Creating More Meaningful Explanations of Audio Deepfake Detection

Explaining Audio Deepfake Detection with Perceptible Features and LIME

Understandable Explanations of Synthetic Voice Detection with Perceptible Features and LIME

Bachelorarbeit

im Studiengang
Wirtschaftsinformatikim 6. Semester

vorgelegt von

Jaocb LaRock

Matr.-Nr.: 1667321

am 17.03.2025

Universität Siegen

Fakultät III: Wirtschaftswissenschaften, Wirtschaftsinformatik und Wirtschaftsrecht

Erstprüfer: Gunnar Stevens Gunnar.Stevens@uni-siegen.de

1 Abstract

With the ever-increasing threats posed by fake voice synthesis, reliable detection methods have become more important than ever. Previous research has tackled this problem using several methods, including various features and model architectures, but there remains a need in the area of fake voice detection for a method that offers a kind of explainability that can also be understood by the average person. We present a method of classification of audio samples as synthetic or bona fide using a combination of features with a focus on aspects of a voice sample that can be heard by the human ear, perceptible features, that stitches multiple subnetworks together in order to evaluate many differently shaped input features in a singular model. We then use part of this classifier as a surrogate model to produce explanations using the LIME method, with the hope of introducing a kind of explanation into this area that has the potential to be understood by a wider range of people and not only people who are already familiar with the field. In order to evaluate the contributions of the features to the average local explanation, we also introduce two metrics for aggregate evaluations of many local explanations, with the goal of making a robust and well justified conclusion about the potential usefulness of explanations generated using this method. Experiments were performed using the In-the-Wild dataset, a dataset created with a focus on testing generalization of previous methods of synthetic audio detection. Several of the surrogate models demonstrated strong performance on the dataset, even in comparison to previous methods. However, the explanations generated by this method do not seem to offer much potential for useful interpretation using the perceptible features, based on an evaluation of aggregated local explanations using the proposed metrics, as the imperceptible features have a much larger influence on the classification result and only some of the perceptible features had a positive influence on the classification correctness. We believe, however, that the proposed method offers a promising basis for future research on creating more useful explanations using such feature combinations.

2 Introduction

As the use of generative methods for the creation of synthetic voices becomes more widespread, so grows the need for reliable and usable detection methods for the protection of the security of people and businesses alike. In particular, the rise of deepfake technology has led to concerns about its potential misuse in areas such as politics, entertainment and national security by, for instance, potentially allowing malicious actors to create fake audio recordings that appear to be genuine statements made by public figures or to create genuine-seeming recordings of events that never happened, which could have significant consequences if used to spread misinformation, from defaming individuals to creating political tension [22, 1].

The field of Explainable AI (XAI) shows promise for producing useful and interpretable results from such models. Explainable AI refers to the ability of artificial intelligence systems, such as machine learning models and neural networks, to provide understandable and interpretable explanations for their decisions or predictions [11]. This means that XAI systems can articulate why they made a particular decision, what factors influenced it and how confident they are in their conclusion [11]. For the detection of audio deepfakes, this could mean that the model can justify its classification with feature-based evidence, allowing for verification of the result by the end user. Much of the existing research, further discussed in the next section, follows the path of producing a model with the best possible benchmarks against datasets of samples, without taking into account if the results can be usefully interpreted, while existing explorations into making audio deepfake detection explainable have produced results that often require a lot of background knowledge to meaningfully interpret. Aside from this, explainability in the area of audio deepfake detection remains an open challenge [6], one even promoted by regulatory bodies such as the European Union with the "right to explain" [9].

In this work, we will make a distinction of two main categories of input features

for a synthetic voice detection model: perceptible and imperceptible features. Perceptible features are features that can be perceived by the human ear, often vocal qualities such as jitter, shimmer or pitch fluctuation that have a wide range of uses even outside of audio classification such as diagnosis of disease [5], while imperceptible features are typically out of the range of human hearing, and may not be directly reflect a vocal quality, otherwise referred to as "speaker-independent" features [14]. We use a new combination of features for our method, with an emphasis being placed on perceptible features, as we believe these to be the most likely to be understood by the layperson.

We then combine these features together using a novel model architecture, with separate sub-models for each input feature that are combined and processed in a final model, referred to as the terminus model, allowing us to process different kinds of features separately and avoid unnecessary, costly preprocessing.

Explanations are generated using the LIME method [19], which allows us to, for an individual classification assess the impact of each input feature on the final result. In order to assess how useful the average local explanation is to a potential end user, and due to a lack of such assessments in previous research, we also introduce two metrics for aggregate evaluation of large number of generated LIME explanations, which we use to make a conclusion about the value of the selected feature set for the purpose of sensible explanations.

