# Language-focused Deepfake Detection Using Phonemes, Mouth Movements, and Video Features

Jonas Krause
*Computational Intelligence Laboratory*
*UTFPR*
Curitiba, Brazil
jonaskrause@utfpr.edu.br

Andrei de Souza Inacio
*Federal Institute of Santa Catarina*
*IFSC*
Gaspar, Brazil
andrei.inacio@ifsc.edu.br

Heitor Silvério Lopes
*Computational Intelligence Laboratory*
*UTFPR*
Curitiba, Brazil
hslopes@utfpr.edu.br

***Abstract*—The potential implications of Artificial Intelligence (AI) and Deep Learning (DL) algorithms in generating highly realistic deepfake videos have raised concerns regarding the reliability of our human senses. In response to this challenge, we propose a deepfake detection system based on phonemes, the transcribed text, associated mouth movements, and video-extracted features. As a proof-of-concept, we develop a deepfake detection system specifically designed for the Portuguese language, employing three presidential candidates from the 2022 Brazilian elections. Additionally, we introduce a unique dataset comprising real and fake videos involving these three individuals and deliberately blending their identities. The extracted features consolidate relevant attributes, which we utilized to train multiple classification algorithms. Notably, our computational models demonstrate satisfactory performance when authenticating or detecting fake videos containing at least one of the trained phonemes from the Portuguese language. Hence, we conclude that deepfake detection is feasible, primarily due to the absence of natural expressions, particularly in non-English language deepfake videos. Furthermore, developing individual-guided deepfake detection systems may facilitate the authentication of videos featuring celebrities or politicians during future online events.**

***Index Terms*—Deepfake, Language-focused, Phoneme-based.**

## I. Introduction

Deepfakes, which involve the manipulation of videos, images, and audio to create convincing yet fabricated content, have gained significant attention since their popularization in 2017. The initial focus of deepfake videos revolved around inserting celebrity faces into pornographic movies, exemplified by the manipulated video featuring "Wonder Woman" actress Gal Gadot. However, the emergence of deepfakes featuring prominent individuals such as North American Speaker of the House Nancy Pelosi, Facebook CEO Mark Zuckerberg, and "Game of Thrones" character Jon Snow in 2019 highlighted the urgent need for effective detection methods and regulatory measures. Consequently, computer scientists have dedicated considerable efforts towards developing deepfake detection systems, aiming to assess the authenticity of manipulated images, audio, and videos.

While most deepfake videos have primarily been used for humorous content, often involving reenactments or facial transfers in music videos and memes, there is a growing concern that they will soon incorporate political messages and target public figures. Some of these seemingly humorous deepfakes have already emerged, aiming to discredit local celebrities and government officials, attaining an alarming level of realism. Consequently, the influence of deepfake political videos on numerous elections is anticipated, necessitating proactive measures to mitigate their impact. In light of this, we initiated a project that involves reviewing academic articles, creating deepfake videos in the Portuguese language, and developing countermeasures to authenticate or debunk videos and online streams featuring target individuals.

We organize the remainder of this paper as follows: Section II provides an overview of related work in deepfake detection. Section III details the methodology employed in our project, encompassing dataset creation, feature extraction, and classification methods. Section IV presents our experimental results and performance evaluation. Finally, Section V concludes the paper by summarizing the findings and discussing future directions for combating the increasing threat posed by deepfakes in a global context.

Our methodology seeks to unveil these mouth reenactment videos by analyzing specific phonemes inherent to the target language alongside the corresponding articulatory movements of the mouth during phoneme pronunciation. And we exploid the prevalence of deepfake videos derived from deep learning (DL) architectures trained on English audio and visual data. Moreover, emerging technologies are delving into the distinctive characteristics of various languages worldwide. For instance, the film industry has already harnessed deepfake technology to achieve realistic voice dubbing in alternative languages and facilitate post-production editing. The video game industry has similarly embraced this technology, employing virtual characters to impart a natural linguistic quality to their products. Likewise, educational media has capitalized on this technology by reenacting historical figures and delivering notable speeches.

