Every Breath You Don't Take: Deepfake Speech Detection Using Breath

SETH LAYTON, THIAGO DE ANDRADE, DANIEL OLSZEWSKI, and KEVIN WARREN,
University of Florida, Gainesville, Florida, USA
CARRIE GATES, Dalhousie University, Halifax, Nova Scotia, Canada
KEVIN BUTLER and PATRICK TRAYNOR, University of Florida, Gainesville, Florida, USA

Deepfake speech represents a real and growing threat to systems and society. Many detectors have been created to aid in defense against speech deepfakes. While these detectors implement myriad methodologies, many rely on low-level fragments of the speech generation process. We hypothesize that breath, a higher-level part of speech, is a key component of natural speech and thus improper generation in deepfake speech is a performant discriminator. To evaluate this, we create a breath detector and leverage this against a custom dataset of online news article audio to discriminate between real/deepfake speech. Additionally, we make this custom dataset publicly available to facilitate comparison for future work. Applying our simple breath detector as a deepfake speech discriminator on in-the-wild samples allows for accurate classification (perfect 1.0 AUPRC and 0.0 EER on test data) across 33.6 hours of audio. We compare our model with the state-of-the-art SSL-wav2vec and Codecfake models and show that these complex deep learning model completely either fail to classify the same in-the-wild samples (0.72 AUPRC and 0.89 EER), or substantially lack in the computational and temporal performance compared to our methodology (37 seconds to predict a 1 minute sample with Codecfake vs. 0.3 seconds with our model).

CCS Concepts: • **Security and privacy** → **Intrusion/anomaly detection and malware mitigation**; **Domain-specific security and privacy architectures**;

Additional Key Words and Phrases: breath, synthetic, speech, audio, detection, deepfake

ACM Reference format:
Seth Layton, Thiago De Andrade, Daniel Olszewski, Kevin Warren, Carrie Gates, Kevin Butler, and Patrick Traynor. 2025. Every Breath You Don't Take: Deepfake Speech Detection Using Breath. *Digit. Threat. Res. Pract.* 6, 3, Article 13 (September 2025), 18 pages.
https://doi.org/10.1145/3754456

This work was supported in part by the National Science Foundation grants CNS-1933208 and CNS-2446321. Any findings and opinions expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

Authors' Contact Information: Seth Layton (corresponding author), University of Florida, Gainesville, Florida, USA; e-mail: sethlayton@ufl.edu; Thiago De Andrade, University of Florida, Gainesville, Florida, USA; e-mail: tdeandradebezerr@ufl.edu; Daniel Olszewski, University of Florida, Gainesville, Florida, USA; e-mail: dolszewski@ufl.edu; Kevin Warren, University of Florida, Gainesville, Florida, USA; e-mail: kwarren9413@ufl.edu; Carrie Gates, Dalhousie University, Halifax, Nova Scotia, Canada; e-mail: carrie.gates@gmail.com; Kevin Butler, University of Florida, Gainesville, Florida, USA; e-mail: butler@ufl.edu; Patrick Traynor, University of Florida, Gainesville, Florida, USA; e-mail: traynor@ufl.edu.

ACM 2576-5337/2025/9-ART13
https://doi.org/10.1145/3754456

1 Introduction

Deepfake speech (e.g., **text-to-speech (TTS)**, deepfakes, voice assistants) aims to make the differentiation between synthetic and organic speech difficult [25, 34]. While such audio has many benign uses, the potential for dangerous applications has created the need for accurate and automated demarcation of human-spoken from synthetically generated audio.

The research community has responded with competitions such as ASVspoof [21, 43, 45, 47], ADD [49], and SASV [20]. These competitions curate datasets of deepfake and real speech and invite participants to create detection algorithms to test on these datasets. Subsequently, these datasets are the *de facto* standard for deepfake speech and give a baseline of comparison for all current and future deepfake speech detectors. Most of the currently existing speech deepfake detectors focus on low-level spectral (e.g., spectrogram, MFCC, LFCC, and **constant-Q cepstral coefficient (CQCC)**) imperfections created during the audio generation pipeline and show starkly different classification results vs. human interpreters [44]. This technique of low-level spectral detection will be rendered obsolete due to the rapid advancement of the speech generation field. Thus, a paradigm shift toward high-level speech features such as prosody detection [5], emotion detection [19], and anatomical shape detection [8] is underway.

One promising avenue for high-level speech feature exploration is breath, as breathing is one of the subtle ways that humans subconsciously perceive naturalness in speech [1, 9, 40]. Additionally, the demand for automatic breath detection methods for medical research purposes is elevated due to COVID-19 [14]. This demand influenced the INTERSPEECH 2020 Computational Paralinguistics Challenge to create the Breathing Sub-Challenge, a track dedicated to breath detection [36]. Current state-of-the-art breath detection methods include spectral-based models [17, 30], acoustic models [9], and raw speech waveform deep learning [30]. While these methods automatically detect breath in audio, applying these techniques to the realm of deepfake speech to demarcate real from speech deepfakes is an open challenge. Toward this, we explore the viability of breaths as a detection mechanism for deepfake speech.

Our work emphasizes the practical significance of addressing in-the-wild deepfake speech, particularly in real-world applications such as news outlets that use high-quality synthetic speech to maintain listener engagement. Although academia often focuses on the development of cutting-edge deepfakes, these are seldom encountered extensively online. Thus, understanding the industry standard, which impacts the largest audience, forms the core of our work. Toward this, we collect speech samples from online news vendors that provide both TTS and human-read audio options, allowing us to assess the efficacy of breath detection as a novel discriminator for deepfake speech.

This work is essential because it leverages simple yet effective models that focus on breath patterns, a feature largely overlooked in distinguishing between real and synthetic speech. This focus is crucial considering the growing prevalence of deepfake technology and its potential misuse. Our approach addresses the challenge of detecting deepfake speech with reliability and efficiency, without succumbing to the common pitfalls associated with complex machine learning techniques that often struggle with generalization.

By conducting a comparative analysis between our models and state-of-the-art systems, we underscore the practical utility and relevance of using breath events. This highlights the necessity for adaptable and efficient detection strategies suited to real-world deepfake scenarios. Moreover, it reinforces the argument against the exclusive use of academic datasets like ASVspoof for a comprehensive evaluation, as they may not fully capture the diverse and evolving nature of deepfake speech used in everyday contexts.

Our main contributions are:

- We perform a study of currently deployed in-the-wild synthetic speech.
- We create a generation-agnostic deepfake speech detector, based solely on breaths.
- We publish a dataset of in-the-wild TTS and human-read speech.
- We highlight shortcomings and over-reliance on complex deep learning detection models.

The remainder of this article is organized as follows: Section 2 states our hypothesis; Section 3 details our methodology; Section 4 presents our experimental results; Section 5 offers discussion and insights based on our findings; Section 6 discusses related work; and Section 7 presents our concluding remarks.

2 Hypothesis

We hypothesize that current speech deepfakes generation techniques do *not* sufficiently incorporate breaths.

To investigate this claim, we must first determine if breath is a generalizable speech feature. Thus, we define our first **research question (RQ)**:

RQ1 · Are breaths automatically detectable, intra and inter-speaker?

As breath is one of the ways that humans determine naturalness in speech; we define our second RQ:

RQ2 · Do current deepfake voices generate breaths?

Combining the previous RQs, we define our final RQ:

RQ3 · Are breaths able to accurately discriminate between in-the-wild deepfake and real samples?

3 Methodology

To determine if breaths are a generalizable feature between speakers and subsequently useful as a discriminator against real and fake speech, we define deepfake speech, gather real and fake data, implement and test detection models, and evaluate the performance of breathing. This section describes our methodologies for collecting breathing samples, deepfake samples, and our algorithms for detecting these samples along with our evaluation metrics.

3.1 Cheapfake vs. Deepfake

Synthetic media is a spectrum that spans from generative (e.g., machine learning and artificial intelligence) to manually altered (e.g., Photoshop and speech waveform manipulation) samples. The use of **deep neural networks (DNNs)** creates a deepfake and using cheap manual software manipulation creates a cheapfake [33]. Specifically, a cheapfake requires a pre-existing sample of a specific individual for manual modification whereas a deepfake may be fully generative and not require a source sample. While both forms of synthetic media influence society, we focus on the dominant form of synthetic media known as deepfakes for the remainder of this article. More critically, we only consider media samples that are entirely real or entirely fake (i.e., no partially fake samples with segments altered). Additionally, this extends to manipulation techniques such as altering the rate of speech or changing the pitch.