3 Related Work

Several researchers have already followed similar or relevant directions to that of this work. This section will summarize some efforts in previous research divided into categories of perceptible and imperceptible features, as well as previous efforts at explainability in this field.

3.1 Synthetic Voice Detection With Imperceptible Features

The most common approach in this field makes use of either learned or hand-crafted imperceptible features. Examples include spectrographic features such as the mel-spectrogram and its hand-crafted derivative the mel-frequency cepstral coefficients (MFCCs), both of which are present in our method due to their widespread successful use.

The work of Anagha et al. [2] is an example that makes use of mel-spectrograms in combination with convolutional neural networks, achieving well-performing results on the ASVSpoof2019 dataset [23].

A further work [25] makes use of MFCCs, in combination with other hand-crafted imperceptible features such as linear frequency cepstral coefficients (LFCCs) to not only classify audio samples as synthetic or authentic, but also to correctly classify the vocoder used to produce the sample with nearly perfect accuracy on a handcrafted dataset.

Other researchers, such as in the work of Qais et al. [17] make use of Fourier transforms, such as the short term Fourier transform (STFT) to achieve accurate reuslts on the ASVSpoof2017 dataset [7].

3.2 Synthetic Voice Detection With Perceptible Features

There is also a smaller but still notable body of work encompassing the use of perceptible features for the detection of synthetic voices, making use of a variety of features, some of which were selected for use in our experiments.

The work of Barrington et al. [3] explores the idea of classification using perceptible features, highlighting their potential for improving explainability in the space of deepfake audio detection. They do, however also demonstrate a drop in performance in comparison with imperceptible hand-crafted and deep-learning features in their experiments, which were performed on a combination of syn-

thetic and real audio datasets.

Chaiwongyen et al. [4, 5] also approach using perceptible features for classification in their works. They trained and tested using the dataset of the ADD2022 Challenge [27] and a simple model with a single hidden layer, resulting in lackluster performance in 2022, with better results using an expanded and improved feature set in 2023.

Li et al. [13] also approach perceptible features in their work, combining them with imperceptible (referred to as physical in the work) features and using various neural networks in their experiments. Overall, the combination of perceptible and imperceptible features was able to produce the best performance results of the experiments, which were performed as well on the dataset of the ASV2022 Challenge [27].

3.3 Explainable Models

One previous example of an implementation of an explainable model for use in audio deepfake detection came from Ge et al. [8], who explain feature influence on models using the SHAP (SHapley Additive exPlanations) method. They apply this method using log-scaled power spectrograms as a feature, training and testing using the datasets from the ASV2019 challenge [23], they are able to determine and graphically represent areas of importance on the spectrogram, as well as globally summarize the SHAP-values.

Another example is presented by Haq et al. [10], who use the changes in emotional state as an input feature and represent "unlikely" changes on a graph so that it can be understood by an eventual end user. In this case, they achieved their results by combining the output of fake video and fake audio classifiers in order to produce a final classification for a video sample with audio. Testing against the presidential deepfake dataset [20], they achieve impressive results in comparison to the standing benchmark on the dataset at the time.

4 Method

4.1 Black-Box Model

The model used for this experiment was made, trained and evaluated using the tools available in the TensorFlow and Keras libraries [15]. This section will go into further detail about the experimental setup for the black-box model as well as the way in which explanations were created and summarized.

4.1.1 Features

In order to increase the likelihood of the explanations being useful and understandable to the end user, a focus was placed on using multiple perceptible features as inputs for the classifier. Two imperceptible features were also added with the hypothesis, based on previous research, that they would positively of increasing the model performance.

Several features were used for the final version of the black-box model. Some are features that can be called generic, widely used in research involving the detection of synthetic while others are more specific and are cited accordingly. The features that are not summarized for a whole sample are extracted using a sliding window method, preventing a loss of fidelity that can be caused by compression of the feature to a standard size. A more detailed description of the individual features used follows:

- Harmonic-to-noise ratios: A perceptible feature inspired by previous research [4, 5, 13] but done in a sliding window fashion instead of on the whole file, the HNRs are the ratios of the strengths of harmonic frequencies to the strengths of "noise", the total strength outside the harmonic frequencies. Unlike some previously seen examples [4, 5], I calculate this ratio for each fundamental frequency length, instead of using one value for the whole sample. Given that γ_i is the harmonic energy in a given fundamental frequency cycle and ι_i is the residual energy in a given fundamental

frequency cycle, the HNR at that cycle is calculated as follows:

$$20log\frac{\gamma_i}{\iota_i}$$

- Mel-spectrograms: A widely-used imperceptible feature in synthetic audio detection, the mel-spectrogram is a spectrographic representation of an audio sample transformed in a manner to better represent human perception of frequencies [17].
- Mel-frequency cepstral coefficients: The derivative cepstral coefficients of the mel-spectrogram also considered imperceptible, also widely-seen in research on synthetic audio detection.
- Fundamental frequency lengths: Considered a perceptible feature, the fundamental frequency lengths (f0 lengths) are the lengths of every fundamental frequency cycle in the sample. This has also been previously used in certain research on synthetic audio detection [24]. The output is one-dimensional with time as the axis.
- Onset strength: As used in previous research [13], this perceptible feature represents the strengths of each onset in the audio sample, where an onset is point where there is a sudden rise in energy across the audio spectrum. This results in a one-dimensional output with time as the axis.
- Intensity: Also inspired by previoous research [13], intensity, also a perceptible feature, is the total power at each point in the audio sample, given in db. This is calculated by creating a fourrier transformation of the sample and then summing across the frequency-axis for every point on the time-axis. Resulting in a one-dimensional output.
- Pitch-fluctuations: Also classified as a perceptible feature, the pitch fluctuations are calculated for every sample in the audio as the difference between pitch at the given point and the pitch at the previous point. Pitch is estimated based on the maximum power harmonics. This feature is similar to the use of summarized pitch fluctuations in previous research

[12]. This feature is also one-dimensional with time as the axis. Given that H_i is the set of harmonic frequencies at fundamental frequency cycle i with $h_i \in H_i$ as a frequency of the set and $s(h_i)$ is the power of a given harmonic frequency at cycle i, the pitch can be estimated as follows:

$$p_i = s(max(H_i))$$

Then, given an offset x, the pitch fluctuations can be calculated as follows:

$$p_i - p_{i-x}$$

- Jitter features: As defined in previous attempts to identify synthetic audio [5] with perceptible features, jitter-based features measures the absolute variations in fundamental frequency cycles in comparison to the nearest x neighbors using the following method:

$$\frac{\frac{1}{N-1}\sum_{i=1}^{N-1}|T_i(\frac{1}{x}\sum_{n=i-m}^{i+m}T_n)|}{\frac{1}{N}\sum_{i=1}^{N}T_i}$$

  where T_i represents the extracted fundamental frequency lengths and N is the number of fundamental frequency periods.

- Shimmer features: Similarly defined to the jitter features, shimmer features, also considered perceptible and are used in the same research. They instead make use of the amplitudes at each fundamental frequency period. Their purpose is to capture irregular vocal fold vibrations which may be an indication but not a guarantee of a synthetic voice. [5].

$$\frac{\frac{1}{N-1}\sum_{i=1}^{N-1}|A_i(\frac{1}{x}\sum_{n=i-m}^{i+m}A_n)|}{\frac{1}{N}\sum_{i=1}^{N}A_i}$$

4.1.2 Model Architecture

The general architecture is designed to allow the combination of diverse features into one final prediction, regardless of whether the individual features have the same shape. This is achieved by using a separate model for each feature, with its own input and output layers (further referred to as sub-models), which are

then concatenated together, processed though a final model that pools into a final output value (which I will further refer to as the terminus model). The modular structure allows not only better performance and less memory use, due to the fact that fewer transformations are required in input preparation, but also allows for more flexibility in the construction of the model, making it easy to add and remove features using pre-defined functions depending on feature type. This kind of architecture also provides an advantage to the generation of explanations, as only the terminus part of the entire must be explained to interpret the importance of each input feature. Further explanation is in the following section. Figure 1 represents the layout of the model layers.

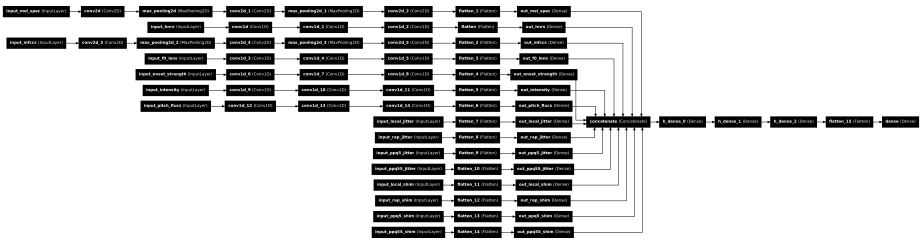

Figure 1: The architecture of the model used for the experiment.

4.1.3 Training and Evaluation

The black-box model described in the previous section was trained on the first portion of the dataset, differing depending on the exact experiment (more details are provided in the next section). Evaluation was then performed on the remaining samples in the dataset. Training batches consisted of 100 samples each with every window position for each sample and an upper limit of 1,000,000 lines per batch. For each batch, two epochs were performed. Because a single sample can have multiple input lines, evaluation cannot be performed on a per-line basis, but must instead be summarized. Two methods are possible: median and mean, with the difference in results being negligible in our experiments. The default threshold is 0.5, meaning that a result under 0.5 is a false result and over 0.5 is a true result. This threshold can however be adjusted, useful for

computing the EER.