Nonetheless, it is imperative to acknowledge that unscrupulous individuals are poised to exploit this technology as a weapon to manipulate public opinion and disseminate misinformation. For example, malevolent actors can assume the identity of a targeted individual, thereby gaining the trust of a family member and acquiring illicit financial or material

979-8-3503-4807-1/23/$31.00 ©2023 IEEE

gains. Additionally, the generation of compromising content featuring prominent figures may be employed for purposes of blackmail, or content manipulation could be orchestrated to influence public sentiment toward a particular political leader. This technology also harbors the potential to tamper with surveillance footage and other archival visual material, thereby introducing fabricated evidence into legal proceedings. Ultimately, these attacks may manifest as online threats, including real-time impersonation in conversations or the dissemination of falsified media. Consequently, mouth reenactment has garnered significant attention within the academic literature due to its role as one of the most substantial threats in the context of the ongoing misinformation warfare.

## II. Literature Review

Existing research in this field includes the work of Güera and Delp [1], who proposed a deepfake detection system utilizing a Recurrent Neural Network (RNN) within a temporal-aware pipeline. Their approach involved extracting frame-level features using a Convolutional Neural Network (CNN), which they used to train the RNN classifier for discerning manipulated videos. Nguyen et al. [2] employed capsule networks, a form of CNN-based models, to detect forged images and videos, encompassing various spoofs ranging from replay attacks to computer-generated content. Nhu et al. [3] focused on forensics face detection and utilized Generative Adversarial Networks (GANs) to create fake training data for their CNN-based analysis. It allowed them to generate diverse facial images with different resolutions and sizes, enabling robust face feature extraction for deepfake recognition systems.

Additionally, Ciftci et al. [4] proposed the FakeCatcher, a system designed for detecting synthetic portrait videos. Their method involved extracting biological signals from facial regions in authentic and fake portrait video pairs. Through transformations to assess spatial coherence and temporal consistency, they obtained feature sets capturing the signal characteristics, and trained a Support Vector Machine (SVM) and a CNN for classification. Moreover, Li and Lyu [5] focused on facial region signals, detecting face-warping artifacts as distinctive clues for deepfake detection. These artifacts resulted from limited-resolution deepfake image generation, subsequently warped to match the original faces in the source video. And they used state-of-the-art CNNs capable of identifying such artifacts, although manipulations like video compression could potentially obscure these clues.

Furthermore, computer science researchers have extensively employed DL algorithms to investigate specifically mouth reenactment videos, commonly known as "dubbing". These investigations have involved different approaches, and specific techniques, including one-to-one (identity to identity), many-to-one (multiple identities to a single identity), and many-to-many (multiple identities to multiple identities) deepfake creation methods. We summarized the techniques utilized for manipulating mouth movements in Table 1, where similarities in the implemented DL techniques become apparent when grouping them based on their approaches. Notably, all papers employing a one-to-one approach have used Generative Adversarial Networks (GANs). And over half of the reviewed papers employing many-to-one approaches also implemented GANs. We observed Recurrent Neural Networks (RNNs) in papers utilizing many-to-one and many-to-many approaches, as these DL architectures enhance deepfake models with temporal awareness to accommodate pose and expression variations. The listed many-to-many approaches involve the combination of multiple techniques and DL architectures. In these approaches, authors used Encoders/Decoders (EDs) to generate deepfake videos from numerous sources to multiple targets, accompanied by Gated Recurrent Units (GRUs) and RNNs for audio and identity encoding. In a recent approach, Mazaheri and Roy-Chowdhury [27] focused on detecting and localizing facial expression manipulations, using the Facial Expression Recognition (FER) system and Ensemble with Shared Representations (ESR) to identify any alterations made to facial expressions, which aids in the advancement of reliable methods for facial verification. Although DL-based approaches are achieving good results, traditional Machine Learning (ML) techniques can also obtain satisfactory performance in detecting deepfakes. And as detailed in Rana et al. [28], the ML approaches allow better understandability and interpretability of the model with reduced computational cost.