3.2 Dataset

We employ a multi-tiered model pipeline that requires independent datasets during the training phase. We gather single-speaker podcast audio for training the breath detector and online news articles read by humans and TTS algorithms for the final deepfake speech detection training and testing.

3.2.1 Why Not ASVspoof. ASVspoof is the *de facto* standard dataset when creating and testing a deepfake speech detector and it gives a community baseline for comparison. However, for our task, this dataset is not representative, sufficient, or realistic. First, our model relies on features of breathing, and thus, samples must be sufficiently long to contain a breath (*93%* of ASVspoof 2021 samples are shorter than 5 seconds). As breaths in read and spontaneous speech are expected at a rate of 8–14 per minute [31], each ASVspoof sample is not expected to contain any of these important high-level features. Additionally, many of the samples longer than 5 seconds are filled with incoherent speech due to generation issues. Figure 1 shows the CDF for the length of

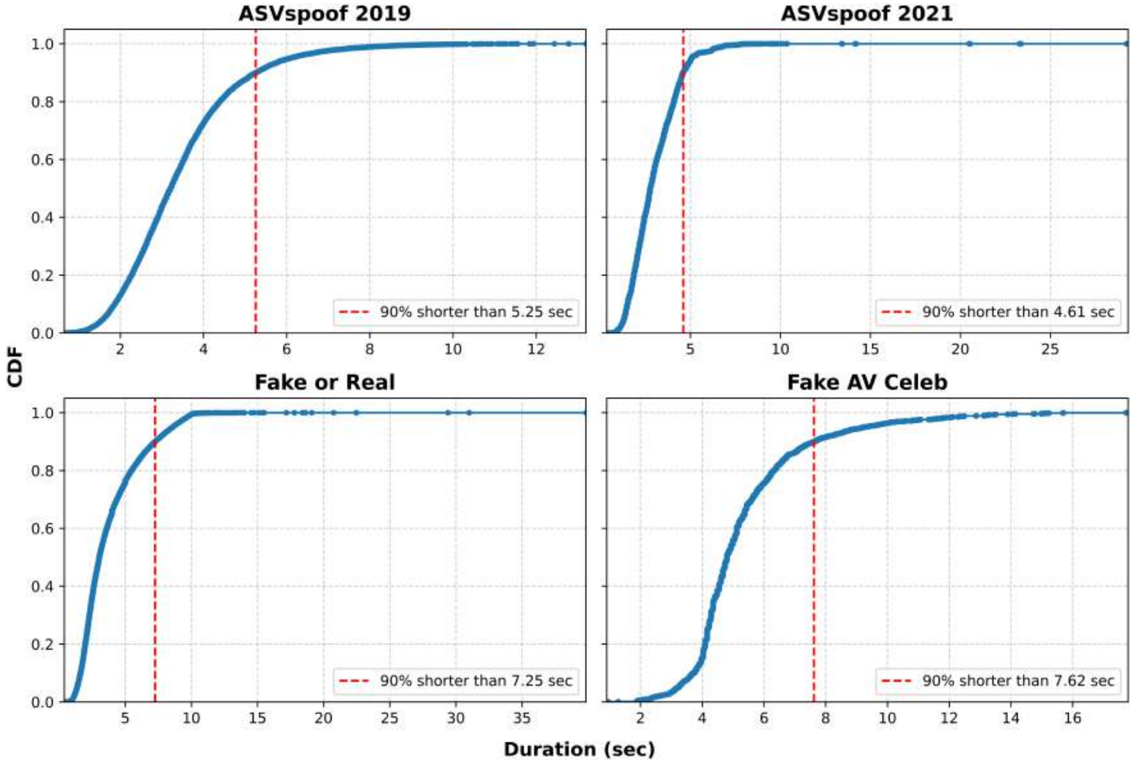

Fig. 1. CDFs of the length of samples for four popular audio deepfake datasets. We show that the vast majority of samples in these datasets are shorter than 5 seconds for ASVspoof and 7.5 seconds for Fake or Real and Fake AV Celeb, thus showing that these datasets are unlikely to contain breathing.

samples from some popular deepfake speech datasets. This shows that ASVspoof is not alone in containing a majority of short samples and as such none of these datasets make sense for our application.

Moreover, the class distribution of the ASVspoof 2021 samples is unrealistic (*97%* deepfake and 3% real in the evaluation set). This class distribution is vastly different from the expected in-the-wild distribution and would require multiple resampling techniques to correct, which could bias the results due to the vast imbalance of the dataset [23].

To handle these issues, we opt to gather samples that are *currently* in the wild today as a better representation of deepfake speech and contain a balanced real/fake class distribution.

3.2.2 Deepfakes in the Wild. Wild deepfakes differ from traditional (i.e., research/academically created) deepfakes in several key aspects. While normal deepfakes are often created in controlled environments with high-quality audio and complex methodologies, deepfakes found in the wild are generated using more accessible, less sophisticated tools, resulting in variable quality. Furthermore, their rapid distribution across social media platforms introduces challenges in detection owing to diverse audio sources and environments. This nature of deepfakes in the wild creates difficulties in maintaining consistent audio artifacts for detection, necessitating adaptive detection technologies. As such, developing technologies must focus on robust, real-time analysis to handle the unpredictability and scale of wild deepfake proliferation.

3.2.3 Podcasts. We curate a training dataset of podcasts that meet specific criteria and manually annotate these podcasts for breath locations. Each podcast is single-speaker, contains no background music, is free from obvious background noise, and breathing is noticeable. In total, we selected 10 podcasts from 4 speakers totaling just over 5 hours. Each podcast is manually annotated for precise breath locations by two independent annotators, and a third annotator verifies and reconciles any discrepancies in the annotations from each initial annotator to ensure all breaths are captured.

3.2.4 News Articles. We gather news article audio from online news vendors for training and testing our deepfake speech detector. We search for news articles with strings like "Listen to this article" and "Hear the full article" keywords on the Web page to indicate that there is an audio version of the transcript available. Manual checking of the news vendor is required to determine if the news articles provided are TTS generated or spoken by a human. In total, we collected 282 TTS (29.49 hours) and 51 human-read (22.99 hours) news articles from four different news outlets (two TTS sets and two human-read sets).[1]

3.3 Breath Detection

First, we create a breath location detector to highlight breaths in a given audio sample. Toward this, we take inspiration from rare-event detection [3, 39] to create our pipeline for breath detection. Unlike these works, however, we do not perform image analysis on spectrograms, but instead use the raw values computed from the spectrogram. Thus, all the layers of our model lose a dimension, reducing model complexity and providing speed increases to training and inference which helps real-time predictions.

3.3.1 Feature Selection. We first calculate raw values of the mel-spectrogram (dB converted), **zero crossing rate (ZCR)**, and **root mean squared energy (RMSE)** (dB converted). Each of these features slides a window over a waveform to calculate an overlapping temporal-based value. As the size of the window and the duration between windows may be optimized, we test a variety of values ranging from 5 ms to 200 ms *window_length* and 2.5 ms to 25 ms *hop_length*. The result of this feature extraction is an array of frames that contain spectrogram, ZCR, and RMSE values for an entire audio sample. The number of frames in an array is calculated as $num_{frames} = \frac{sample\_duration(ms)}{hop\_length}$, and the value of each frame is the aggregated mel-spectrogram, ZCR, and RMSE for *window_length* duration. For example, a 5-second excerpt of audio with a *window_length* of 50 ms and a *hop_length* of 5 ms creates a 1000-frame array. We test a spectrum of sizes, shapes, and durations for these features and select a *window_length* of 20 ms, a *hop_length* of 2.5 ms, and 128 mel-spectrogram buckets for our final model as these produce the best results.

Next, using the manually annotated location of breath we denote each frame in the array as either a breath (i.e., positive class) or not a breath (i.e., negative class). A frame is considered to contain breath if more than half the *window_length* of that frame is annotated as a breath. Figure 2 shows our selected features and how they change during a breath and speaking before/after breathing. For breaths, ZCR and RMSE tend toward medium values between silence and spoken segments and the mel-spectrogram shows only energy at lower frequencies.

3.3.2 Model Architecture. Based on Székely et al. [39], we build a multi-tiered convolutional and recurrent neural net detection model. The model architecture starts with two 1D convolutional layers. The first Conv1D layer has 16 filters with a kernel size of 3, strides of 1, same padding, and ReLU activation. This is followed by batch normalization, max pooling with a pool size of 3, and a 0.2 dropout. The second Conv1D layer has eight filters with a kernel size of 1, similarly accompanied by batch normalization, max pooling (pool size of 3), and dropout.