4.2 Explanations

The explanations using the previously described surrogate model are generated with the Local Interpretable Model-agnostic Explanation (LIME) method [19]. This section will describe in detail how individual explanations are generated at a local level for individual samples, as well as how the usefulness of the explanations can be evaluated in general.

4.2.1 Generation of the Explanations

Because of the multi-shaped nature and independence of the input layers of the model, the explanations cannot be generated directly from the input itself. However, because the sub-models are separate from one another, having no influence on each other before being concatenated at the terminus, only the terminus part of the model is relevant for assessing the importance of the features in a single evaluation. This does, however, present two challenges:

- There are multiple input rows per sample.
- The features cannot be used directly for assessment at the terminus input layer.

The first problem can be solved by taking the mean of the explanation weights of each feature produced by each row of the sample features. This results in an average contribution of each feature to the classification of the model. Because the end-classification of the surrogate is also a mean, the means of the LIME-results are an accurate representation of the aggregate influence.

We approached the second problem by generating intermediate data for the LIME-evaluation. Intermediate data was generated by first decomposing the surrogate model into its component models, the sub-models and the terminus, and for each sub-model, running a prediction on the given input features. The results of the predictions of the sub-models were stored and used for further

analysis. This was done not only for the sample explanation data but also for the training data, to use as an input for the LIME explainer.

The results of the LIME explainer are then summarized together on a per-feature basis and normalized to produce a decimal number between -1 and 1 for every feature. In this case, a negative value implies that the given feature pushed the result of the model in the negative direction (i.e. not fake), while a positive value indicates the opposite.

4.2.2 Evaluation of the Explanations

In order to assess if the explanations have the potential to be useful to an end user, we pursued the goal of summarizing many explanations generated using the previously described method together in a meaningful way, in order to make a conclusion about the general usefulness of the individual features and the influence and potential use of the perceptible features. However, due to a lack of existing standardized metrics for the global evaluation of local metrics, we propose two metrics, with which we will make conclusions about the usefulness of this method.

In order to contextualize the metrics, we will make the following definitions.

- Let S be the set of samples with $s \in S$ as an element of the set.
- Let F be the set of features with $f \in F$ as an element of the set.
- Let w_{fs} be the weight value of feature f of sample s as produced by the explainer.
- Let l_s be the correct label of the sample s.

The first metric we will define is the mean of the absolute values of the weights of the explanations, summarized along the sample axis, delivering one value per feature. We will further refer to this metric as importance, and it can be mathematically defined as follows:

$$\frac{\sum_{s \in S} |w_{fs}|}{|S|}$$

The second metric we will define is the average aggregate correctness on a per-feature basis, which we will further refer to as trust. The trust per feature can be defined as follows:

$$\frac{\sum_{s \in S} w_{fs}(2l_s - 1)}{|S|}$$

5 Experimental Results

Using the methods described in the previous section, we performed experiments training and testing the surrogate model, as well as generating many explanations for the purpose of aggregate evaluation. This section will first discuss the results and performance of the surrogate model, followed by an evaluation of the aggregated explanations.

5.1 Dataset

The dataset used for these experiments is the "In-the-Wild" dataset, one which focuses on the generalization of audio deepfake detection models by collecting real-world data, in comparison to other examples that use more controlled laboratory conditions [16]. We have chosen it for these experiments because of the aforementioned focus on generalization, its relative recency, compared to some others and the fact that it has also been used previously by some other experiments, which provides a useful perspective against which we can compare our method.

5.2 Black-Box Model Results

Terminus	Features	Training	Accuracy	EER	AUC
Simple	standard	until 3474	95.02%	0.03702	0.90297
3 hidden layers	standard	until 10000	96.27%	0.03214	0.81763
Simple	standard	until 23833 2 epochs	90.07%	0.90069	0.78889
Simple	standard	until 10000 2 epochs	94.43%	0.04408	0.89445
Simple	standard	until 10000	94.28%	0.04840	0.90182
3 convolutional layers	standard	until 10000	62.80%	0.37198	0.50051
Simple	expand pitchflucs	until 10000	93.31%	0.05661	0.92727
3 hidden layers	expand pitchflucs	until 10000	93.87%	0.05317	0.83263

Table 1: Summary of the evaluation results of the experiments

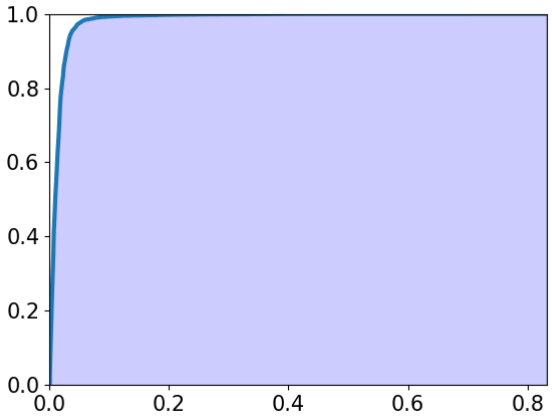

Figure 2: ROC curve of the best-performing model, with 3 hidden layers in the terminus and that standard feature set.

