Synthesized Speech Attribution Using The Patchout Spectrogram Attribution Transformer

Kratika Bhagtani
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, USA
kbhagtan@purdue.edu

Emily R. Bartusiak
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, USA
ebartusi@purdue.edu

Amit Kumar Singh Yadav
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, USA
yadav48@purdue.edu

Paolo Bestagini
Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano
Milan, Italy
paolo.bestagini@polimi.it

Edward J. Delp
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, USA
ace@ecn.purdue.edu

ABSTRACT

The malicious use of synthetic speech has increased with the recent availability of speech generation tools. It is important to determine whether a speech signal is authentic (spoken by a human) or is synthesized and to determine the generation method used to create it. Identifying the synthesis method is known as synthetic speech attribution. In this paper, we propose the use of a transformer deep learning method that analyzes mel-spectrograms for synthetic speech attribution. Our method known as Patchout Spectrogram Attribution Transformer (PSAT) can distinguish new, unseen speech generation methods from those seen during training. PSAT demonstrates high performance in attributing synthetic speech signals. Evaluation on the DARPA SemaFor Audio Attribution Dataset and the ASVSpoof2019 Dataset shows that our method achieves more than 95% accuracy in synthetic speech attribution and performs better than existing deep learning approaches.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; **Machine learning algorithms**; *Cross-validation*; • **Applied computing** → **Sound and music computing**; **System forensics**; • **Hardware** → **Digital signal processing**.

KEYWORDS

deep learning, audio forensics, synthetic speech, transformers, mel-spectrograms

ACM Reference Format:
Kratika Bhagtani, Emily R. Bartusiak, Amit Kumar Singh Yadav, Paolo Bestagini, and Edward J. Delp. 2023. Synthesized Speech Attribution Using The Patchout Spectrogram Attribution Transformer. In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '23), June 28–30, 2023, Chicago, IL, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3577163.3595112

IH&MMSec '23, June 28–30, 2023, Chicago, IL, USA

ACM ISBN 979-8-4007-0054-5/23/06.
https://doi.org/10.1145/3577163.3595112

1 INTRODUCTION

Several recent techniques, such as Wavenet and Tacotron 2, have been proposed to generate high quality synthetic speech [26, 31] from text. Text-to-speech (TTS) systems that produce human-like speech are widely used in many applications [1, 5] such as Siri.

Synthesized speech can also be used maliciously. Synthetic speech has been used to spread misinformation [4] and to deceive Automatic Speaker Verification (ASV) systems [11]. Therefore, it is important to develop speech forensic methods that can detect generated speech. In addition to detection, the ability to attribute synthetic speech to its generation method can provide valuable information about its source. Methods used to attribute cameras [14], microphones [21], and synthetic images [22] have proved useful in determining the creators of unauthorized content. Most audio analyses focus on synthesized speech detection [12, 20, 36], we extend this analysis further and investigate synthetic speech attribution.

In this paper, we propose a mel-spectrogram transformer method known as the Patchout Spectrogram Attribution Transformer (PSAT), for synthetic speech attribution in both closed set and open set scenarios. Our approach is based on the Patchout faSt Spectrogram Transformer (PaSST) that was originally proposed for audio recognition [19]. In the closed set scenario, we attribute synthetic speech to one of the generation methods included in a training dataset. In the open set scenario, our method also discriminates between *unknown* and *known* synthesis methods. In other words, PSAT attributes all speech signals from *known* synthesizers, and additionally identifies any signal from new speech synthesis techniques as being generated by an *unknown* synthesizer. Thus, one additional category describes all *unknown* speech synthesis techniques. Our method extends PaSST [19] to the open set attribution scenario. We demonstrate the performance of our approach on two datasets and compare with two synthetic speech attribution methods. In many cases, speech signals recorded at one sampling rate (*e.g.*, 16KHz) can also be sampled at low sampling rates (*e.g.*, 8KHz) for attribution analysis. We examine this situation in our experiments where 8KHz

is chosen as the lowest sampling rate to represent telephone quality speech. The contributions of this paper are as follows. We propose a mel-spectrogram transformer known as PSAT for synthetic speech attribution. It can distinguish between *known* and *unknown* generators. If PSAT is trained using only synthetic speech from multiple speech generators and if a bonafide (authentic) speech is evaluated (even chosen from a different dataset distribution than the training dataset), PSAT can detect the signal as bonafide.