These initial convolutional layers are used as input to a bidirectional LSTM layer, which processes the sequences learned from the convolutional layers. This layer is crucial for modeling temporal dependencies within the audio data. The LSTM layer's output is connected to a dense layer with a sigmoid activation for the final prediction. We use binary cross-entropy as the loss function and Adam as the optimizer for efficient training.

[1] Each sample is entirely deepfake or entirely real. There is no scenario for deepfake speech injected into real audio. We do not redistribute the audio; however, links to news article Web sites are freely available at: https://sites.google.com/view/ebydt/.

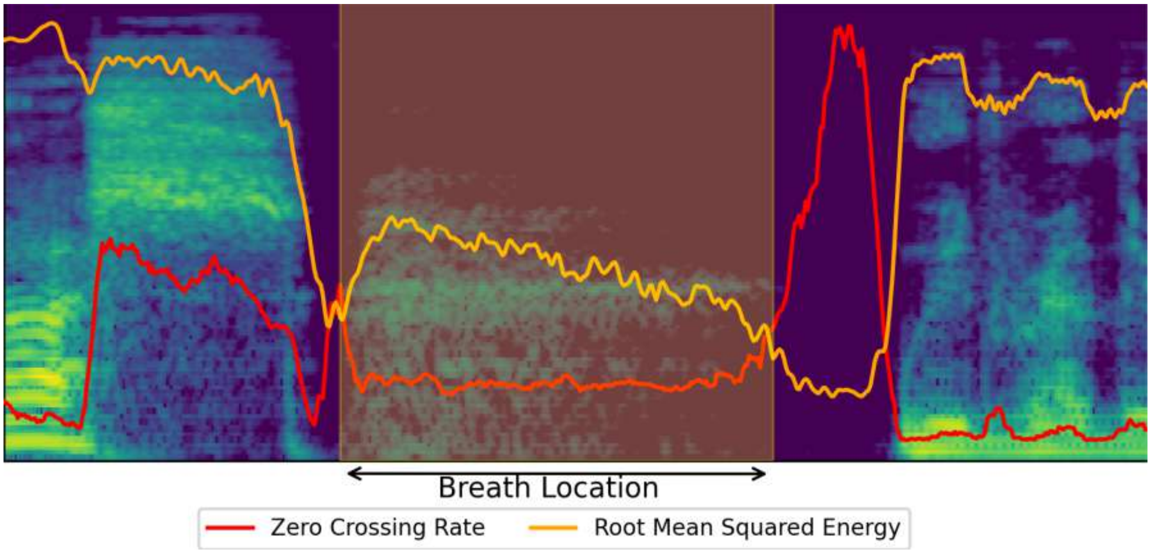

Fig. 2. A visual representation for a segment of speech containing a breath using a *window_length* of 20 ms and a *hop_length* of 2.5 ms. During the spoken segments before and after the breath RMSE is at peak values while the ZCR is at minimum values. Immediately surrounding a breath is a non-voiced segment where the RMSE values drop and ZCR values rise, but then both move to a medium value during the breath. Additionally, the background mel-spectrogram shows higher energy across all frequencies during spoken segments, medium energy at lower frequencies during breaths, and relatively little energy at all frequencies for silence. RMSE, root mean squared energy; ZCR, zero crossing rate.

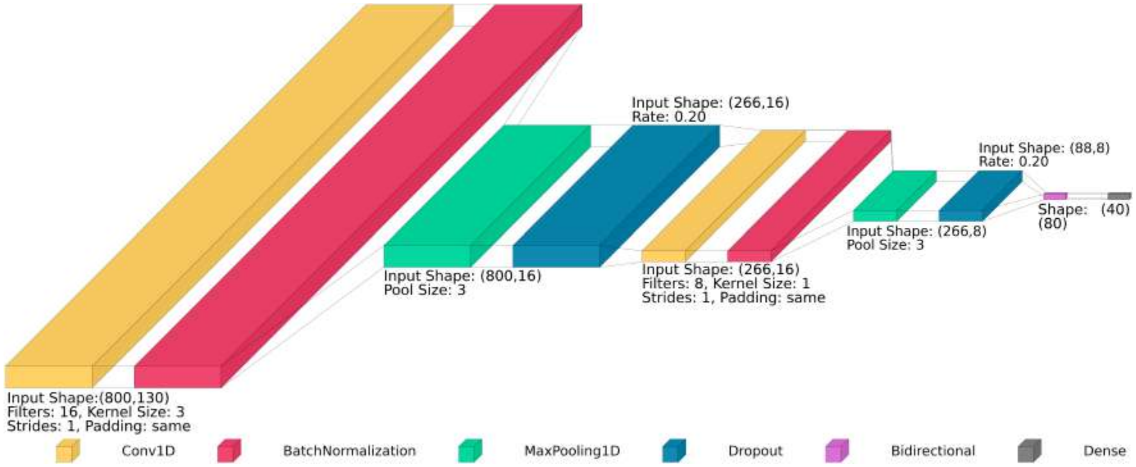

Fig. 3. A visual representation of the breath detection model architecture.

The pipeline for this model is shown in Figure 3. We employ a *window_length* of 20 ms and a *hop_length* of 2.5 ms to generate 128 mel-spectrogram buckets, leveraging a batch size of 32. We experimented with many different shapes and sizes for our breath detection model architecture such as changing the total number of convolutional layers, adjusting the number of filters and kernel sizes of each convolutional layer, altering the default max pool size, and changing the size of the dropout layer. Ultimately, we select the sizes and parameters defined in Figure 3.

Input to our breath detection model is crafted by sectioning an entire audio sample into 2-second segments, which are fed sequentially to the model, yielding predictions every 50 ms (40 predictions per segment). Choosing 2-second slices optimizes training and inference times without sacrificing performance. Each segment consists of 800 slices of 2.5 ms features, with 128 mel-spectrogram values, alongside the ZCR and RMSE features, totaling 130 features. With a batch size of 32, the input shape is (32, 800, 130).

Finally, the output for each 2-second audio chunk is a series of 40 binary classifications, determining the presence of a breath within each 50 ms slice. This detailed architecture ensures efficient and accurate breath detection across varied audio inputs.

3.3.3 Model Result Post-Processing. Our pipeline implements a simple mechanism on the resultant predictions. We remove any predicted breaths that are shorter than 150 ms as we measured no breaths shorter than 150 ms in any of our podcasts. This post-processing affects few samples, yet helps smooth out the resultant breath locations. We then use these breath locations as input features for the deepfake speech detection algorithm.

3.3.4 Metrics. We use a multitude of metrics for our full pipeline and evaluation of intermediary and final results. For breath detection, we use the binary cross-entropy loss function for hyperparameter tuning and use **area under the precision-recall curve (AUPRC)** for evaluating model performance. We use AUPRC as our main metric as it is robust to imbalanced data [4, 16] and effectively highlights performance on the important class. As breaths are an imbalanced class distribution problem (i.e., non-breathing heavily outweighs breathing in normal speech) and are the important class, AUPRC is a suitable metric for honestly evaluating performance.

3.3.5 Dependency on Breath Extraction. In the context of our hypothesis, a hyper-accurate breath detector is not required. The primary objective is not the exhaustive identification of every breath in an audio file, but rather the sufficient detection of breath events to serve as reliable discriminators for building an accurate deepfake speech detector. If our hypothesis holds (explored in Section 4.1), even a moderate level of precision in detecting breath events will effectively identify deepfake speech. This approach leverages the presence of natural, physiological cues as a robust differentiator between genuine and synthetically generated speech, thereby providing effectiveness without requiring comprehensive breath detection.

3.4 Deepfake Speech Detection

The final component of our pipeline uses the predicted breath locations as input. For an audio sample, we use the predicted breath locations to calculate three features: average breaths per minute, average breath duration, and average spacing between breaths. With these features, we create multiple simple deepfake speech detectors to test the viability of computationally inexpensive methods and compare models to a complex deep learning model (Section 4.2.3). The first is a thresholding classifier that uses previously measured breath statistics. The second is a **C-support vector classification (SVC)** algorithm with a poly kernel, a regularization parameter of 1, and a degree of 2. Finally, we implement a three-tiered decision tree with default parameters. Each of these mechanisms produces a single binary prediction for the entire audio sample. The pipeline for this deepfake detection model is shown in Figure 4. Furthermore, to ensure that we are not overfitting when training, we perform leave-one-out k-fold cross-validation for the SVC and decision tree models [18].

