way to minimize such a problem in a real-life application would be an alert for the users never to describe a symptom they are not experiencing, restricting themselves to the ones they could identify.

In (c), we have a correlation problem because the model uses a description of "$losso f taste$" to indicate "$losso f smell$". A look in the dataset is beneficial to identify that, in almost all the cases, these two symptoms emerge together, confusing the model when it has to analyze a solo case.

LIME provides a tool to understand how a network identifies symptoms and pinpoints situations where the network lacks training data. A prediction model can fail in cases under-represented in the training data. LIME gave two main insights using only these three examples and others similar to these. Firstly, the necessity of more data so the model can generalize its ability to identify symptoms. More examples of alternative ways of describing the same symptoms and new sentences that describe *loss of taste* or *loss of smell* separately would be necessary to do so, as indicated by Figure 9 (a) and (c). Secondly, it is essential to ensure that the model is not used with a negative description of a symptom. Despite this limitation, the network helps track patients at risk for COVID-19 since it involves false positives rather than false negatives.

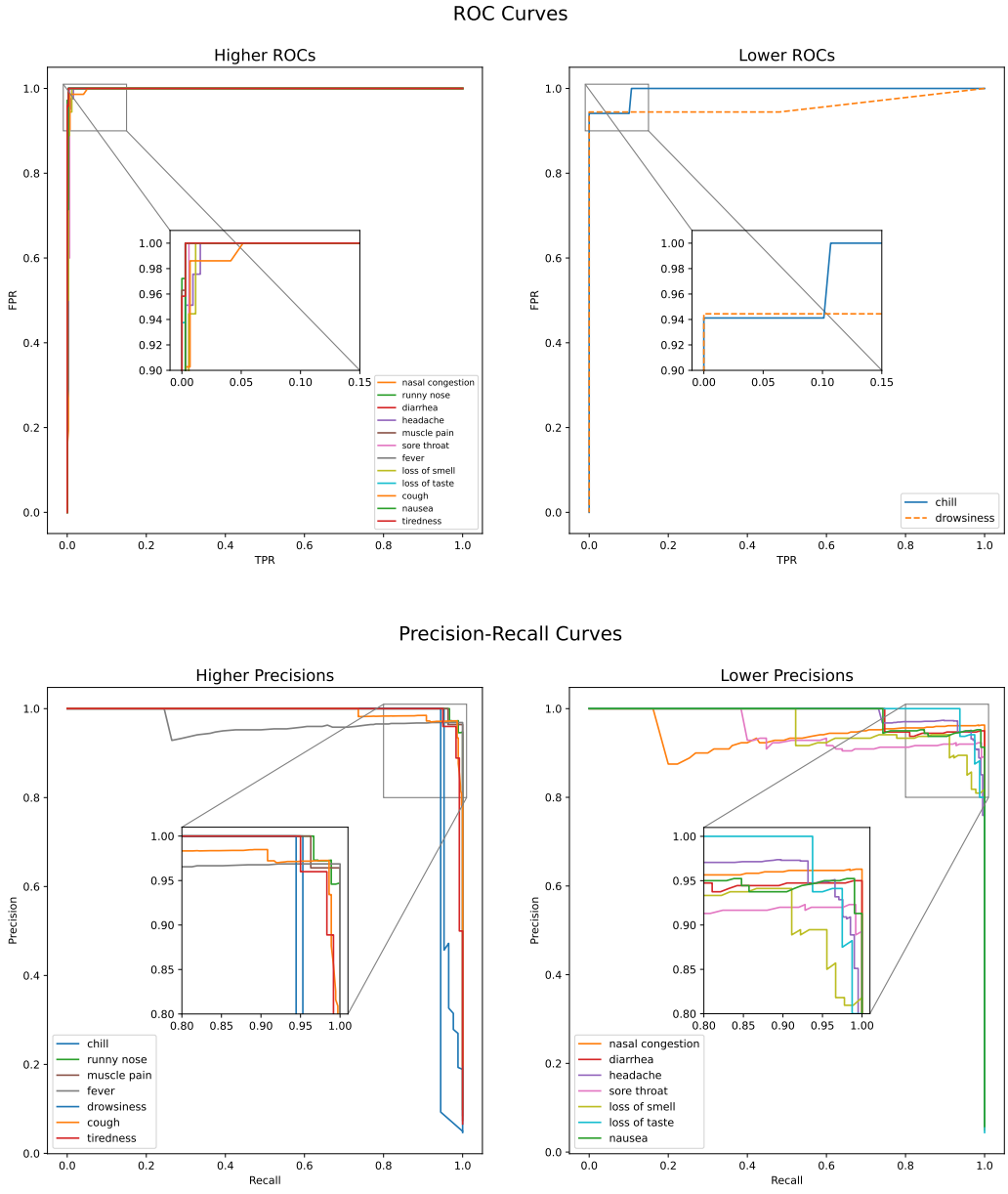


Figure 8: Final ROC and Precision-Recall curves for each symptom of the BERTimbau model.