TABLE I
MOUTH MANIPULATION PAPERS AND THEIR DL ARCHITECTURES.

| Approach | Paper | Year | DL Architecture |
|---|---|---|---|
| One-To-One | [6] | 2017 | CycleGAN |
| | [7] | 2018 | RecycleGAN |
| | [8] | 2019 | RealisticFaceGAN |
| | [9] | 2020 | DeepFaceLabGAN |
| Many-To-One | [10] | 2017 | RNN+MFCC |
| | [11] | 2018 | RNN+Char2Wav |
| | [12] | 2018 | ReenactGAN |
| | [13] | 2018 | MoCoGAN |
| | [14] | 2018 | Vid2VidGAN |
| | [15] | 2018 | RNN+MFCC |
| | [16] | 2019 | RNN+CGAN |
| | [17] | 2019 | Pix2PixGAN+AdaIN |
| | [18] | 2019 | Vid2VidGAN+AdaIN |
| Many-To-Many | [19] | 2015 | RNN+MFCC |
| | [20] | 2018 | EDs+CNN |
| | [21] | 2018 | EDs+GRU |
| | [22] | 2019 | EDs+GRU+RNN |
| | [23] | 2019 | GAN+RNN+MFCC |
| | [24] | 2019 | CGAN |
| | [25] | 2019 | CNN+RNN+CGAN |
| | [26] | 2021 | EDs+GAN |
| | [28] | 2021 | ML+FeatureSelection |
| | [27] | 2022 | FER+ESR |

## III. Methodology

Our proposed methodology comprehensively addresses various deepfake creation strategies and approaches by analyzing specific phonemes in any given language. We establish a correlation between the transcription of these phonetic units and the corresponding mouth movements observed during their pronunciation. This paper introduces the Language-focused and Phoneme-based Deepfake Detection System (LPDDS),

which utilizes pre-trained computational models to detect distinctive language phonemes. These models are mathematical algorithms based on ML techniques, trained using a dataset derived from carefully selected video clips containing spoken language segments associated with specific phonemes. The LPDDS testing process, implemented in Python 3.10, involves the following steps:

1) Extraction of the video's duration, measured in seconds;
2) Transcribe audio into text, with time stamps assigned at 5-second intervals;
3) Selection of phonemes for analysis;
4) Segmentation of the corresponding 5-second sections;
5) Extraction of facial images from the segmented clips;
6) Extraction of mouth landmarks and video features;
7) Classification of data, grouped into segments of 7 frames (representing one phoneme), as either true or false.

In this methodology, we create a database of videos comprising both real and fake samples. And within each testing video, regardless of its authenticity, we localize the specific phoneme based on its transcription, focusing on 5-second segments (steps 4, 5, and 6). To test our system, we applied a swap technique to the 5-second clip with the localized phoneme, seeking trained patterns of mouth movement and video features (step 7). For training our ML models, we manually extracted mouth landmarks and video data features from small clips representing fractions of a second (00:00:00.20), corresponding to the phonemes our LPDDS aims to classify. These small clips yielded seven (7) frames when extracting face images, providing sufficient information to establish patterns of mouth landmark movement. By manually gathering data, we ensured that the training database for the ML algorithms contained representative information of the selected phoneme.

For classification, we associated the observed mouth movement and other video-related features with specific phonemes, searching for these patterns within the suggested 5-second clip(s) based on the transcription of a new video. If our computational model identifies these patterns in the indicated clip of the target video, it will authenticate it as genuine. Otherwise, if the LPDDS does not verify the video's authenticity, it will classify the target video as a deepfake. And this approach allows our system to authenticate specific segments of the target video multiple times during long speech videos.