Following our hypothesis, we expect that the thresholding method should be sufficient as a final discriminator if deepfake speech does not produce any breaths. However, if breaths are produced, then a more robust mechanism would be required to accurately discriminate between real and fake samples. Thus, we apply both techniques and compare the results to determine viability and then compare them to complex and computationally expensive state-of-the-art.

3.4.1 Metrics. For the final deepfake speech detector, we implement a range of metrics to help contextualize model performance, as single metrics may fail to honestly describe model performance. These metrics are accuracy,

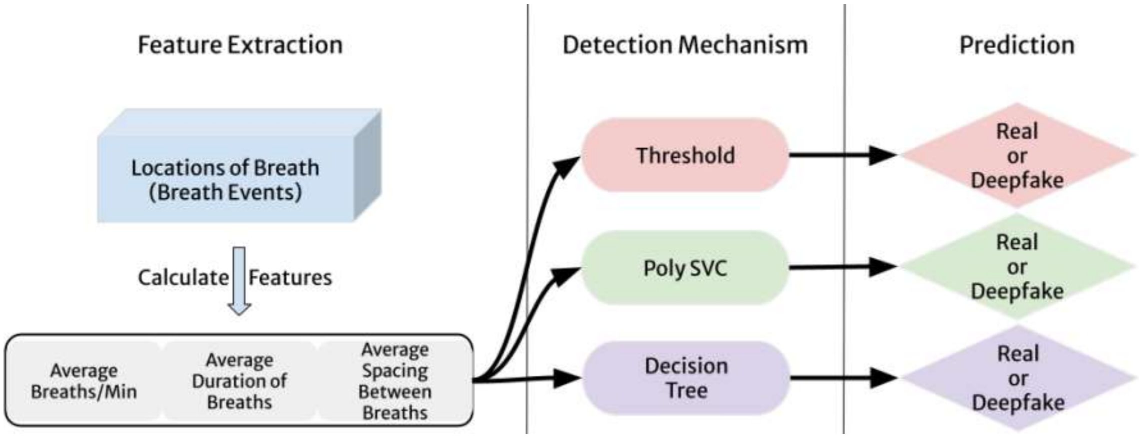

Fig. 4. A visual representation of the final stage of the detection pipeline. We use/compare three different simple classifiers in the last stage to showcase the relative interchangeability of models for final prediction.

F1 score, precision, recall, AUPRC, **equal error rate (EER)**, true positives, true negatives, false positives, and false negatives. For the thresholding method, we are unable to use AUPRC and EER as these metrics require probabilities, and only a prediction is generated. Finally, we note that the use of EER as a performance metric is deprecated by ISO/IEC standards [13] and has been shown to obfuscate results [10, 38]; however, EER remains the current community standard for deepfake speech detection.

4 Experiments

To sufficiently answer our RQs defined in Section 2, we perform two experiments starting with breath generalizability and finishing with deepfake speech (deepfake) detection. This section outlines our results and answers each RQ proposed in Section 2; additionally we compare to current state-of-the-art.

4.1 Breath Generalizability (RQ1)

To understand if breath sounds are generalizable between different individuals (thereby justifying them as a deepfake speech discriminator), we create three tests. RQ1 tests the generalizability of breathing with Test 1 obtaining a comparative baseline for successive tests, Test 2 examining the performance of each podcast, and Test 3 examining the performance of each speaker. Each test uses our breath detection model defined in Section 3.3 without any changes.

4.1.1 Test 1. Test 1 creates a best-case baseline for comparison with Tests 2 and 3 employing k-fold cross-validation. Test 1 takes $(\frac{1}{x} * 100)\%$ consecutive frames, where x is the total number of podcasts, randomly from each podcast to use as the validation set and uses the remaining $100 - (\frac{1}{x} * 100)\%$ from each podcast as the training set. We do this process 100 times and calculate the validation AUPRC to obtain a baseline for comparison. This test gives the best-case scenario due to having a subsample of every podcast/speaker in each training and validation batch. This is in contrast to the following two tests which avoid overlapping training and validation speakers/podcasts.

4.1.2 Test 2. Test 2 employs a leave-one-out strategy to determine the impact on breath detection for each podcast. A podcast is set aside individually as the validation set, and the remaining $x - 1$ podcasts become the training set. We retrain our breath detection model using the training set and calculate the AUPRC for the validation set. We do this for all x podcasts and compare the results to Test 1 to highlight any podcasts that may be problematic, or not generalizable.

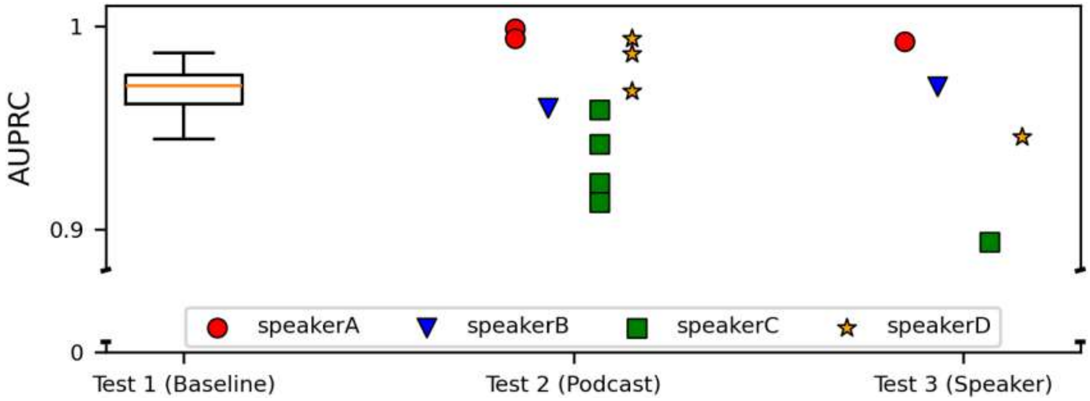

Fig. 5. The baseline validation testing on all podcasts vs. the leave-one-out testing for each podcast and each speaker. Each point is a specific speaker/podcast as the validation set. We show that breath measured in this capacity is generalizable and thus useful as a deepfake speech discriminator.

4.1.3 Test 3. While similar to Test 2, Test 3 evaluates the impact on breath detection for a single speaker. Toward this, all podcasts from a specific speaker are set aside as the validation set and the remaining speaker's podcasts become the training set. We retrain our breath detection model using the training set and calculate the AUPRC for the validation set. This process is repeated for each speaker and compared against Test 1 to determine generalizability between speakers.

4.1.4 Results. Figure 5 shows the results of these three tests. Our model outputs a prediction for every 50 ms of an audio file; all consecutive positive predictions are considered breath events and if a breath event prediction temporally overlaps a ground truth breath event by 33%, we classify that as a true positive. We use the average baseline AUPRC of ~0.97 from Test 1 to compare with Tests 2 and 3. Test 2 shows podcast-specific leave-one-out performance between ~0.91 and ~0.99, marking a reduction from the baseline (Test 1). Test 3 shows speaker-specific leave-one-out performance between ~0.89 and ~0.99, which is also a reduction in performance from the baseline. The performance reduction from Tests 2 and 3 is expected as in Test 1, there is training data of similar distribution to the validation in every iteration since validation is derived as a subset of each podcast as explained in Section 3.3.

Statistical tests allow for evaluating our model's performance in varied scenarios. Toward this, we employ the Mann-Whitney U test [26] for its robustness in comparing independent samples, offering insights into performance consistency without requiring normal distribution assumptions. Table 1 shows this test's results; notably, every p-value is greater than 0.05, which suggests we cannot reject the null hypothesis–indicating no significant difference between leave-one-out scenarios and the best case (Test 1), thus reinforcing the breath feature's robustness.

From Table 1, we show that almost all metrics in both leave-one-out scenarios exhibit equivalence, as assessed by the two one-sided tests [35] with an equivalence margin of 0.05. Equivalence is determined by the CI of the mean differences falling within this predefined margin, suggesting that the performance differences are practically insignificant. This affirms that the breath feature consistently performs well, even amidst variations in the test data, underscoring its reliability as a feature.

We observe an AUPRC mean and standard deviation of 0.969 ± 0.010, 0.964 ± 0.029, and 0.951 ± 0.037 for Tests 1, 2, and 3, respectively. This consistency supports the conclusion that breaths are generalizable, serving as a reliable discriminator for detecting deepfake speech (*RQ1*).