2 RELATED WORK

Synthetic speech generation has a long history [18]. For example, Schröder *et al.* described a method that uses waveform concatenation and Hidden Markov Models (HMM) for speech generation [25]. Matrouf *et al.* proposed a voice conversion system based on spectral filtering [23]. Synthesized speech from traditional methods can in many cases be distinguished from genuine human speech because it is robotic, lower in pitch, and unnaturally paced [18, 28]. Recently, deep learning methods for synthetic speech generation have become popular because they can generate synthetic speech that is indiscernible from authentic human speech [3]. Oord *et al.* proposed WaveNet, which is a TTS method that uses a neural network to generate speech waveforms [31]. Tacotron 2, another TTS neural network, uses a recurrent prediction network for creation of mel-spectrograms, which are then used to create speech waveforms by a modified WaveNet network [26].

Many methods have been proposed for synthetic speech detection [8, 9, 12, 36]. Some methods use transform coefficients as features, such as Constant-Q Cepstral Coefficients (CQCC) [30], and Gammatone Cepstral Coefficient (GTCC) [16] for synthetic speech detection. Other methods use speech waveforms along with a neural network to detect generated speech [13, 32]. There are methods which represent the speech signal as a spectrogram or a mel-spectrogram and use it as an input image to a neural network for synthetic speech detection [34]. Bartusiak *et al.* use the spectrogram representation of speech as an input to a Compact Convolutional Transformer (CCT) for synthetic speech detection [6].

Mel-spectrogram and transformer approaches have shown success in various types of audio tasks [33]. Koutini *et al.* and Gong *et al.* use mel-spectrograms for audio classification using PaSST [19] and Audio Spectrogram Transformer [15], respectively. In this paper, we extend PaSST for synthetic speech attribution. We use PaSST because of its high performance in audio recognition [19].

For synthetic speech attribution, Borrelli *et al.* used handcrafted Short-Term Long-Term (STLT) and Bicoherence features of the signal (this method is denoted as STLT+Bico) [10]. Bartusiak *et al.* proposed a spectrogram transformer approach known as the Compact Attribution Transformer (CAT) for synthetic speech attribution [7]. We compare our method with both these approaches.

3 SYNTHETIC SPEECH ATTRIBUTION

In this section, we describe our system for synthetic speech attribution known as Patchout Spectrogram Attribution Transformer (PSAT). First we present the architecture adapted from PaSST. Then we show how to adapt PaSST for an open set scenario.

3.1 Patchout faSt Spectrogram Transformer (PaSST)

In this section, we describe the architecture used in the PSAT system. The block diagram of PSAT is shown in Figure 1. PSAT uses the architecture of the Patchout faST Spectrogram Transformer (PaSST), which was originally proposed for audio recognition [19]. PSAT uses an input speech signal of fixed length T seconds ($T = 5$ seconds in our experiments). We denote the input speech signal as $\mathbf{x}$, where $\mathbf{x}$ is a $T \times S_r$-dimensional vector, and S_r is the sampling rate in samples/second. If the signal has a different length than T, we pad it with *zeroes* or truncate it to T seconds. This T-second signal is then used as input to the PaSST architecture [19]. The output of the PaSST architecture is a set of probabilities. Our method PSAT interprets these probabilities for closed or open set attribution.

As originally proposed by Koutini *et al.* [19] for audio recognition, PaSST works as follows. We first obtain the 2D mel-spectrogram $\mathbf{X}$ of $\mathbf{x}$ following the approach proposed by Gong *et al.* [15] using the magnitude of the Short Time Fourier Transform [24]. The mel-spectrogram is a 2D representation of the speech signal in the time-frequency domain, where frequencies are in mel-scale [27]. If the frequencies are in the Hertz scale, it is called a spectrogram. The conversion between mel-scale and Hertz scale is given by:

$f_{mel} = 2595 \cdot \log_{10}(1 + \frac{f_{Hz}}{700})$, where f_{mel} and f_{Hz} are the frequencies in mel-scale and Hertz scale, respectively.

Equally distant frequencies on mel-scale correspond to equally distant pitches as perceived by the human ear. In our experiments $\mathbf{x}$ has a sampling rate of 16KHz, we use a 50 ms Hamming window with a shift of 20 ms to construct a sequence of 128 log mel features [15]. This generates a 2D mel-spectrogram of size 128×250.

Once the mel-spectrogram $\mathbf{X}$ of $\mathbf{x}$ is obtained, we input it into the PaSST transformer architecture shown in Figure 1. The output of the transformer architecture is a set of probabilities that indicate the likelihoods of the input speech signal belonging to each of the N classes. Formally, this step is the operation $\mathbf{p} = \mathcal{T}(\mathbf{x})$, where $\mathcal{T}$ is the PaSST operator. The N-dimensional vector $\mathbf{p}$ is formed such that its i^{th} element $\mathbf{p}_i$ corresponds to the probability that the input speech signal belongs to the i^{th} class.