In this study, we focus on developing the LPDDS tailored to non-English deepfake videos and targeting the Portuguese language. One of the challenges we encountered during this research was the scarcity of realistic deepfake videos in Portuguese. As previously mentioned, deepfakes often comprise humorous advertisements, commonly featuring celebrities and political figures engaged in activities such as dancing and celebrating victories or defeats. To address this challenge and facilitate the development of the LPDDS, we generated a novel dataset consisting exclusively of real and fake videos with Portuguese audio. The dataset construction commenced with videos of three prominent 2022 Brazilian presidential candidates: Jair Bolsonaro, Luís Inácio Lula da Silva, and Simone Tebet. From 21 original videos, we collected 37 clips and created 50 deepfake videos with less than 60 seconds mixing the three target individuals' faces and containing the selected phonetic unit. To create fake videos, we employed two distinct existing deepfake techniques: the *MyVoiceYourFace* and the *FaceSwap*. These techniques operate based on similar inputs, involving either a single image or multiple images, and output the projection of the provided inputs onto the target video. In the next step, the number of clipped videos of the target phoneme "ÃO" depends on how many times each pronounces it. And labeling these clips as true or false, we created a dataset with 133 phoneme-based scenes (52 real and 81 fake) for training and testing our ML classification models. To promote transparency and accessibility, we have made all the original videos, deepfakes, and phoneme-based clips available in an online GitHub[1] repository.

Using the dataset of the selected clips, we extract aligned faces from each frame and Cartesian points that represent the position of the mouth during the clip. To achieve this, we employ the OpenCV library, which specializes in detecting facial landmarks in images or frames. Specifically, we focus on extracting the points related to the mouth. In addition, we utilize Kinetics CNN architectures to extract features related to the movement of the video. These architectures, namely the Two-Stream Inflated 3D ConvNet (I3D), are based on the inflation of 2D ConvNets and build upon the work of Carreira and Zisserman [29]. By adopting these architectures, our classification models can effectively learn spatiotemporal features from videos. This approach allows us to capitalize on the well-established designs and parameters of CNNs, resulting in spatial and temporal integration for accurate classification.

Our computational model, shown in Figure 1, uses the input of a CSV file containing the extracted mouth coordinates and Kinetics values. To enhance the model's performance, we incorporate a feature selection stage, which involves a ranking process to identify the most discriminative features. Implementing a testing and scoring process, we train multiple ML algorithms using the previously processed data. Each classifier (Logistic Regression, Decision Tree, Neural Network, Random Forest, Support Vector Machine, and k-Nearest Neighbors) undergoes a five-fold cross-validation process to ensure robustness and avoid overfitting. As a result, our computational model generates a confusion matrix for each trained ML algorithm, serving as evaluation metrics to assess their performance.

## IV. Experiments

Our experiments seek to determine the most appropriate combination of feature extraction, preprocessing techniques, and ML algorithms for authenticating a selected phoneme on real and fake video datasets. In the ML classifiers, we use traditional parameters for a fair comparison between them and future optimization when compared with DL-based

[1]https://github.com/jonaskrause/DeepFake-PhonemeAO

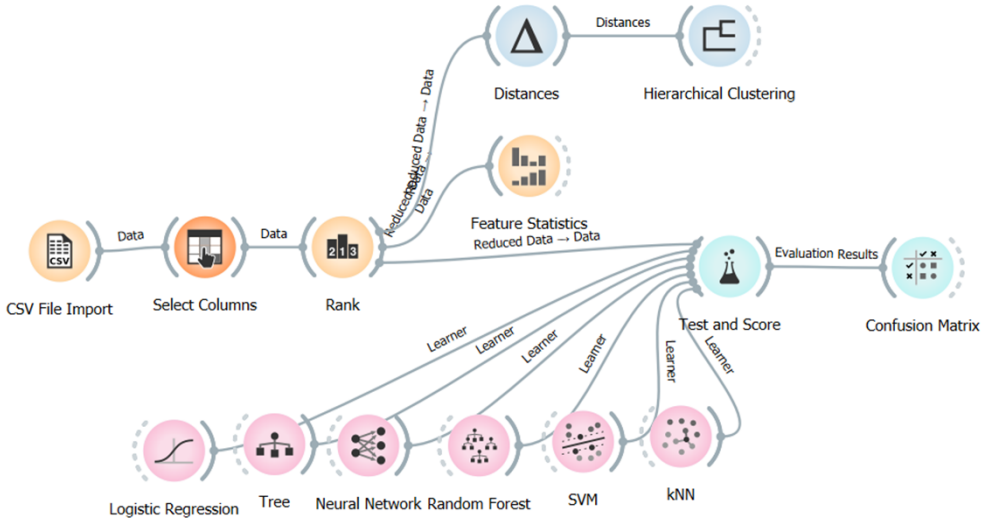


Fig. 1. LPDDS Computational Model.

approaches. We started by applying the LPDDS computational model without feature selection and using all the 280 mouth coordinates plus the 1024 Kinetics values. Table II presents the following metrics of each ML algorithm: the Area Under the Curve (AUC), the Classification Accuracy (CA), the F1 score (F1), the Precision (Prec), the Recall, and the Matthews Correlation Coefficient (MCC). And they are sorted by the highest F1 score value.