## 7. DISCUSSION AND FINAL REMARKS

This work demonstrates the possibility of using artificial intelligence to identify symptoms in Portuguese texts and the high-quality results of using modern Natural Language Processing techniques, such as Transformer Networks. However, it also points out that a more diverse dataset would be necessary to develop a more accurate model that can account for a broader range of COVID-19 symptoms.

Several challenges at processing real biological or clinical databases must be adapted in computational experiments using ML-based techniques. For example, data analysts and medical specialists can treat data to diminish noise and regularize the symptoms' distribution. However, a systematic error can be introduced by this method. Therefore, a large-scale experiment, including more complex and automatically processed data in a different time cohort, is needed to further develop the methodology.

To the best of our knowledge, this is the first study to have utilized Twitter to curate COVID-19's symptoms in Portuguese posted by users analyzed by Transformer techniques. Future studies can be done to analyze the generalization ability of the proposed method in a dataset containing reports from patients in clinical settings.

### Acknowledgment

The authors wish to acknowledge Prontlife, that provided the problem proposal and the grant

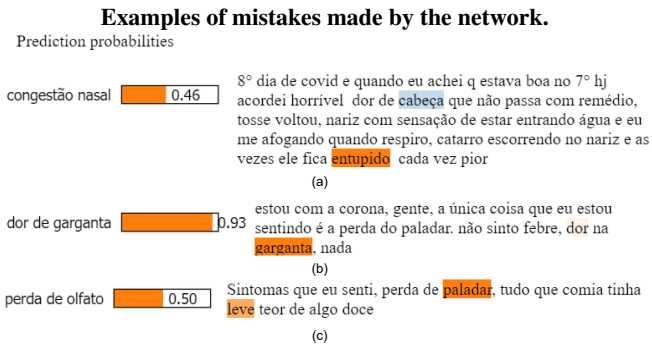


Figure 9: (a): false negative case; (b): false positive case; (c): false positive case.

that made this research possible. In addition, the authors acknowledge the Brazilian Center for Physical Research (CBPF/MCTI, Brazil) for providing the multi GPU Sci-Mind machines.

## Bibliography

[1] A. Reyner, W. Tjiptomongsoguno, A. Chen, H. Sanyoto, E. Irwansyah, and B. Kanigoro, Medical Chatbot Techniques: A Review (2020), pp. 1–11, ISBN 978-3-030-63321-9.

[2] L. Athota, V. K. Shukla, N. Pandey, and A. Rana, Chatbot for Healthcare System Using Artificial Intelligence (2020).

[3] W.-j. Guan, Z.-y. Ni, Y. Hu, W.-h. Liang, C.-q. Ou, J.-x. He, L. Liu, H. Shan, C.-l. Lei, D. S. Hui, et al., New England Journal of Medicine **382**, 1708 (2020), https://doi.org/10.1056/NEJMoa2002032, URL https://doi.org/10.1056/NEJMoa2002032.

[4] Y. Bai, L. Yao, T. Wei, F. Tian, D.-Y. Jin, L. Chen, and M. Wang, JAMA **323**, 1406 (2020), ISSN 0098-7484, https://jamanetwork.com/journals/jama/articlepdf/2762028/jama_bai_2020_ld_200013.pdf, URL https://doi.org/10.1001/jama.2020.2565.

[5] Q. Li, X. Guan, P. Wu, X. Wang, L. Zhou, Y. Tong, R. Ren, K. S. Leung, E. H. Lau, J. Y. Wong, et al., New England Journal of Medicine **382**, 1199 (2020), pMID: 31995857, https://doi.org/10.1056/NEJMoa2001316, URL https://doi.org/10.1056/NEJMoa2001316.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova (2019), 1810.04805.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin (2017), 1706.03762.

[8] A. Adhikari, A. Ram, R. Tang, and J. Lin, CoRR **abs/1904.08398** (2019), 1904.08398, URL http://arxiv.org/abs/1904.08398.

[9] Y. Liu and M. Lapata, CoRR **abs/1908.08345** (2019), 1908.08345, URL http://arxiv.org/abs/1908.08345.

[10] M. A. Khan, N. Hussain, A. Majid, M. Alhaisoni, B. Syed Ahmad Chan, S. Kadry, Y. Nam, and Z. Yu-Dong, Computers, Materials, & Continua pp. 2923–2938 (2021).

[11] C. Shorten, T. M. Khoshgoftaar, and B. Furht, Journal of big Data **8**, 1 (2021).