Table 1. Statistical Analysis for Tests 2 and 3 vs. Test 1 Using the Mann-Whitney U Statistic and the Two One-Sided Tests

Leave-One-Out Speaker					
Metric	**Mann-Whitney U**		**Two One-Sided Tests**		
	Statistic	**p-Value**	**Mean Difference**	**95% CI**	**quivalence**
Accuracy	150.0	0.1037	0.0239	(0.0089, 0.0389)	True
Positive Precision	53.0	0.1280	−0.0025	(−0.0071, 0.0020)	True
Positive Recall	150.0	0.1037	0.0505	(0.0209, 0.0802)	False
Positive F1 Score	150.0	0.1037	0.0311	(0.0133, 0.0488)	True
Positive AUROC	146.0	0.1369	0.0367	(0.0229, 0.0505)	False
Positive AUPRC	124.0	0.4525	0.0181	(0.0033, 0.0329)	True
Negative Precision	150.0	0.1037	0.0326	(0.0104, 0.0548)	False
Negative Recall	53.0	0.1280	−0.0028	(−0.0066, 0.0010)	True
Negative F1 Score	150.0	0.1037	0.0187	(0.0058, 0.0316)	True
Negative AUROC	146.0	0.1369	0.0367	(0.0229, 0.0505)	False
Negative AUPRC	147.0	0.1280	0.0865	(0.0574, 0.1155)	False

Leave-One-Out Podcast					
Metric	**Mann-Whitney U**		**Two One-Sided Tests**		
	Statistic	**p-Value**	**Mean Difference**	**95% CI**	**Equivalence**
Accuracy	181.0	0.1769	−0.0122	(−0.0246, 0.0003)	True
Positive Precision	189.0	0.2341	−0.0003	(−0.0041, 0.0035)	True
Positive Recall	169.0	0.1112	−0.0244	(−0.0494, 0.0007)	True
Positive F1 Score	178.0	0.1584	−0.0123	(−0.0270, 0.0024)	True
Positive AUROC	294.0	0.3939	0.0045	(−0.0025, 0.0114)	True
Positive AUPRC	262.0	0.8223	0.0049	(−0.0057, 0.0154)	True
Negative Precision	169.0	0.1112	−0.0226	(−0.0415, −0.0036)	True
Negative Recall	191.0	0.2501	0.0000	(−0.0034, 0.0034)	True
Negative F1 Score	183.0	0.1902	−0.0117	(−0.0225, −0.0009)	True
Negative AUROC	294.0	0.3939	0.0045	(−0.0025, 0.0114)	True
Negative AUPRC	271.0	0.6887	0.0131	(−0.0022, 0.0285)	True

We show that, regardless of metric, no p-value is below 0.05 and most metrics are equivalent to Test 1 with standard deviations within ±0.05. Both of these suggest that when a podcast or speaker is excluded from training, the performance remains consistent, affirming breath as generalizable.

4.2 Deepfake Speech Detection (RQ2 and RQ3)

4.2.1 Setup. Now that we have shown the generalizability of breaths between individuals, we apply this methodology as a discriminator for deepfake and real speech using the three methods described in Section 3.4. First, we train our breath detection model on all the podcast data to ensure the best possible final model. Next, we pass the news article data outlined in Section 3.2.4 to this final model to get breath locations for each news article. We then use the breath locations and calculate the three features detailed in Section 3.2. Figure 6 shows the clear delineation between the deepfake and real news article breath features. We demonstrate that deepfake speech, as seen in the wild today, *does not appropriately produce breaths*, which indicates a strong discriminator using these features (*RQ2*). Finally, we split the news articles into a training and testing set where no samples from the same news outlet are in the training and test sets. This split comes out to 101 training samples (18.9 hours) and 232 testing samples (33.6 hours) and ensures no bias when testing.

For the poly SVC and decision tree methodologies, we train our model (Section 3.4) with the training set and evaluate performance on the test set. Whereas for the thresholding methodology, we do not need to split the news articles into a training and test set as there is no training taking place since this is not a machine learning model;

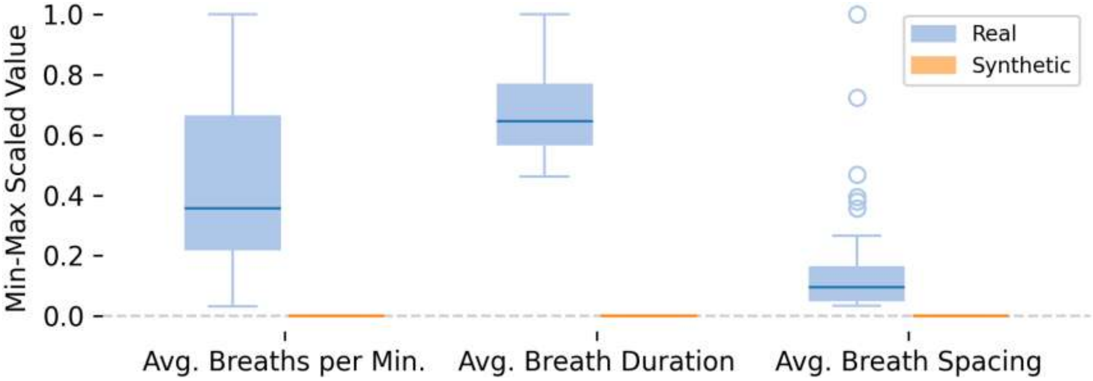

Fig. 6. We show that there is a clear distinction (i.e., no overlap) between human-read and synthetically generated news articles with respect to breath statistics; showcasing breath statistics are a strong discriminator.

however, we show results for both the training and test sets to compare against the poly SVC and decision tree methodologies. Observing Figure 6, we show that in each calculated metric real samples have a non-zero value. We use this as our guideline and set the threshold for each metric to be $Value > 0.0 = Real, Value <= 0.0 = Fake$. For this simple thresholding, we require that all features (i.e., avg. bpm, avg. breath spacing, and avg. breath duration) be > 0.0 to be considered real. Stated concisely, for any sample feature that contains a zero value (i.e., no breaths present) that sample is considered fake.

4.2.2 Results. Table 2 shows the training and test results for the news article dataset. We show that a simple SVC, decision tree, or thresholding model can perfectly discriminate between real/deepfake speech samples.[2] Furthermore, in Table 2, we show that when performing **leave-one-out cross-validation (LOO-CV)** with the SVC and decision tree models, we achieve perfect results, thus showing that our models are not overfit.

The simple reason that all of these models produce perfect results is that current deepfake audio simply does not produce breathing sounds and using breaths, we correctly predict all deepfake samples without falsely predicting any real samples (*RQ3*). Additionally, we show performant results on a varying distribution of data, as the generation techniques for each of the news articles are unknown. As such, we posit that simple thresholding is sufficient to capture the nuance in breathing for real and synthetically generated audio samples (*RQ3*). Through this, we prove that with our methodology, few, or even no, training samples are required.

4.2.3 Experimental Comparison to Other Detectors. To fully contextualize the performance of our breath detector, we compare our models against a highly complex and heavily trained model: SSL-wav2vec2.0. Toward this, we implement the XLS-R-based deepfake detector [41], as this model is the current best performer on the ASVspoof 2021 dataset by a substantial margin. While we do not use the ASVspoof 2021 dataset in any capacity, the XLS-R model was pretrained on 436,000 hours of non-ASVspoof speech data and is sufficiently performant on non-ASVspoof datasets.

We deploy the pretrained ($wav2vec-_P$) model and additionally, to ensure a fair comparison, we fine-tune SSL-wav2vec2.0 with additional training ($wav2vec-_R$). We implement this fine-tuning by augmenting the SSL-wav2vec2.0 training data with the 18.9 hours of news articles from our training set, as such the SSL-wav2vec2.0 model is given the best chance to obtain performant results. Furthermore, this is the same methodology used to retrain SSL-wav2vec2.0 for ASVspoof 2021.

[2] We note that AUPRC may not be 1.00 for a perfect model, due to it being calculated based on a sliding threshold of probabilities, and these models each have a small portion where moving the threshold changes predictions.