As shown in Figure 1, the mel-spectrogram of $\mathbf{x}$ is split into overlapping patches of size 16×16. To account for the input patch order information, both time and frequency positional encodings [19] are appended to each patch. An important step in PaSST called Patchout [19] allows some of the patches to be not used in training. This reduces the computational complexity because fewer patches are analyzed [19]. Each 2D patch is passed through linear projection which flattens it to a 768-dimensional patch representation. Next, classification and distillation tokens are added to the patch representation [19]. Finally, these are passed through self-attention layers which provide the classification probabilities $\mathbf{p}$.

3.2 Closed and Open Set Attribution

Given a synthetic speech signal, we want to identify its generation method from a set of *known* methods (*i.e.*, a set of speech generators included in the training dataset). We refer to this as the closed set problem. If the speech signal was generated from a method never seen before, we want to detect it as an *unknown* method. This is known as the open set problem. More formally, the input

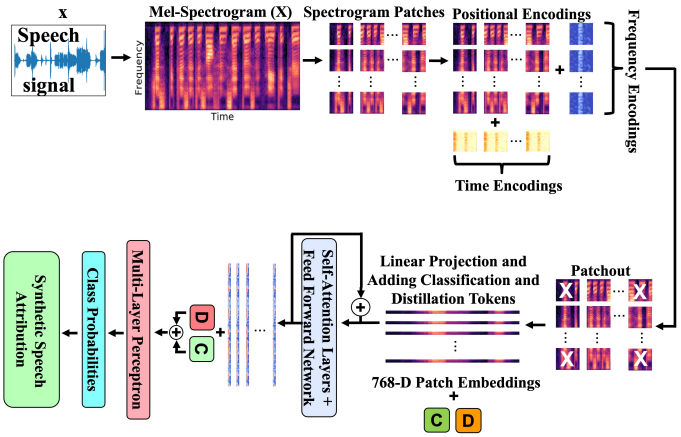

Figure 1: Block diagram of our proposed system, Patchout Spectrogram Attribution Transformer (PSAT), for synthetic speech attribution. PSAT uses the architecture of Patchout faSt Spectrogram Transformer (PaSST) [19]. Note that "C" refers to classification tokens, and "D" refers to distillation tokens [19].

signal $\mathbf{x}$ has a label y associated with it that indicates its generation method. The goal of PSAT is that, given $\mathbf{x}$, find $\hat{y}$, which is an estimate of y. We consider N generators to be *known*, which means $y \in \{1, 2, \ldots, N\}$. In the closed set scenario, our system finds a label $\hat{y} \in \{1, 2, \ldots, N\}$, thus attributing the input to a *known* class. In the open set scenario, our system finds a label $\hat{y} \in \{1, 2, \ldots, N+1\}$, where $\hat{y} = N+1$ refers to all the *unknown* generation methods.

PSAT estimates $\hat{y}$ using the classification probabilities $\mathbf{p}$ obtained from the PaSST architecture. The label $\hat{y}$ indicates the generation method used for the input speech signal $\mathbf{x}$. For closed set attribution, the dataset used for training and testing consists of N classes. We attribute the input $\mathbf{x}$ to the generator using the maximum probability found by PaSST (*i.e.*, $\hat{y} = \arg\max_i \mathbf{p}_i$). For the open set scenario, we first determine if the input comes from a *known* or *unknown* class. To do this, we evaluate the variability of the elements of $\mathbf{p}$. If the input was generated from a *known* class, we should observe only one high probability $\mathbf{p}_i$ pointing to that class, and all other probabilities in $\mathbf{p}$ should be small. Conversely, if the input was generated from an *unknown* synthetic speech generator, we should not observe any particularly large $\mathbf{p}_i$. To distinguish between these two cases, we obtain the largest and the second largest elements of $\mathbf{p}$ (*i.e.*, $\mathbf{p}_1$ and $\mathbf{p}_2$). If the ratio $\frac{\mathbf{p}_1}{\mathbf{p}_2}$ is large, we attribute the input to one of the *known* generators. Otherwise (*i.e.*, if the two largest values of $\mathbf{p}$ are comparable), we attribute it to the *unknown* class. Formally,

$$\hat{y} = \begin{cases} \arg\max_i \mathbf{p_i} & \text{if } \frac{\mathbf{p}_1}{\mathbf{p}_2} > \gamma, \\ N+1, & \text{otherwise,} \end{cases} \tag{1}$$

where γ is a threshold empirically determined on a small set of inputs $\mathbf{x}$. This method is called the Maximum Softmax Probability approach for out-of-distribution detection in machine learning [17].