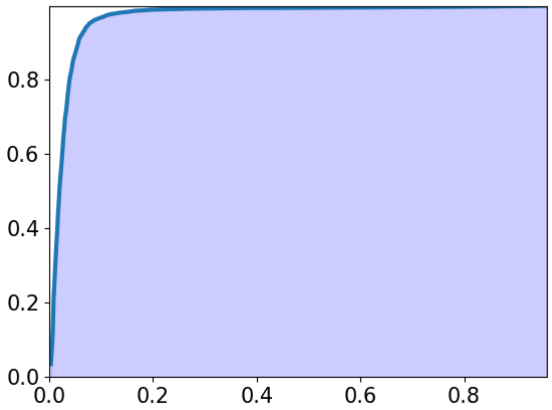

Figure 3: ROC curve of the model with expanded pitch fluctuation features.

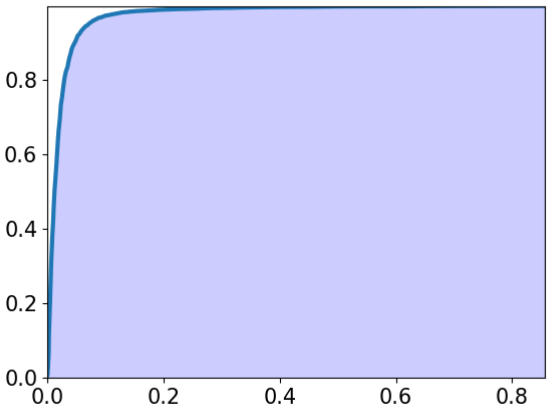

Figure 4: ROC curve of the model with expanded pitch fluctuation features and a terminus with three hidden layers.

A summary of some experiments based on the method described above is provided in table 1. The model that achieved the best results was trained on the first 10000 samples of the dataset and evaluated on the rest evaluated using the methods previously described. It has an evaluation accuracy of 96.3% with the default threshold of 0.5, with an EER of approximately 0.03214. This outperforms other results tested in the original paper on this dataset [16]. This result also exceeds some other results tested on this dataset, such as an example [26] produced using a variety of imperceptible features and trained on the ASV2019 dataset, while underperforming one other method [18] that made use of raw waveform-based features, training on 70 percent of the data in the dataset, validating on 10 percent and testing on the remaining 20 percent.

5.3 Explanation Results

In order to obtain a full picture of the quality of the explanations generated by this method, we have used selected two models from the last section and generated explanations on the first 500 samples of the testing data. We then

Feature	importance	trust
HNRs	0.1140	-0.0543
mel spectrogram	0.1084	-0.0378
MFCC	0.7029	0.3753
f0 lengths	0.0	0.0
onset strengths	0.0	0.0
intensities	0.0	0.0
pitch fluctuations	0.0	0.0
jitter features	0.1277	0.0250
shimmer features	0.1227	0.0017

Table 2: Summary of the aggregated explanation results of the best performing model

Feature	importance	trust
HNRs	0.1072	-0.0148
mel spectrogram	0.0428	-0.0187
MFCC	0.4383	0.2345
f0 lengths	0.0340	-0.0057
onset strengths	0.0101	-0.0027
intensities	0.0021	0.0021
pitch fluctuations	0.0059	0.0020
jitter features	0.2439	0.0019
shimmer features	0.1664	0.0070

Table 3: Summary of the aggregated explanation results of the model with expanded pitch fluctuation features

Feature	importance	trust
HNRs	0.2667	-0.0788
mel spectrogram	0.0982	-0.0434
MFCC	0.2879	0.2408
f0 lengths	0.1690	-0.0521
onset strengths	0.1384	-0.0613
intensities	0.1576	-0.0504
pitch fluctuations	0.0	0.0
jitter features	0.1444	0.0118
shimmer features	0.1747	0.0116

Table 4: Summary of the aggregated explanation results of the model with expanded pitch fluctuation features and a terminus with three hidden layers

summarized the explanations generated using both of the previously introduced metrics. A summary of the results of both metrics summarized by the feature is presented in tables 2, 3 and 4.

5.3.1 Best-Performing Model

Measuring by both metrics, the MFCCs had the most positive and the most correct influence on the classification results of the model, With the jitter features as a distant second for trust and importance. This indicates that the MFCCs had not only the most influence, but also the most correct influence on the result of the classification, while the other features had either little, no or negative influence on the classification.