Table 2. Results for All Simple Detection Models (Poly SVC, Decision Tree, and Thresholding)

	Model	Dataset	EER	AUPRC	Accuracy	F1	Precision	Recall	TP	FP	TN	FN	Inference Time (CPU Seconds)
Simple models	$Poly\ \ SVC$	Train	0.00	1.00	1.00	1.00	1.00	1.00	77	0	24	0	–
	$(OurModel)$	Test	0.00	1.00	1.00	1.00	1.00	1.00	205	0	27	0	717
		LOO-CV	0.00	1.00	1.00	1.00	1.00	1.00	282	0	51	0	–
	$Decision\ \ Tree$	Train	0.00	0.99	1.00	1.00	1.00	1.00	77	0	24	0	–
	$(OurModel)$	Test	0.00	1.00	1.00	1.00	1.00	1.00	205	0	27	0	717
		LOO-CV	0.00	0.99	1.00	1.00	1.00	1.00	282	0	51	0	–
	$Thresholding$	Train	–	–	1.00	1.00	1.00	1.00	77	0	24	0	–
	$(OurModel)$	Test	–	–	1.00	1.00	1.00	1.00	205	0	27	0	717
Complex models	$wav2vec{-}_P$	Train	–	–	–	–	–	–	–	–	–	–	–
	$(Pretrained)$	Test	0.11	0.72	0.90	0.57	0.95	0.56	205	24	3	0	3,083
	$wav2vec{-}_R$	Train	0.06	0.96	0.95	0.95	0.97	0.91	72	0	24	5	–
	$(Retrained)$	Test	0.99	0.52	0.10	0.09	0.05	0.44	0	2 3	24	205	3,323
	$Codecfake{-}_P$	Train	–	–	–	–	–	–	–	–	–	–	–
	$(Pretrained)$	Test	0.00	0.99	1.00	1.00	1.00	1.00	205	0	27	0	73,198

With all models, we obtain perfect results on all metrics for the training and testing sets. We demonstrate that the pretrained wav2vec model falls short compared to our model in every metric and augmenting the train set with "in-distribution" data and retraining does not solve these shortcomings. Furthermore, we show that the new detector Codecfake also achieves perfect results on the testing set, at great computational/temporal costs compared to our models (we Infer 102 times quicker than Codecfake).

We pass the test set of news articles to both versions of SSL-wav2vec (in 4-second chunks, due to model requirements) and use a probability soft voting scheme to get a single prediction for each audio file. Table 2 shows the results of these models and shows that our model outperforms in every metric. The pretrained SSL-wav2vec predicts nearly the entire test set as deepfake speech whereas the retrained model predicts nearly all the test data as real speech (while predicting the train set with 95% accuracy). The highlighted sections in Table 2 show the classification decision for all news articles in the test set for each of the model types (i.e., SVC, Decision Tree, SSL-wav2vec2.0). We note that $wav2vec{-}_R$ predicts all news articles as real and may seem as though the training phase was incorrectly handled; however, looking at the results of $wav2vec{-}_R$ on the training data shows that it indeed picks up the signal in the data. Through this, we show that our models perform substantially better in every category than the complex models, specifically looking at the false positives it is clear to see that alarm fatigue [12] would overwhelm $wav2vec{-}_P$. Comparing this to our models which have essentially no false positives, it is clear to see the potential shortcomings of complex models.

As the thesis of our paper argues, we demonstrate that models focused on low-level speech features can completely fail when tasked with predicting new data as seen by the contradicting results between $wav2vec{-}_P$ and $wav2vec{-}_R$. Simply put, using a complex and highly trained model trained on low-level speech features provides unpredictable results, even when augmenting additional "in-distribution" training data whereas higher-level speech features can help prevent failure.

Building on this analysis, the limitations of models such as SSL-wav2vec2.0 highlight the challenges of relying on complex architectures primarily focused on low-level speech features and underscore the necessity for robustness beyond training data intricacies. In response to advancements, we revisit our comparative analysis with a newer model—Codecfake [46]. We select the Codecfake model for its recent emergence in the field, its publicly availability, and its use of a novel paradigm of employing dual training across vocoder and codec-based deepfake generators. The bottom portion of Table 2 shows the results of the pretrained Codecfake model in our test scenario. In contrast to SSL-wav2vec, Codecfake's pretrained model achieves perfect results on our test news articles.

However, it is important to note that Codecfake's computational demands and extended inference time present significant practical limitations. Unlike our model, which predicts in real time, the Codecfake model requires several hours to process data. The right-most column in Table 2 shows the CPU time required to predict the entire test set for each model. We show that while Codecfake achieves perfect predictions, it takes 102 times longer than our model requires. Reframing this, it would take Codecfake ~37 seconds (i.e., not real-time) to predict, whereas our model would only require 0.3 seconds (i.e., real-time). This issue is further compounded by the fact that our tests of Codecfake employed the use of multiple GPUs, which without these would drastically lengthen the prediction time. On the other hand, our model achieves real-time predictions without a GPU. This computational and temporal inefficiency restricts the model's flexibility and usability in the real world.

4.3 Summary of RQs

Through our experiments, we show that breaths are automatically and accurately detectable between the same speaker and throughout different speakers (*RQ1*), current in-the-wild deepfake speech does not produce breaths (*RQ2*), and using breaths as a discriminator between real and fake samples obtains performant results (*RQ3*).

5 Discussion

In this section, we discuss additional material that is important to the fundamental science of machine learning and deepfake discrimination.

5.1 On Our Model's Performance

While achieving perfect classification results (1.0 AUPRC and 0.0 EER) might raise concerns of overfitting, our findings are not attributed to such issues. If we were consistently obtaining 100% accuracy on our breath detector alone, it would warrant further scrutiny. However, our results indicate a clear distinction between real and fake speech samples based on breath features.

Current deepfake speech models frequently fail to incorporate natural breath patterns, and this characteristic forms the basis of our model's effectiveness. To additionally ensure that our results are not due to overfitting, we employ LOO-CV, which ensures the model is able to train and test on every sample.

Moreover, evidence supporting our hypothesis comes from the performance of a simple thresholding mechanism that achieves perfect accuracy without any training. This further proves that the observed delineation between real and fake speech is robust and not a result of complex model fitting. Such clear distinctions underscore the validity of our results, affirming that the absence of breath features in deepfake speech is a reliable discriminator.

5.2 Over-Reliance on and Shortcomings of Deep Learning Models

While complex deep learning models have achieved impressive results in various domains, their application in deepfake detection has revealed notable shortcomings. These models often rely heavily on large datasets and intricate architectures, making them susceptible to overfitting and unpredictable performance on unseen data. For instance, we show that models like SSL-wav2vec2.0 have variability in detection accuracy, especially when confronted with novel or out-of-distribution inputs. Such dependencies on complex feature extraction can lead to failures when adaptability is required. Furthermore, these models often require extensive computational setups and large time investments for training and inference.

This article highlights the over-reliance on these intricate models, suggesting that simpler, feature-based approaches can achieve comparable, if not superior, performance. We suggest that by focusing on high-level speech features, detection systems are more resilient to variability and distribution shifts.

Potential alternatives include hybrid models that integrate both deep learning and hand-crafted feature techniques, as well as lightweight algorithms capable of real-time processing. This shift toward a balanced approach aims to enhance robustness and efficiency, addressing the limitations inherent in purely complex deep learning frameworks.

5.3 Reproducibility

Reproducibility is a growing concern, especially for machine learning [32]. To alleviate these issues and to aid future work and comparison, we make efforts to increase the reproducibility of our work. First, we publish the code and framework we use for our entire pipeline.[3] Second, we publish the trained models and raw model scores files. We release the trained model and scores file as an additional artifact as it has been shown that retraining the same model on different GPUs may produce different results [2]. This gives future researchers the ability to calculate *any and all* metrics that may be desired for comparison.

5.4 Limitations and Future Work

If these deepfake samples start producing frequent breathing, our relatively simple discriminators will likely produce worse results. To accommodate this, a shift toward natural language processing could be combined with our work. This combination would allow contextualization between the breaths that are identified and the intra-speech location relative to other parts of speech. This contextualization would minimize false positive breath detection and improve the results of a deepfake detector.

Additionally, we acknowledge the importance of dataset diversity in enhancing the generalization and robustness of detection models. While our current dataset primarily consists of four podcast speakers (one female and three males, all native English speakers) and news articles from four different outlets, a more varied dataset would indeed be beneficial, particularly for analyzing additional high-level speech features.

Our focus, however, is specifically on breath detection in the context of deepfake speech. The primary feature—the presence or absence of breath—is less influenced by accent or speaker variability compared to other speech characteristics. That said, incorporating a wider range of speakers, accents, and environments in future work could further substantiate our findings by ensuring robust performance across diverse conditions. Currently, our dataset effectively serves its purpose by highlighting the stark contrast in breath patterns between real and deepfake audio, which is integral to our detection approach.