4 EXPERIMENTAL RESULTS

4.1 Experimental Datasets

We use two datasets for our experiments:

DARPA SemaFor Audio Attribution Dataset (SAAD). This dataset was created by DARPA for the SemaFor program [2]. The average duration of the speech signals in this dataset is 6.9 seconds. All speech signals are sampled at 16KHz. It contains 17,000 synthetic speech signals generated from 11 different synthetic speech generators. 8 out of 11 generators are present in both the training set and the evaluation dataset. The 8 generators are Fastpitch, Fastspeech2, Glowtts, Gtts, Riva, Tacotron, Tacotron 2 and Talknet. Each generator contains 1,000 speech signals in the training set except for two, which contain 500 speech signals each. These 8 generators are considered as the *known* generators and are used for closed set attribution. The evaluation set contains 3 additional generators, which are Mixertts, Speedyspeech and Vits. All 11 generators in the evaluation set are used for open set attribution. Details about the speech generators present in this dataset are provided in [7]. This dataset does not have a validation set, so we use 5-fold cross validation. During training for each fold, we split the training set into 5,600 speech signals for training and 1,400 for validation. We save training and validation losses for all the epochs for all 5 folds. If the validation losses do not vary by more than a patience value (0.01 for our experiments) for at least 10 epochs, we stop training. We select the fold with highest validation accuracy for evaluation.

ASVSpoof2019 Dataset. We use the Logical Access (LA) part of the ASVSpoof2019 Dataset for our experiments [35]. This dataset contains genuine human speech along with synthetic speech [35]. The average duration of the speech signals in this dataset is 3.3 seconds. The distribution of ASVSpoof2019 Dataset and details about the speech generators are provided in [10, 35]. All speech signals are sampled at 16KHz. The authentic human speech signals are known as the bonafide (BF) class. Classes A01 through A19 are labels for synthetic speech generation methods. The 19 synthetic generators are WaveNet, WORLD Vocoder, Merlin, waveform concatenation by MaryTTS, Variational Auto-Encoder (VAE) and WORLD vocoder, spectral filtering, WORLD VOCODER and WaveCycleGAN2, Neural Source Filter waveform model, Vocaine Vocoder, Tacotron 2

and WaveRNN neural vocoder, Griffin-Lim waveform generator, Wavenet TTS, VC and TTS by neural network, STRAIGHT vocoder, speaker-dependent WaveNet vocoders, waveform concatenation by MaryTTS, VAE and direct waveform modification, non-parallel VC and spectral filtering. The training and development sets consist of the bonafide class and synthetic speech classes A01 to A06. The evaluation set consists of classes A07 to A19. A04 and A06 correspond to the same generation methods as A16 and A19, respectively. The difference between A04 and A16 (and likewise A06 and A19) is that they were synthesized from different training sets. These classes are the only ones common in training and evaluation sets.

4.2 Experiments and Results

We use the following performance metrics for our experiments (TP, FP, FN, TN, TPR, and TNR represent the number of True Positives, False Positives, False Negatives, True Negatives, True Positive Rate and True Negative Rate, respectively) [29]:

- **Accuracy (Acc):** $\frac{(TP+TN)}{(TP+FP+FN+TN)}$
- **Balanced Accuracy (BalAcc):** $\frac{(TPR+TNR)}{2}$
- **Precision (Prec):** $\frac{(TP)}{(TP+FP)}$
- **Recall (Rec):** $\frac{(TP)}{(TP+FN)}$
- **F1-Score (F1):** $\frac{(2 \cdot Precision \cdot Recall)}{(Precision+Recall)}$

We train two versions of our proposed method PSAT. The first version is trained from scratch on the DARPA SAAD Dataset or the ASVSpoof2019 Dataset (we denote this as PSAT_{fs}). The second version of PSAT is pretrained on the AudioSet Dataset described in PaSST [19] and fine-tuned on the DARPA SAAD Dataset or the ASVSpoof2019 Dataset (we denote this as PSAT_{pre}).

Using the DARPA SAAD Dataset, we experiment with two hyperparameters – batch size and learning rate – to achieve the best results. For these experiments, the training, validation and testing speech signals are sampled at 16KHz. From these experiments, we select training batch size 12, validation batch size 20 and learning rate 10^{-5} as the best combination. We use a frequency patches stride [19] of 3 and time patches stride [19] of 10 to create overlapping patches from mel-spectrogram. We use the AdamW optimizer with a weight decay of 10^{-4} to train PSAT_{fs} for 200 epochs and PSAT_{pre} for 100 epochs. For open set attribution, we use $\gamma = 10$.

Table 1: Closed and Open Set results from Experiment 1 using the DARPA SAAD Dataset showing that PSAT trained on 16KHz speech signals performs well when evaluated on 16KHz speech signals but performs poorly on 8KHz speech signals. S_r is the sampling rate used for evaluation.