5.3.2 Model with Expanded Pitch-Fluctuation Features

This model had similarly unpromising results, with the main difference being that all metrics have non-zero values, meaning that all the present features had an influence on the classifications which were used to generate the explanations. As with the previous model, HNRs and mel-spectrograms had negative overall

trust values, with the addition of the onset strengths also presenting a negative value, implying that these features have caused more harm than good in the tested classifications.

5.3.3 Model with Expanded Pitch-Fluctuation Features and Complex Terminus

The model with multiple pitch-fluctuation features in combination with a terminus containing multiple hidden layers fared similarly to the other two, this time with the pitch fluctuations having not influence on the outcome whatsoever. Unlike the previous examples, the trust value of the intensities feature was negative here.

6 Discussion

In this section, we will discuss the implications of the experimental results of the previous sections with a special focus being placed on the results of the explanation tests as it relates to the potential explainability for potential end users of such a system.

6.1 Viability and Potential of the Explanations

Because of these models' inclusion of several perceptible features, we made the hypothesis that this method has the potential to produce explanations that can be used as the basis for something more presentable to an end user. As can be seen in the previous section in the tables 2 3, the perceptible features did not contribute as much to the end result of the classification as their imperceptible counterparts, in spite of their increased presence in the overall set of input features. Two features, the HNRs and the mel-spectrograms even consistently had a net-negative impact on the correctness of the classifications in the experiments, leaving the question open, if future experiments may perform better without these features. In the case of the best performing model, four

perceptible features, f0 lengths, onset strengths, intensities and pitch fluctuations had an overall average influence on the classification of 0, measured on both metrics, indicating that the model did not learn a correlation between these features and the realness of audio samples in the dataset. These features did have an impact on the performence of the model with expanded pitch fluctuations, but the difference was still minimal. In spite of this, we believe that it is still worth investigating if other kinds of perceptible features will have more of an influence on the overall classification or if there are other imperceptible features that could be used in the place of MFCCs that may provide a better balance between classification correctness and a proportional influence on the end result.

6.2 Limitations

In spite of the potential that we see with this method, it does also present some limitations when it comes to its real-world use, aside from the shortfalls of the perceptible features discussed in the previous section. The first such limitation is in the computational performance of the model. Because of its relative size, the model requires relatively capable hardware in addition to several gigabytes of memory in order to classify samples. This means that the devices this or a similar model can run on are limited, and the possibility of running this or a similar model locally on a handheld device such as a smartphone is limited to none at the time of writing. Another limitation with this method is that, even though it is known that perceptible features such as the ones used in our method are able to be heard by the human ear and are even used in medicine, for example, for diagnoses [5], it is not yet clear to what extent exactly these features can be understood by the average person. Cursory research [21] is already present into what factors humans use to identify fake and real audio samples, as well as their performance, but the named aspects of the samples have not yet been mapped to such perceptible features as the ones used in this method, leaving the question of the usability of such explanations open. Finally,

we have not addressed in our experiments how the features can be presented to the end user. Certain implementations of LIME already have means to present their results graphically, but the effectiveness of such methods has yet to be studied.

7 Conclusion

To conclude, we have presented a model architecture using a combination of various features, perceptible and imperceptible, with the hypothesis that such a model could be used to generate explanations of its own results using the LIME method that are more useful to a potential end user who is not already familiar with the field. This model architecture resulted in a high level of performance in comparison to several other methods, indicating the potential for consistent and correct classifications. The explanations, however, were not able to produce the desired result, placing most of the weight on MFCCs, with most perceptible features either having little, no or negative impact on the end classification with the exception of jitter and shimmer.

We believe that this is a method that offers a large potential, but is not useful for the generation of understandable explanations in its current state. We suggest, however, that this method could be changed and improved in the future with other features or other changes to base parameters that could potentially produce more useful explanations while either holding the accuracy high like it is now or even improving it further.

References

[1] Marwan Albahar and Jameel Almalki. "DEEPFAKES: THREATS AND COUNTERMEASURES SYSTEMATIC REVIEW". In: . *Vol.* 22 (2005).

[2] R. Anagha et al. "Audio Deepfake Detection Using Deep Learning". In: *2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART)*. 2023 12th International Conference on System Modeling & Advancement in Research Trends (SMART). Moradabad, India: IEEE, Dec. 22, 2023, pp. 176–181. ISBN: 9798350369861 9798350369885. DOI: 10.1109/SMART59791.2023.10428163. URL: https://ieeexplore.ieee.org/document/10428163/ (visited on 05/28/2024).

[3] Sarah Barrington et al. *Single and Multi-Speaker Cloned Voice Detection: From Perceptual to Learned Features*. Sept. 27, 2023. DOI: 10.48550/arXiv.2307.07683. arXiv: 2307.07683[cs]. URL: http://arxiv.org/abs/2307.07683 (visited on 12/18/2024).