TABLE II
EVALUATION METRICS OF ML CLASSIFIERS USING ALL EXTRACTED FEATURES.

| Classifier | AUC | CA | F1 | Prec | Recall | MCC |
|---|---|---|---|---|---|---|
| SVM | 0.830 | 0.865 | **0.862** | 0.867 | 0.865 | 0.714 |
| Neural Network | 0.868 | 0.812 | 0.811 | 0.811 | 0.812 | 0.602 |
| Random Forest | 0.794 | 0.774 | 0.766 | 0.777 | 0.774 | 0.602 |
| Decision Tree | 0.720 | 0.752 | 0.750 | 0.750 | 0.752 | 0.474 |
| Logistic Reg. | 0.796 | 0.714 | 0.717 | 0.795 | 0.692 | 0.375 |
| kNN | 0.618 | 0.692 | 0.632 | 0.795 | 0.692 | 0.375 |

Figure 2 presents the output of our computational model (confusion matrix) using the ML algorithm with the highest F1 score, in this case, the SVM classifier. It details how many fake (F) and true (T) videos of the database our computational model correctly predicted versus the actual ones and, consequently, the ones it missed.

| Predic / Expec | F | T | Σ |
|---|---|---|---|
| F | 76 | 5 | 81 |
| T | 13 | 39 | 52 |
| Σ | 89 | 44 | 133 |

Fig. 2. The output of the LPPDS Computational Model (Confusion Matrix) of the highest F1 score classifier (SVM) without feature selection.

The next experiment compares the feature extraction strategies by applying the LPPDS computational model over different segments of the inputted data. We started by selecting only the mouth (X, Y) coordinates extracted from the dataset-clipped videos of the phoneme and reproduced the previous experiment. Table III presents the evaluation metrics of ML classifiers using only the mouth-corresponding extracted features. In this case, the Neural Network algorithm achieved the highest F1 score.

TABLE III
EVALUATION METRICS OF ML CLASSIFIERS USING MOUTH EXTRACTED FEATURES.

| Classifier | AUC | CA | F1 | Prec | Recall | MCC |
|---|---|---|---|---|---|---|
| Neural Network | 0.815 | 0.774 | **0.772** | 0.772 | 0.774 | 0.519 |
| Logistic Reg. | 0.786 | 0.714 | 0.717 | 0.722 | 0.714 | 0.414 |
| SVM | 0.734 | 0.714 | 0.714 | 0.714 | 0.714 | 0.400 |
| Decision Tree | 0.634 | 0.647 | 0.645 | 0.643 | 0.647 | 0.250 |
| Random Forest | 0.669 | 0.647 | 0.635 | 0.635 | 0.647 | 0.229 |
| kNN | 0.618 | 0.692 | 0.632 | 0.795 | 0.692 | 0.375 |

As the previous analysis, Figure 3 shows the confusion matrix of the ML algorithm with the highest F1 score. And in the classification of mouth coordinates, the Neural Network achieved the best result. It is noticeable when comparing Tables II and III, as well as Figures 2 and 3, that the performance of the LPDDS decreased when we reduced the number of features excluding the larger segment of the dataset (Kinetics data).

The following experiment isolates the Kinetics extracted features by applying the LPDDS computation model over this second part of the collected dataset. Therefore, we trained the classification algorithms using only video-extracted features achieving the following results. Table IV lists the evaluation metric values achieved by each ML algorithm. And, with Kinetics features, the Logistic Regression achieved the best F1 score (**0.876**) of the ML algorithms in all our experiments.