	S_r	Closed Set					Open Set
Model	(KHz)	Acc	BalAcc	F1	Prec	Rec	Acc
PSAT_{fs}	8	64.98%	53.73%	59.26%	68.57%	64.98%	38.26%
PSAT_{pre}	8	51.75%	42.30%	47.63%	78.41%	51.75%	40.44%
PSAT_{fs}	16	91.65%	74.73%	89.15%	91.20%	91.65%	81.91%
PSAT_{pre}	16	**95.45%**	**84.59%**	**93.88%**	**92.54%**	**95.45%**	**87.19%**

We conducted four experiments. In Experiment 1, PSAT is trained on speech signals from the training set of the DARPA SAAD Dataset, which has a 16KHz sampling rate. The trained PSAT is first evaluated on speech signals from the test set of DARPA SAAD Dataset sub-sampled at 8KHz. The trained PSAT is then evaluated on speech signals in the test set of DARPA SAAD Dataset sampled at 16KHz. In Table 1, we summarize the results of Experiment 1 on the DARPA SAAD Dataset. With PSAT_{pre}, we obtain a closed set accuracy of 95.45% and an open set accuracy of 87.19% when evaluated on 16KHz speech signals. PSAT_{fs} achieves 91.65% and 81.91% closed and open set accuracy, respectively. Note that the pre-trained version does better than the version trained from scratch. We feel this is because the AudioSet Dataset is a large audio dataset and pre-training on it helps the transformer with extra knowledge about audio characteristics. PSAT in general shows very good performance in synthetic speech attribution when trained and tested on 16KHz signals. When PSAT_{fs} and PSAT_{pre} trained on speech signals with a 16KHz sampling rate are evaluated at 8KHz sampling rate, we observe a huge decline in performance. The closed set accuracy of PSAT_{pre} decreased from 95.45% to 51.75% when it was evaluated on 8KHz speech. We feel this is because while evaluating on 8KHz, we are only using low frequencies and only half of the frequency information from the speech signal as compared to that during training when 16KHz sampling rate was used.

Table 2: Closed and Open Set results from Experiment 2 using DARPA SAAD Dataset showing performance of PSAT and CAT [7]. S_r is the sampling rate used for training and evaluation.

	S_r	Closed Set					Open Set
Model	(KHz)	Acc	BalAcc	F1	Prec	Rec	Acc
CAT	16	92.53%	83.27%	91.27%	90.37%	92.53%	84.10%
PSAT_{fs}	16	91.65%	74.73%	89.15%	91.20%	91.65%	81.91%
PSAT_{pre}	16	**95.45%**	**84.59%**	**93.88%**	**92.54%**	**95.45%**	**87.19%**
PSAT_{fs}	8	91.23%	73.42%	89.62%	89.80%	91.23%	75.77%
PSAT_{pre}	8	**95.62%**	**85.81%**	**94.21%**	**93.01%**	**95.62%**	**87.79%**

Experiment 2 uses the DARPA SAAD Dataset and has two parts. For the first part, we train PSAT_{fs} and PSAT_{pre} on speech signals with a 16KHz sampling rate and evaluate on 16KHz sampled signals. For the second part, we train PSAT_{fs} and PSAT_{pre} on 8KHz sampling rate and evaluate on 8KHz sampled signals. Note that we use the same speech signals as the first part, just sub-sampled to 8KHz. We compare our performance with CAT [7] where CAT uses the original 16KHz sampling rate of the DARPA SAAD Dataset for training and evaluation. We use CAT for comparison because it also uses a spectrogram transformer. Table 2 summarizes the results of our second experiment. In both closed and open set scenarios, and for both 16KHz and 8KHz sampling rates, PSAT_{pre} performs better than CAT for all evaluation metrics. Values of all evaluation metrics for PSAT_{pre} are approximately 2-3 percentage points higher when compared to CAT. We observe that the closed set performance of both PSAT_{fs} and PSAT_{pre} is nearly unaffected when we move from 16KHz sampling rate (trained and evaluated on 16KHz signals) to a lower sampling rate 8KHz (trained and evaluated on 8KHz). There is a difference of less than 2 percentage points. PSAT_{pre} trained and evaluated on 8KHz still performs better than CAT, which is trained and evaluated on 16KHz. We can conclude that for synthetic speech attribution, we do not require high sampling rate (*e.g.*, 16KHz) and a sampling rate of 8KHz, which is telephone-quality speech, is sufficient for accurate attribution. PSAT_{pre} with 8KHz sampling

rate can be used for synthetic speech attribution even with speech signals which have higher sampling rate.