[12] T. B. Alakus and I. Turkoglu, Chaos, Solitons & Fractals **140**, 110120 (2020).

[13] A. Sarker, S. Lakamana, W. Hogg-Bremer, A. Xie, M. A. Al-Garadi, and Y.-C. Yang, Journal of the American Medical Informatics Association **27**, 1310 (2020), ISSN 1527-974X, https://academic.oup.com/jamia/article-pdf/27/8/1310/34153333/ocaa116.pdf, URL https://doi.org/10.1093/jamia/ocaa116.

[14] D. Kumar, N. Kumar, and S. Mishra, NLP@NISER: Classification of COVID19 tweets containing symptoms (2021), URL https://aclanthology.org/2021.smm4h-1.19.

[15] T. e. a. Nadarzynski1, Digital Health **5**, 1 (2019).

[16] A. Valdes, J. Lopez, and M. Montes, in Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task (Association for Computational Linguistics, Mexico City, Mexico, 2021), pp. 65–68, URL https://aclanthology.org/2021.smm4h-1.10.

[17] Y. Luo, L. Pereira, and K. Ichiro, in Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task (Association for Computational Linguistics, Mexico City, Mexico, 2021), pp. 123–125, URL https://aclanthology.org/2021.smm4h-1.25.

[18] T. Mackey, V. Purushothaman, J. Li, N. Shah, M. Nali, C. Bardier, B. Liang, M. Cai, and R. Cuomo, JMIR Public Health Surveill **6**, e19509 (2020), ISSN 2369-2960, URL http://publichealth.jmir.org/2020/2/e19509/.

[19] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning (MIT press, 2016).

[20] S. Lai, L. Xu, K. Liu, and J. Zhao, Recurrent convolutional neural networks for text classification (2015), URL https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9745/9552.

[21] X. Liu, K. Duh, L. Liu, and J. Gao, CoRR **abs/2008.07772** (2020), 2008.07772, URL https://arxiv.org/abs/2008.07772.

[22] I. Lopez-Gazpio, M. Maritxalar, M. Lapata, and E. Agirre, Expert Systems with Applications **132**, 1 (2019), ISSN 0957-4174, URL https://www.sciencedirect.com/science/article/pii/S0957417419302842.

[23] S. González-Carvajal and E. C. Garrido-Merchán, CoRR **abs/2005.13012** (2020), 2005.13012, URL https://arxiv.org/abs/2005.13012.

[24] W. Chang, H. Yu, K. Zhong, Y. Yang, and I. S. Dhillon, CoRR **abs/1905.02331** (2019), 1905.02331, URL http://arxiv.org/abs/1905.02331.

[25] Z. Gao, A. Feng, X. Song, and X. Wu, IEEE Access **7**, 154290 (2019).

[26] H. T. Madabushi, E. Kochkina, and M. Castelle, CoRR **abs/2003.11563** (2020), 2003.11563, URL https://arxiv.org/abs/2003.11563.

[27] F. Souza, R. Nogueira, and R. Lotufo, arXiv preprint arXiv:1909.10649 (2019), URL http://arxiv.org/abs/1909.10649.

[28] M. T. Ribeiro, S. Singh, and C. Guestrin, "why should i trust you?": Explaining the predictions of any classifier (2016), 1602.04938.

Notas Técnicas é uma publicação de trabalhos técnicos relevantes, das diferentes áreas da física e afins, e áreas interdisciplinares tais como: Química, Computação, Matemática Aplicada, Biblioteconomia, Eletrônica e Mecânica entre outras.

Cópias desta publicação podem ser obtidas diretamente na página web http://revistas.cbpf.br/index.php/nt ou por correspondência ao:

Centro Brasileiro de Pesquisas Físicas
Área de Publicações
Rua Dr. Xavier Sigaud, 150 – 4º andar
22290-180 – Rio de Janeiro, RJ
Brasil
E-mail: alinecd@cbpf.br/valeria@cbpf.br
http://portal.cbpf.br/publicacoes-do-cbpf

Notas Técnicas is a publication of relevant technical papers, from different areas of physics and related fields, and interdisciplinary areas such as Chemistry, Computer Science, Applied Mathematics, Library Science, Electronics and Mechanical Engineering among others.

Copies of these reports can be downloaded directly from the website http://notastecnicas.cbpf.br or requested by regular mail to:

Centro Brasileiro de Pesquisas Físicas
Área de Publicações
Rua Dr. Xavier Sigaud, 150 – 4º andar
22290-180 – Rio de Janeiro, RJ
Brazil
E-mail: alinecd@cbpf.br/valeria@cbpf.br
http://portal.cbpf.br/publicacoes-do-cbpf