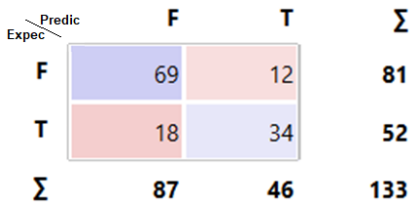


Fig. 3. The confusion matrix of the highest F1 score classifier (Neural Network) with mouth extracted features.

TABLE IV
EVALUATION METRICS OF ML CLASSIFIERS USING KINETICS EXTRACTED FEATURES.

| Classifier | AUC | CA | F1 | Prec | Recall | MCC |
|---|---|---|---|---|---|---|
| Logistic Reg. | 0.855 | 0.880 | **0.876** | 0.889 | 0.880 | 0.751 |
| SVM | 0.820 | 0.865 | 0.862 | 0.867 | 0.865 | 0.714 |
| kNN | 0.814 | 0.857 | 0.849 | 0.884 | 0.857 | 0.717 |
| Neural Network | 0.869 | 0.827 | 0.825 | 0.826 | 0.827 | 0.633 |
| Random Forest | 0.827 | 0.805 | 0.797 | 0.810 | 0.805 | 0.585 |
| Decision Tree | 0.755 | 0.774 | 0.773 | 0.772 | 0.774 | 0.521 |

Figure 4 presents the output of the most accurate computational model of the LPDDS using the experimental dataset. By analyzing the confusion matrix of this trained model, one can note that it uncovered the most deepfake phoneme expressions (79), incorrectly classifying only two of them. It also performed well when authenticating real videos (38), but it loses in this matter when compared with the LPDDS computational model using SVM and all features combined (39, in Figure 2).

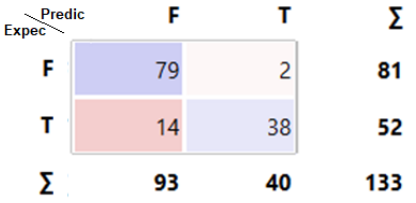


Fig. 4. The confusion matrix of the highest F1 score classifier (Logistic Regression) with Kinetics extracted features.

This initial analysis guided our research for determining the appropriate feature selection process and from which group (mouth coordinates or Kinetics data) detains the most deterministic features. So we used the Fast Correlation-Based Filter (FCBF), a feature selection method that examines the class relevance and the dependency between each feature pair. It also uses an entropy-based measure to identify redundancy due to pairwise correlations between features. Figure 5 presents the results of the FCBF analysis and points out the group of ten (10) most discriminative features when compared with another group. It is valuable to notice that these ten selected features are all part of the Kinetics group of features.

And using only these ten FCBF selected features on the LPDDS computational model, we retrained the classification to achieve the ML metrics reported on Table V. In this case, the Random Forest outperformed the other ML algorithms but

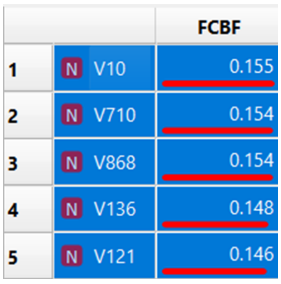


Fig. 5. FCBF indicators for each group of features.

performed similarly to the Neural Network on the previous experiment, including the identical output of the confusion matrix reported on Figure 4.

TABLE V
EVALUATION METRICS OF ML CLASSIFIERS USING FCBF SELECTED FEATURES.

| Classifier | AUC | CA | F1 | Prec | Recall | MCC |
|---|---|---|---|---|---|---|
| Random Forest | 0.852 | 0.865 | **0.862** | 0.867 | 0.865 | 0.714 |
| SVM | 0.858 | 0.857 | 0.853 | 0.863 | 0.857 | 0.700 |
| kNN | 0.770 | 0.805 | 0.785 | 0.852 | 0.805 | 0.615 |
| Logistic Reg. | 0.852 | 0.782 | 0.767 | 0.800 | 0.782 | 0.542 |
| Neural Network | 0.827 | 0.752 | 0.754 | 0.760 | 0.752 | 0.494 |
| Decision Tree | 0.763 | 0.752 | 0.751 | 0.751 | 0.752 | 0.477 |