Table 3: Closed Set Results from Experiment 3 using ASVSpoof2019 Dataset showing performance of PSAT and STLT+Bico [10]. S_r is the sampling rate used for training and evaluation.

	S_r	Closed Set					Open Set
Model	**(KHz)**	**Acc**	**BalAcc**	**F1**	**Prec**	**Rec**	**Acc**
STLT + Bico	16	93.53%	93.57%	93.71%	93.90%	93.53%	85.34%
$PSAT_{fs}$	16	97.05%	96.02%	96.97%	97.37%	97.05%	85.56%
$PSAT_{pre}$	16	**99.49%**	**99.30%**	**99.49%**	**99.50%**	**99.49%**	**89.34%**
$PSAT_{fs}$	8	97.04%	96.80%	97.09%	97.22%	97.04%	85.89%
$PSAT_{pre}$	8	**99.44%**	**99.26%**	**99.44%**	**99.45%**	**99.44%**	88.78%

Experiment 3 uses the ASVSpoof2019 Dataset and also has two parts. For the first part, we train $PSAT_{fs}$ and $PSAT_{pre}$ on speech signals with 16KHz sampling rate and evaluate on 16KHz sampling rate signals. For the second part, we train $PSAT_{fs}$ and $PSAT_{pre}$ on 8KHz sampling rate and evaluate on 8KHz sampled signals. We compare our performance with STLT+Bico feature based method which uses the original 16 KHz sampling rate for the ASVSpoof2019 Dataset, for training and evaluation. [10]. Following the strategy proposed in [10], we use the development set for closed set attribution. For open set attribution, STLT+Bico [10] train their method on different dataset splits. During training, two classes from the training set A04 and A06 are labeled as Known-Unknown (KN-UNKN). Although the labels for speech signals from these classes are known, the method characterizes these speech signals as synthesized from *unknown* generators during training. The rest of the classes in the training set (BF, A01, A02, A03 and A05) are considered as *known*. For evaluation in the open set scenario, the development and evaluation sets of the ASVSpoof2019 Dataset are used. During evaluation, all the *known* generators are detected, the KN-UNKN generators are detected as *unknown*, and all the *unknown* generators are also detected as *unknown*. Generator classes A16 and A19 are also detected as *unknown* which is expected because they correspond to the same generation methods as A04 and A06, respectively. The results of our experiments on the ASVSpoof2019 Dataset are summarized in Table 3. In both closed set and open set scenarios, $PSAT_{fs}$ and $PSAT_{pre}$ achieve performance higher than STLT+Bico method, which is a feature-based approach. The open set accuracy for $PSAT_{pre}$ trained and evaluated on 16KHz speech signals is 4 percentage points higher than STLT+Bico method. This experiment also confirms that we can use $PSAT_{pre}$ with low sampling rate of 8KHz during training and evaluation for synthetic speech attribution. The comparison of confusion matrices for $PSAT_{pre}$ (trained and evaluated at 8KHz) and the STLT+Bico method in closed set scenario is shown in Figure 2. The size of the confusion matrices is 7×7 because there are 7 classes in the closed set scenario for the ASVSpoof2019 dataset (bonafide, and A01 to A06). The confusion matrices show breakdown performances of $PSAT_{pre}$ and the STLT+Bico method for each of the *known* classes. As shown in Figure 2, $PSAT_{pre}$ achieves 100% closed set accuracy for all speech generators (A01 to A06). $PSAT_{pre}$ is able to detect BF (genuine speech) speech signals with 95% accuracy.

In Experiment 4, we input BF signals (genuine speech signals) from the ASVSpoof2019 Dataset sub-sampled at 8KHz to $PSAT_{pre}$ trained on the 8KHz speech signals from the DARPA SAAD Dataset. Since the DARPA SAAD Dataset has only synthetic speech signals, we expect that evaluating on a genuine signal will result in low probabilities for all N classes. For this experiment, we choose 12,483 BF signals (2,580 from training, 2,548 from validation, and 7,355 from evaluation) from the ASVSpoof2019 Dataset. We set a threshold of 97%, which is empirically determined from the training and validation set. During testing on the evaluation set, if the maximum probability of all classes is less than this threshold, we label it as a genuine signal (BF class). If the maximum probability is higher than the threshold, we attribute the speech signal to the generator with the maximum probability. We observe that $PSAT_{pre}$ is able to identify BF (genuine speech signals) with an accuracy of 90.94%.