[4] Anuwat Chaiwongyen et al. "Contribution of Timbre and Shimmer Features to Deepfake Speech Detection". In: *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2022 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). Chiang Mai, Thailand: IEEE, Nov. 7, 2022, pp. 97–103. ISBN: 978-616-590-477-3. DOI: 10.23919/APSIPAASC55919.2022.9980281. URL: https://ieeexplore.ieee.org/document/9980281/ (visited on 05/28/2024).

[5] Anuwat Chaiwongyen et al. "Deepfake-speech Detection with Pathological Features and Multilayer Perceptron Neural Network". In: *2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). Taipei, Taiwan: IEEE, Oct. 31, 2023, pp. 2182–2188. ISBN: 9798350300673. DOI: 10.1109/APSIPAASC58517.2023.10317331. URL: https://ieeexplore.ieee.org/document/10317331/ (visited on 05/28/2024).

[6] Luca Cuccovillo et al. “Open Challenges in Synthetic Speech Detection”. In: *2022 IEEE International Workshop on Information Forensics and Security (WIFS)*. Dec. 12, 2022, pp. 1–6. DOI: 10.1109/WIFS55849.2022.9975433. arXiv: 2209.07180[eess]. URL: http://arxiv.org/abs/2209.07180 (visited on 12/18/2024).

[7] Héctor Delgado et al. “ASVspoof 2017 Version 2.0: meta-data analysis and baseline enhancements”. In: *The Speaker and Language Recognition Workshop (Odyssey 2018)*. The Speaker and Language Recognition Workshop (Odyssey 2018). ISCA, June 26, 2018, pp. 296–303. DOI: 10.21437/Odyssey.2018-42. URL: https://www.isca-archive.org/odyssey_2018/delgado18_odyssey.html (visited on 02/07/2025).

[8] Wanying Ge et al. *Explaining deep learning models for spoofing and deepfake detection with SHapley Additive exPlanations*. Apr. 26, 2024. DOI: 10.48550/arXiv.2110.03309. arXiv: 2110.03309[eess]. URL: http://arxiv.org/abs/2110.03309 (visited on 12/18/2024).

[9] Bryce Goodman and Seth Flaxman. “European Union regulations on algorithmic decision-making and a ”right to explanation””. In: *AI Magazine* 38.3 (Sept. 2017), pp. 50–57. ISSN: 0738-4602, 2371-9621. DOI: 10.1609/aimag.v38i3.2741. arXiv: 1606.08813[stat]. URL: http://arxiv.org/abs/1606.08813 (visited on 01/18/2025).

[10] Ijaz Ul Haq, Khalid Mahmood Malik, and Khan Muhammad. “Multimodal Neurosymbolic Approach for Explainable Deepfake Detection”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications* (Sept. 20, 2023), p. 3624748. ISSN: 1551-6857, 1551-6865. DOI: 10.1145/3624748. URL: https://dl.acm.org/doi/10.1145/3624748 (visited on 05/11/2024).

[11] Michael Hind. “Explaining explainable AI”. In: *XRDS: Crossroads, The ACM Magazine for Students* 25.3 (Apr. 10, 2019), pp. 16–19. ISSN: 1528-4972, 1528-4980. DOI: 10.1145/3313096. URL: https://dl.acm.org/doi/10.1145/3313096 (visited on 06/05/2024).

[12] Zahra Khanjani et al. "Learning to Listen and Listening to Learn: Spoofed Audio Detection Through Linguistic Data Augmentation". In: *2023 IEEE International Conference on Intelligence and Security Informatics (ISI)*. 2023 IEEE International Conference on Intelligence and Security Informatics (ISI). Charlotte, NC, USA: IEEE, Oct. 2, 2023, pp. 01–06. ISBN: 9798350337730. DOI: 10.1109/ISI58743.2023.10297267. URL: https://ieeexplore.ieee.org/document/10297267/ (visited on 05/28/2024).

[13] Menglu Li, Yasaman Ahmadiadli, and Xiao-Ping Zhang. "A Comparative Study on Physical and Perceptual Features for Deepfake Audio Detection". In: *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*. MM '22: The 30th ACM International Conference on Multimedia. Lisboa Portugal: ACM, Oct. 14, 2022, pp. 35–41. ISBN: 978-1-4503-9496-3. DOI: 10.1145/3552466.3556523. URL: https://dl.acm.org/doi/10.1145/3552466.3556523 (visited on 05/11/2024).

[14] Xin Liu et al. "Hidden-in-Wave: A Novel Idea to Camouflage AI-Synthesized Voices Based on Speaker-Irrelative Features". In: *2023 IEEE 34th International Symposium on Software Reliability Engineering (ISSRE)*. 2023 IEEE 34th International Symposium on Software Reliability Engineering (ISSRE). Florence, Italy: IEEE, Oct. 9, 2023, pp. 786–794. ISBN: 9798350315943. DOI: 10.1109/ISSRE59848.2023.00029. URL: https://ieeexplore.ieee.org/document/10301243/ (visited on 05/28/2024).