To extend the feature extraction analysis, we research for another feature selection algorithm to select a new group of features and compare with the FCBF. We choose the ReliefF, a weight-based algorithm designed to determine the significance of predictors when the outcome variable is a multiclass categorical variable. This algorithm operates by applying penalties to predictors that produce dissimilar values for neighboring instances belonging to the same class. Contrariwise, it rewards predictors that generate distinct values for neighboring instances from different classes. In this manner, ReliefF effectively assesses the importance of predictors in distinguishing between various classes. Using the 110 most discriminative features selected by the ReliefF algorithm, we evaluate the LPDDS computational model and present the ML metric results in Table VI.

TABLE VI
EVALUATION METRICS OF ML CLASSIFIERS USING RELIEFF SELECTED FEATURES.

| Classifier | AUC | CA | F1 | Prec | Recall | MCC |
|---|---|---|---|---|---|---|
| SVM | 0.865 | 0.872 | **0.868** | 0.883 | 0.872 | 0.736 |
| Logistic Reg. | 0.873 | 0.872 | **0.868** | 0.833 | 0.872 | 0.736 |
| kNN | 0.807 | 0.857 | 0.850 | 0.877 | 0.857 | 0.711 |
| Neural Network | 0.859 | 0.827 | 0.824 | 0.827 | 0.827 | 0.632 |
| Random Forest | 0.804 | 0.805 | 0.799 | 0.807 | 0.805 | 0.583 |
| Decision Tree | 0.764 | 0.774 | 0.775 | 0.776 | 0.774 | 0.530 |

In this experiment, the SVM and the Logistic Regression classifiers performed similarly and outputted the same F1 score. Figure 6 presents this result, illustrating the identical confusion matrix generated by these two classifiers.

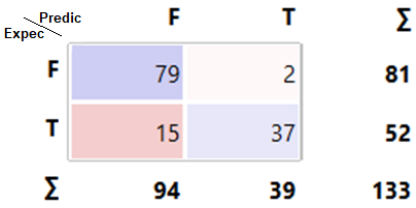


Fig. 6. The confusion matrix of the highest F1 score classifiers (SVM and Logistic Regression) with ReliefF extracted features.

## V. Conclusion

In this paper, we presented a Languaged-focused and Phoneme-based Deepfake Detection System (LPDDS) to address the growing concern regarding deepfake videos. And we detailed a methodology where we analyze specific phonemes, their corresponding mouth movements, and video-extracted features of the selected phonetic unit. We also presented a novel dataset comprising real and fake videos of three 2022 Brazilian presidential candidates for training and testing the classification algorithms. Furthermore, the LPDDS computational model demonstrated satisfactory results in authenticating or detecting fake videos containing the target phoneme.

In conclusion, we support the hypothesis that deepfake detection is feasible, particularly in non-English language deepfake videos where mouth expressions are often unique. And we believe that the findings of this paper contribute to the ongoing efforts to combat the increasing threat of deepfakes and safeguard the reliability of digital content. Regarding the conducted experiments, differences between classification ML algorithms and attribute selection methods are minor, with the best result obtained with Logistic Regression using all Kinetics features. For future experiments, we intend to utilize new DL-based approaches and the clustering analysis pre-implemented in the LPDDS computational model (Figure 1) to delimitate which extracted features cause the main classification errors and further populate the existing dataset with real and fake videos that could assist in this classification problem.

## Acknowledgment

We would like to thank the National Council for Scientific and Technological Development (CNPq) for the scholarship (164765/2020-4) and support of our research. A. S. Inacio thanks UNIEDU/FUMDES for the scholarship.

## References

[1] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2018.

[2] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," CoRR, vol. abs/1810.11215, 2018.

[3] T. Do Nhu, I. Na, and S. Kim, "Forensics face detection from gans using convolutional neural network," in International Symposium on Information Technology Convergence (ISITC), 2018.

[4] U. A. Ciftci and I. Demir, "Fakecatcher: Detection of synthetic portrait videos using biological signals," CoRR, vol. abs/1901.02212, 2019.