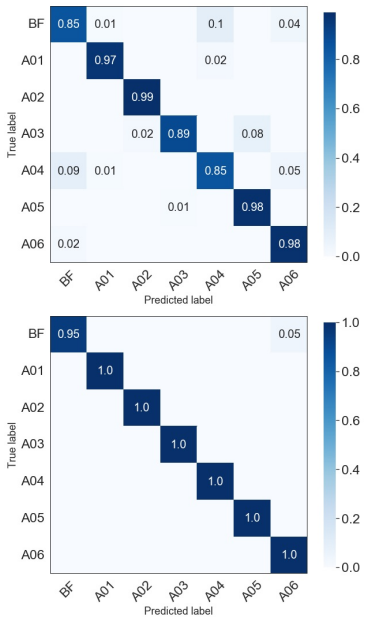

Figure 2: Confusion matrices of STLT+Bico (top) taken from [10] and $PSAT_{pre}$ trained and evaluated on 8KHz (bottom) on the ASVSpoof2019 Dataset.

In summary, we show that our method performs well for synthetic speech attribution and achieves high performance even with telephone-quality speech. We demonstrate that we do not need high sampling rates for attribution.

5 CONCLUSION

In this paper, we describe a mel-spectrogram approach known as PSAT that can successfully attribute synthetic speech signals from *known* generation methods, distinguish *unknown* generation methods from *known* ones, that performs well compared to existing deep learning approaches. In the future, we will focus on attributing long-duration speech signals that are several minutes long.

ACKNOWLEDGEMENTS

This material is based on research sponsored by DARPA and Air Force Research Laboratory (AFRL) under agreement number FA8750-20-2-1004. The U.S. Government is authorized to reproduce and

distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA and Air Force Research Laboratory (AFRL) or the U.S. Government. Address all correspondence to Edward J. Delp, ace@ecn.purdue.edu.

REFERENCES

[1] 2017. Deep Learning for Siri's Voice: On-device Deep Mixture Density Networks for Hybrid Unit Selection Synthesis. https://machinelearning.apple.com/research/siri-voices

[2] 2020. Semantic Forensics (SemaFor). https://www.darpa.mil/program/semantic-forensics

[3] 2022. AI enabled, real people's voices. https://murf.ai/

[4] 2022. Deepfake Zelenskyy surrender video is the 'first intentionally used' in Ukraine war. https://www.euronews.com/my-europe/2022/03/16/deepfake-zelenskyy-surrender-video-is-the-first-intentionally-used-in-ukraine-war

[5] 2022. Standard, WaveNet, and Neural2 voices. https://cloud.google.com/text-to-speech/docs/wavenet

[6] Emily R. Bartusiak and Edward J. Delp. 2021. Synthesized Speech Detection Using Convolutional Transformer-Based Spectrogram Analysis. *Proceedings of the IEEE Asilomar Conference on Signals, Systems, and Computers* (October 2021). https://doi.org/10.1109/IEEECONF53345.2021.9723142 Asilomar, CA.

[7] Emily R. Bartusiak and Edward J. Delp. 2022. Transformer-Based Speech Synthesizer Attribution in an Open Set Scenario. *Proceedings of the IEEE International Conference on Machine Learning and Applications* (December 2022). https://doi.org/10.48550/arXiv.2210.07546 Nassau, The Bahamas.

[8] Kratika Bhagtani, Amit Kumar Singh Yadav, Emily R. Bartusiak, Ziyue Xiang, Ruiting Shao, Sriram Baireddy, and Edward J. Delp. 2022. An Overview of Recent Work in Media Forensics: Methods and Threats. *arXiv:2204.12067* (April 2022). https://doi.org/10.48550/arXiv.2204.12067

[9] Kratika Bhagtani, Amit Kumar Singh Yadav, Emily R. Bartusiak, Ziyue Xiang, Ruiting Shao, Sriram Baireddy, and Edward J. Delp. 2022. An Overview of Recent Work in Multimedia Forensics. *Proceedings of the IEEE International Conference on Multimedia Information Processing and Retrieval* (August 2022), 324–329. https://doi.org/10.1109/MIPR54900.2022.00064

[10] Clara Borrelli, Paolo Bestagini, Fabio Antonacci, Augusto Sarti, and Stefano Tubaro. 2021. Synthetic speech detection through short-term and long-term prediction traces. *EURASIP Journal on Information Security* 2021, 1 (April 2021). https://doi.org/10.1186/s13635-021-00116-3

[11] Thomas Brewster. 2021. Fraudsters Cloned Company Director's Voice In $35 Million Bank Heist, Police Find. https://www.forbes.com/sites/thomasbrewster/2021/10/14/huge-bank-fraud-uses-deep-fake-voice-tech-to-steal-millions/?sh=6944ebd37559

[12] Tianxiang Chen, Avrosh Kumar, Parav Nagarsheth, Ganesh Sivaraman, and Elie Khoury. 2020. Generalization of Audio Deepfake Detection. *Proceedings of the Speaker and Language Recognition Workshop, Odyssey* (November 2020), 132–137. https://doi.org/10.21437/Odyssey.2020-19 Tokyo, Japan.