[15] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[16] Nicolas M. Müller et al. *Does Audio Deepfake Detection Generalize?* Apr. 21, 2022. arXiv: 2203.16263[cs,eess]. URL: http://arxiv.org/abs/2203.16263 (visited on 07/04/2024).

[17] Abu Qais et al. "Deepfake Audio Detection with Neural Networks Using Audio Features". In: *2022 International Conference on Intelligent Con-*

troller and Computing for Smart Power (ICICCSP). 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP). Hyderabad, India: IEEE, July 21, 2022, pp. 1–6. ISBN: 978-1-66547-258-6. DOI: 10.1109/ICICCSP53532.2022.9862519. URL: https://ieeexplore.ieee.org/document/9862519/ (visited on 05/28/2024).

[18] Rishabh Ranjan, Mayank Vatsa, and Richa Singh. "STATNet: Spectral and Temporal features based Multi-Task Network for Audio Spoofing Detection". In: *2022 IEEE International Joint Conference on Biometrics (IJCB)*. 2022 IEEE International Joint Conference on Biometrics (IJCB). Abu Dhabi, United Arab Emirates: IEEE, Oct. 10, 2022, pp. 1–9. ISBN: 978-1-6654-6394-2. DOI: 10.1109/IJCB54206.2022.10007949. URL: https://ieeexplore.ieee.org/document/10007949/ (visited on 05/28/2024).

[19] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. Aug. 9, 2016. DOI: 10.48550/arXiv.1602.04938. arXiv: 1602.04938[cs]. URL: http://arxiv.org/abs/1602.04938 (visited on 01/18/2025).

[20] Aruna Sankaranarayanan et al. "The Presidential Deepfakes Dataset". In: ().

[21] Filipo Sharevski et al. "Blind and Low-Vision Individuals' Detection of Audio Deepfakes". In: *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*. CCS '24: ACM SIGSAC Conference on Computer and Communications Security. Salt Lake City UT USA: ACM, Dec. 2, 2024, pp. 4867–4881. ISBN: 979-8-4007-0636-3. DOI: 10.1145/3658644.3690305. URL: https://dl.acm.org/doi/10.1145/3658644.3690305 (visited on 12/17/2024).

[22] Namosha Veerasamy and Heloise Pieterse. "Rising Above Misinformation and Deepfakes". In: *International Conference on Cyber Warfare and Security* 17.1 (Mar. 2, 2022), pp. 340–348. ISSN: 2048-9889, 2048-9870. DOI: 10.34190/iccws.17.1.25. URL: https://papers.academic-

conferences.org/index.php/iccws/article/view/25 (visited on 09/23/2024).

[23] Xin Wang et al. *ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech*. July 14, 2020. arXiv: 1911.01601[cs,eess]. URL: http://arxiv.org/abs/1911.01601 (visited on 07/03/2024).

[24] Jun Xue et al. "Audio Deepfake Detection Based on a Combination of F0 Information and Real Plus Imaginary Spectrogram Features". In: *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*. MM '22: The 30th ACM International Conference on Multimedia. Lisboa Portugal: ACM, Oct. 14, 2022, pp. 19–26. ISBN: 978-1-4503-9496-3. DOI: 10.1145/3552466.3556526. URL: https://dl.acm.org/doi/10.1145/3552466.3556526 (visited on 05/11/2024).

[25] Xinrui Yan et al. "An Initial Investigation for Detecting Vocoder Fingerprints of Fake Audio". In: *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*. MM '22: The 30th ACM International Conference on Multimedia. Lisboa Portugal: ACM, Oct. 14, 2022, pp. 61–68. ISBN: 978-1-4503-9496-3. DOI: 10.1145/3552466.3556525. URL: https://dl.acm.org/doi/10.1145/3552466.3556525 (visited on 05/11/2024).

[26] Yujie Yang et al. "A Robust Audio Deepfake Detection System via Multi-View Feature". In: *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Seoul, Korea, Republic of: IEEE, Apr. 14, 2024, pp. 13131–13135. ISBN: 979-8-3503-4485-1. DOI: 10.1109/ICASSP48485.2024.10446560. URL: https://ieeexplore.ieee.org/document/10446560/ (visited on 05/28/2024).

[27] Jiangyan Yi et al. *ADD 2022: the First Audio Deep Synthesis Detection Challenge*. July 2, 2024. DOI: 10.48550/arXiv.2202.08433. arXiv:

2202.08433[cs]. URL: http://arxiv.org/abs/2202.08433 (visited on 01/22/2025).