[5] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," CoRR, vol. abs/1811.00656, 2018.

[6] R. Xu, Z. Zhou, W. Zhang, and Y. Yu, "Face transfer with generative adversarial network," CoRR, vol. abs/1710.06090, 2017.

[7] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, "Recycle-gan: Unsupervised video retargeting," in Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[8] Z. Liu, H. Hu, Z. Wang, K. Wang, J. Bai, and S. Lian, "Video synthesis of human upper body with realistic face," CoRR, vol. abs/1908.06607, 2019.

[9] I. Perov, D. Gao, N. Chervoniy, K. Liu, S. Marangonda, C. Um'e, M. Dpfks, C. S. Facenheim, L. RP, J. Jiang, S. Zhang, P. Wu, B. Zhou, and W. Zhang, "Deepfacelab: A simple, flexible and extensible face swapping framework," CoRR, vol. abs/2005.05535, 2020.

[10] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama: Learning lip sync from audio," ACM Trans. Graph., vol. 36, no. 4, 2017.

[11] R. Kumar, J. Sotelo, K. Kumar, A. de Brebisson, and Y. Bengio, "Obamanet: Photo-realistic lip-sync from text," 2017.

[12] W. Wu, Y. Zhang, C. Li, C. Qian, and C. C. Loy, "Reenactgan: Learning to reenact faces via boundary transfer," CoRR, vol. abs/1807.11079, 2018.

[13] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[14] T. C. Wang, M. Y. Liu, J. Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro, "Video-to-video synthesis," in Advances in Neural Information Processing Systems, Eds., vol. 31. Curran Associates, Inc., 2018.

[15] S. A. Jalalifar, H. Hasani, and H. Aghajan, "Speechdriven facial reenactment using conditional generative adversarial networks," CoRR, vol. abs/1803.07461, 2018.

[16] O. Fried, A. Tewari, M. Zollhöfer, A. Finkelstein, E. Shechtman, D. B. Goldman, K. Genova, Z. Jin, C. Theobalt, and M. Agrawala, "Text-based editing of talking-head video," CoRR, vol. abs/1906.01524, 2019.

[17] E. Zakharov, A. Shysheya, E. Burkov, and V. S. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," CoRR, vol. abs/1905.08233, 2019.

[18] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, B. Catanzaro, and J. Kautz, "Few-shot video-to-video synthesis," in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[19] T. Shimba, R. Sakurai, H. Yamazoe, and J.-H. Lee, "Talking heads synthesis from audio with deep neural networks," in 2015 IEEE/SICE International Symposium on System Integration (SII), 2015.

[20] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang, "Talking face generation by adversarially disentangled audio-visual representation," CoRR, vol. abs/1807.07860, 2018.

[21] K. Vougioukas, S. Petridis, and M. Pantic, "End-to-end speech-driven facial animation with temporal gans," 2018.

[22] K. Vougioukas, S. Petridis, and M. Pantic, "Realistic speech-driven facial animation with gans," in Int. J. Comput. Vision, vol. 128, no. 5, p. 1398–1413, 2020.

[23] Y. Song, J. Zhu, D. Li, X. Wang, and H. Qi, "Talking face generation by conditional recurrent adversarial network," 2019.

[24] A. Pumarola, A. Agudo, A. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: One-shot anatomically consistent facial animation," 2019.

[25] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical cross-modal talking face generation with dynamic pixel-wise loss," CoRR, vol. abs/1905.03820, 2019.

[26] A. Lahiri, V. Kwatra, C. Fruh, J. Lewis, and C. Bregler, "Lipsync3d: Data-efficient learning of personalized 3d talking faces from video using pose and lighting normalization," CoRR, vol. abs/2106.04185, 2021.

[27] G. Mazaheri and A. K. Roy-Chowdhury, "Detection and Localization of Facial Expression Manipulations," CoRR, vol. abs/2103.08134, 2022.

[28] Md. S. Rana, B. Murali, A. Sung, "Deepfake Detection Using Machine Learning Algorithms," 10th International Congress on Advanced Applied Informatics (IIAI-AAI), 2021.

[29] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4724–4733, 2017.