6 Related Work

The emergence of deepfake technology, which leverages deep learning on extensive datasets of media, poses substantial threats to media authenticity. This technology can lead to unethical misuse such as impersonation and the dissemination of false information. To counteract this challenge, some research aims to introduce the concept of innate biological processes to discern between authentic human voices and cloned voices [8, 15, 22, 27]. It has been proposed that the presence or absence of certain perceptual features, such as pauses in speech, can effectively distinguish between cloned and authentic audio [22]. Furthermore, researchers have developed new mechanisms for detecting audio deepfakes by solely relying on limitations of human speech that are the result of biological constraints [8]. Specifically, vocal tract reconstruction using fluid dynamic models has shown that deepfake audio samples often model impossible or highly unlikely anatomical arrangements [8].

[3] https://github.com/SethLayton/every_breath_you_dont_take.

Mostaani and Doss [27] investigated whether breathing patterns are present in TTS and **voice conversion (VC)** algorithms using ASVspoof 2019. This work showed that TTS algorithms fail to properly generate breaths while VC algorithms seem to retain breath pattern-related information. As TTS is rapidly dwarfing VC as the main proponent of deepfake speech, this work shows promise for breath usage as a speech deepfakes discriminator. Additionally, the **breathing-talking-silence encoder (BTS-E)** [15] algorithm was proposed as an addition to existing **countermeasures (CM)** for voice spoofing attacks that use breath and silence event detection to enhance existing CM performance by up to 46%. BTS-E leverages three Gaussian Mixture Models to segment/label input audio into talking, silence, and breath segments. This segmentation is then translated into a latent feature space and combined with the last hidden layer in an existing CM to focus that CM on talking/breathing/silence events. However, the feature importance of breathing vs. silence is not explored in BTS-E, which is problematic as Müller et al. [29] identify an uneven distribution of leading and trailing silence duration in the ASVspoof 2019 and 2021 datasets, which correlate with the target prediction label. Thus, simply examining the silence patterns in the files allows for accurate classification of real/speech deepfakes. As such, it is unclear whether the breath features in BTS-E are important and the value of breaths is unknown. More critically, none of these papers make any claims on current deepfake speech deployed in the wild.

Building upon the broader context of speech Deepfake detection, researchers have explored a variety of acoustic features for distinguishing between genuine and synthetic speech. These include traditional hand-crafted features and those learned through deep learning models [24]. For example, the CQCC and its extensions have been investigated for spoofing detection [42, 48]. Additionally, the use of deep learning architectures such as DNNs, residual networks, and recurrent neural networks has become prevalent in constructing deep embeddings from both raw waveforms and extracted features [6–8, 11].

The generalizability of deepfake detection models, particularly when faced with unseen Deepfake generation techniques or data from different domains, remains a significant challenge [24]. Simply combining data from different domains for training does not always guarantee improved generalization due to potential domain mismatches [37]. Furthermore, techniques like adversarial attack defense and cross-dataset evaluation are emerging as important research topics to address these limitations [24, 28].

The reproducibility of research findings is also a growing concern in the machine learning community [32], including the field of speech Deepfake detection [24]. The lack of publicly available source code and detailed experimental settings in some publications hinders the ability of other researchers to verify and build upon existing work. Efforts to improve reproducibility, such as the public release of code and trained models, are crucial for the advancement of the field.

Our work emphasizes the practical significance of addressing in-the-wild deepfake speech, higher-level speech features, real-time predictions, and reproducibility. Specifically, we confront these gaps by using real-world data to validate breath features as deepfake discriminators and publishing our work. By prioritizing lightweight, adaptable methodologies, we aim to ensure that detection models are not only theoretically robust but also practical for diverse applications, thus enhancing the reliability and utility of breath-based detection strategies in a rapidly evolving field.

7 Conclusion

Deepfake speech generation advancements are reducing the gap between human-spoken and human-sounding audio. Deepfake speech will become imperceivable to human speech, as such a focus needs to be placed on *how* this speech is performed. Toward this, we employ a multi-tiered pipeline that focuses on breathing to discriminate between real/deepfake speech samples. We show that breaths are generalizable between speakers and that simple calculated breath features provide accurate classification results (perfect 1.0 AUPRC and 0.0 EER) on a dataset of in-the-wild real and synthetically generated speech. Furthermore, we show the shortcomings of complex deep learning models with failures to classify the same in-the-wild samples (0.72 AUPRC and 0.89 EER) and substantial computational and temporal performance reductions compared to our methodology (37 seconds to predict a 1 minute sample with Codecfake vs. 0.3 seconds with our model).

Acknowledgments

Finally, the authors would like to thank the members of the Program Committee for their efforts in improving our work.

References

[1] Mikey Elmers, Raphael Werner, Beeke Muhlack, Bernd Mobius, and Jurgen Trouvain. 2021. Take a breath: Respiratory sounds improve recollection in synthetic speech. In *2021 Interspeech*.

[2] Hadi Abdullah, Kevin Warren, Vincent Bindschaedler, Nicolas Papernot, and Patrick Traynor. 2021. Sok: The faults in our ASRS: An overview of attacks against automatic speech recognition and speaker identification systems. In *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 730–747.

[3] Shahin Amiriparian and Björn Schuller. 2018. Deep convolutional recurrent neural network for rare acoustic event detection. In *Proceedings of DAGA*.

[4] Daniel Arp, Erwin Quiring, Feargus Pendlebury, Alexander Warnecke, Fabio Pierazzi, Christian Wressnegger, Lorenzo Cavallaro, and Konrad Rieck. 2022. Dos and don'ts of machine learning in computer security. In *31st USENIX Security Symposium (USENIX Security '22)*, 3971–3988.

[5] Luigi Attorresi, Davide Salvi, Clara Borrelli, Paolo Bestagini, and Stefano Tubaro. 2022. Combining automatic speaker verification and prosody analysis for synthetic speech detection. arXiv:2210.17222. Retrieved from https://arxiv.org/abs/2210.17222

[6] B. T. Balamurali, Kinwah Edward Lin, Simon Lui, Jer-Ming Chen, and Dorien Herremans. 2019. Toward robust audio spoofing detection: A detailed comparison of traditional and learned features. *IEEE Access* 7 (2019), 84229–84241.

[7] Jordan J. Bird and Ahmad Lotfi. 2023. Real-time detection of ai-generated speech for deepfake voice conversion. arXiv:2308.12734. Retrieved from https://arxiv.org/abs/2308.12734

[8] Logan Blue, Kevin Warren, Hadi Abdullah, Cassidy Gibson, Luis Vargas, Jessica O'Dell, Kevin Butler, and Patrick Traynor. 2022. Who are you (I really wanna know)? detecting audio {DeepFakes} through vocal tract reconstruction. In *31st USENIX Security Symposium (USENIX Security '22)*, 2691–2708.

[9] Norbert Braunschweiler and Langzhou Chen. 2013. Automatic detection of inhalation breath pauses for improved pause modelling in HMM-TTS. In The ISCA Speech Synthesis Workshop.

[10] Niko Brümmer, Luciana Ferrer, and Albert Swart. 2021. Out of a hundred trials, how many errors does your speaker verifier make? arXiv:2104.00732. Retrieved from https://arxiv.org/abs/2104.00732

[11] Zexin Cai, Weiqing Wang, Yikang Wang, and Ming Li. 2023. The dku-dukeece system for the manipulation region location task of add. arXiv:2308.10281. Retrieved from https://arxiv.org/abs/2308.10281

[12] Maria Cvach. 2012. Monitor alarm fatigue: An integrative review. *Biomedical Instrumentation & Technology* 46, 4 (2012), 268–277.

[13] Héctor Delgado, Nicholas Evans, Tomi Kinnunen, Kong Aik Lee, Xuechen Liu, Andreas Nautsch, Jose Patino, Md Sahidullah, Massimiliano Todisco, Xin Wang, et al. 2021. Asvspoof 2021: Automatic speaker verification spoofing and countermeasures challenge evaluation plan. arXiv:2109.00535. Retrieved from https://arxiv.org/abs/2109.00535

[14] Gauri Deshpande and Björn Schuller. 2020. An overview on audio, signal, speech, & language processing for COVID-19. arXiv:2005.08579. Retrieved from https://arxiv.org/abs/2005.08579

[15] Thien-Phuc Doan, Long Nguyen-Vu, Souhwan Jung, and Kihun Hong. 2023. BTS-E: Audio deepfake detection using breathing-talking-silence encoder. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.

[16] Guang-Hui Fu, Lun-Zhao Yi, and Jianxin Pan. 2019. Tuning model parameters in class-imbalanced learning with precision-recall curve. *Biometrical Journal. Biometrische Zeitschrift* 61, 3 (2019), 652–664.

[17] Eric E. Hamke, Ramiro Jordan, and Manel Ramon-Martinez. 2016. Breath activity detection algorithm. arXiv:1602.07767. Retrieved from https://arxiv.org/abs/1602.07767

[18] Trevor Hastie, Robert Tibshirani, Jerome H. Friedman, and Jerome H. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Vol. 2. Springer.

[19] Brian Hosler, Davide Salvi, Anthony Murray, Fabio Antonacci, Paolo Bestagini, Stefano Tubaro, and Matthew C. Stamm. 2021. Do deepfakes feel emotions? A semantic approach to detecting deepfakes via emotional inconsistencies. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1013–1022.

[20] Jee-Weon Jung, Hemlata Tak, Hye-Jin Shim, Hee-Soo Heo, Bong-Jin Lee, Soo-Whan Chung, Ha-Jin Yu, Nicholas Evans, and Tomi Kinnunen. 2022. SASV 2022: The first spoofing-aware speaker verification challenge. arXiv:2203.14732. Retrieved from https://arxiv.org/abs/2203.14732

[21] Tomi Kinnunen, Md Sahidullah, Héctor Delgado, Massimiliano Todisco, Nicholas Evans, Junichi Yamagishi, and Kong Aik Lee. 2017. The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection. In *ISCA (The International Speech Communication Association)*.

[22] Nikhil Valsan Kulangareth, Jaycee Kaufman, Jessica Oreskovic, and Yan Fossat. 2024. Investigation of deepfake voice detection using speech pause patterns: Algorithm development and validation. *JMIR Biomedical Engineering* 9 (2024), e56245.

[23] Seth Layton, Tyler Tucker, Daniel Olszewski, Kevin Warren, Kevin Butler, and Patrick Traynor. 2024. SoK: The good, the bad, and the unbalanced: Measuring structural limitations of deepfake datasets. In *USENIX Security Symposium (Security)*.

[24] Menglu Li, Yasaman Ahmadiadli, and Xiao-Ping Zhang. 2025. A survey on speech deepfake detection. *ACM Computing Surveys* 57 (2025), 1–38.

[25] Jaime Lorenzo-Trueba, Fuming Fang, Xin Wang, Isao Echizen, Junichi Yamagishi, and Tomi Kinnunen. 2018. Can we steal your vocal identity from the internet?: Initial investigation of cloning obama’s voice using GAN, WaveNet and low-quality found data. In *Speaker and Language Recognition Workshop*, 240–247.

[26] Henry B. Mann and Donald R. Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics* 18 (1947), 50–60.

[27] Zohreh Mostaani and Mathew Magimai Doss. 2022. On breathing pattern information in synthetic speech. In *INTERSPEECH 2022*.

[28] Nicolas M. Müller, Pavel Czempin, Franziska Dieckmann, Adam Froghyar, and Konstantin Böttinger. 2022. Does audio deepfake detection generalize? arXiv:2203.16263. Retrieved from https://arxiv.org/abs/2203.16263

[29] Nicolas M. Müller, Franziska Dieckmann, Pavel Czempin, Roman Canals, Konstantin Böttinger, and Jennifer Williams. 2021. Speech is silver, silence is golden: What do ASVspoof-trained models really learn? arXiv:2106.12914. Retrieved from https://arxiv.org/abs/2106.12914

[30] Venkata Srikanth Nallanthighal, Zohreh Mostaani, Aki Härmä, Helmer Strik, and Mathew Magimai-Doss. 2021. Deep learning architectures for estimating breathing signal and respiratory parameters from speech recordings. *Neural Networks: The Official Journal of the International Neural Network Society* 141 (2021), 211–224.

[31] Venkata Srikanth Nallanthighal and H. Strik. 2019. Deep sensing of breathing signal during conversational speech. In *Interspeech 2019*.

[32] Daniel Olszewski, Allison Lu, Carson Stillman, Kevin Warren, Cole Kitroser, Alejandro Pascual, Divyajyoti Ukirde, Kevin Butler, and Patrick Traynor. 2023. “Get in researchers; we’re measuring reproducibility”: A reproducibility study of machine learning papers in tier 1 security conferences. In *2023 ACM SIGSAC Conference on Computer and Communications Security*, 3433–3459.

[33] Britt Paris and Joan Donovan. 2019. Deepfakes and cheap fakes. In *United States of America: Data & Society 2019*.

[34] Jonathan Saunders. 2019. Detecting deep fakes with mice: Machines vs biology. In Black Hat USA.

[35] Donald J. Schuirmann. 1987. A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Biopharmaceutics* 15 (1987), 657–680.

[36] Björn W. Schuller, Anton Batliner, Christian Bergler, Eva-Maria Messner, Antonia Hamilton, Shahin Amiriparian, Alice Baird, Georgios Rizos, Maximilian Schmitt, Lukas Stappen, et al. 2020. The INTERSPEECH 2020 computational paralinguistics challenge: Elderly emotion, breathing & masks. In INTERSPEECH *2020*, ISCA, 2042–2046.

[37] Hye-Jin Shim, Jee-Weon Jung, and Tomi Kinnunen. 2023. Multi-dataset co-training with sharpness-aware optimization for audio anti-spoofing. arXiv:2305.19953. Retrieved from https://arxiv.org/abs/2305.19953

[38] Shridatt Sugrim, Can Liu, Meghan McLean, and Janne Lindqvist. 2019. Robust performance metrics for authentication systems. In *Network and Distributed Systems Security (NDSS) Symposium 2019*.

[39] Éva Székely, Gustav Eje Henter, and Joakim Gustafson. 2019. Casting to corpus: Segmenting and selecting spontaneous dialogue for TTS with a CNN-LSTM speaker-dependent breath detector. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP ’19)*. IEEE, 6925–6929.

[40] Éva Székely, Gustav Eje Henter, Jonas Beskow, and Joakim Gustafson. 2020. Breathing and speech planning in spontaneous speech synthesis. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP ’20)*, 7649–7653.

[41] Hemlata Tak, Massimiliano Todisco, Xin Wang, Jee-Weon Jung, Junichi Yamagishi, and Nicholas Evans. 2022. Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation. arXiv:2202.12233. Retrieved from https://arxiv.org/abs/2202.12233

[42] Massimiliano Todisco, Héctor Delgado, and Nicholas W. D. Evans. 2016. A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients. In Odyssey *2016*.

[43] Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee. 2019. ASVspoof 2019: Future horizons in spoofed and fake audio detection. arXiv:1904.05441. Retrieved from https://arxiv.org/abs/1904.05441

[44] Kevin Warren, Tyler Tucker, Anna Crowder, Daniel Olszewski, Allison Lu, Caroline Fedele, Magdalena Pasternak, Seth Layton, Kevin Butler, Carrie Gates, et al. 2024. “Better Be computer or I’m dumb”: A large-scale evaluation of humans as audio deepfake detectors. In *ACM Conference on Computer and Communications Security (CCS)*.

[45] Zhizheng Wu, Tomi Kinnunen, Nicholas Evans, Junichi Yamagishi, Cemal Hanilçi, Md Sahidullah, and Aleksandr Sizov. 2015. ASVspoof 2015: The first automatic speaker verification spoofing and countermeasures challenge. In *16th Annual Conference of the International Speech Communication Association*.

[46] Yuankun Xie, Yi Lu, Ruibo Fu, Zhengqi Wen, Zhiyong Wang, Jianhua Tao, Xin Qi, Xiaopeng Wang, Yukun Liu, Haonan Cheng, et al. 2025. The codecfake dataset and countermeasures for the universally detection of deepfake audio. *IEEE Transactions on Audio, Speech and Language Processing* 33 (2025), 386–400.

[47] Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, et al. 2021. ASVspoof 2021: Accelerating progress in spoofed and deepfake speech detection. arXiv:2109.00537. Retrieved from https://arxiv.org/abs/2109.00537

[48] Jichen Yang, Rohan Kumar Das, and Nina Zhou. 2019. Extraction of octave spectra information for spoofing attack detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27 (2019), 2373–2384.

[49] Jiangyan Yi, Ruibo Fu, Jianhua Tao, Shuai Nie, Haoxin Ma, Chenglong Wang, Tao Wang, Zhengkun Tian, Ye Bai, Cunhang Fan, et al. 2022. Add 2022: The first audio deep synthesis detection challenge. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 9216–9220.

Received 18 April 2024; revised 25 March 2025; accepted 19 June 2025