[13] Akash Chintha, Bao Thai, Saniat Javid Sohrawardi, Kartavya Bhatt, Andrea Hickerson, Matthew Wright, and Raymond Ptucha. 2020. Recurrent Convolutional Structures for Audio Spoof and Video Deepfake Detection. *IEEE Journal of Selected Topics in Signal Processing* 14, 5 (June 2020), 1024–1037. https://doi.org/10.1109/JSTSP.2020.2999185

[14] Filipe de O. Costa, Michael Eckmann, Walter J. Scheirer, and Anderson Rocha. 2012. Open Set Source Camera Attribution. *Proceeding of the Brazilian Symposium on Computer Graphics and Image Processing* (August 2012), 71–78. https://doi.org/10.1109/SIBGRAPI.2012.19 Ouro Preto, Brazil.

[15] Yuan Gong, Yu-An Chung, and James Glass. 2021. AST: Audio Spectrogram Transformer. *Proceedings of Interspeech* (September 2021), 571–575. https://doi.org/10.21437/Interspeech.2021-698 Brno, Czechia.

[16] Farman Hassan and Ali Javed. 2021. Voice Spoofing Countermeasure for Synthetic Speech Detection. *Proceedings of the International Conference on Artificial Intelligence* (April 2021), 209–212. https://doi.org/10.1109/ICAI52203.2021.9445238 Settat, Morocco.

[17] Dan Hendrycks and Kevin Gimpel. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *Proceedings of the International Conference on Learning Representations* (April 2017). https://openreview.net/forum?id=Hkg4TI9xl Toulon, France.

[18] Dennis H. Klatt. 1987. Review of text-to-speech conversion for English. *The Journal of the Acoustical Society of America* 82, 3 (May 1987), 737–793. https://doi.org/10.1121/1.395275

[19] Khaled Koutini, Jan Schlüter, Hamid Eghbal-zadeh, and Gerhard Widmer. 2022. Efficient Training of Audio Transformers with Patchout. *Proceedings of Interspeech* (September 2022), 2753–2757. https://doi.org/10.21437/Interspeech.2022-227 Incheon, Korea.

[20] Xu Li, Na Li, Chao Weng, Xunying Liu, Dan Su, Dong Yu, and Helen Meng. 2021. Replay and Synthetic Speech Detection with Res2Net Architecture. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (June 2021), 6354–6358. https://doi.org/10.1109/ICASSP39728.2021.9413828 Toronto, Canada.

[21] Da Luo, Paweł Korus, and Jiwu Huang. 2018. Band Energy Difference for Source Attribution in Audio Forensics. *IEEE Transactions on Information Forensics and Security* 13, 9 (March 2018), 2179–2189. https://doi.org/10.1109/TIFS.2018.2812185

[22] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. 2019. Do GANs Leave Artificial Fingerprints? *IEEE Conference on Multimedia Information Processing and Retrieval* (March 2019), 506–511. https://doi.org/10.1109/MIPR.2019.00103 San Jose, CA.

[23] Driss Matrouf, Jean-Francois Bonastre, and Corinne Fredouille. 2006. Effect of Speech Transformation on Impostor Acceptance. *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing Proceedings* 1 (May 2006), 933–936. https://doi.org/10.1109/ICASSP.2006.1660175 Toulouse, France.

[24] Lawrence R. Rabiner and Ronald W. Schafer. 2010. *Theory and Applications of Digital Speech Processing* (1st ed.). Prentice Hall Press, USA.

[25] Marc Schröder, Marcela Charfuelan, Sathish Pammi, and Ingmar Steiner. 2011. Open source voice creation toolkit for the MARY TTS Platform. *Proceedings of the Annual Conference of the International Speech Communication Association - Interspeech* (August 2011), 3253–3256. https://doi.org/10.21437/Interspeech.2011-820 Florence, Italy.

[26] Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A. Saurous, Yannis Agiomvrgiannakis, and Yonghui Wu. 2018. Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (April 2018), 4779–4783. https://doi.org/10.1109/ICASSP.2018.8461368 Calgary, Canada.

[27] Stanley Smith Stevens, John Volkmann, and Edwin B. Newman. 1937. A Scale for the Measurement of the Psychological Magnitude Pitch. *Journal of the Acoustical Society of America* 8, 3 (June 1937), 185–190. https://doi.org/10.1121/1.1915893

[28] Xu Tan, Tao Qin, Frank Soong, and Tie-Yan Liu. 2021. A Survey on Neural Speech Synthesis. *arXiv:2106.15561* (July 2021). https://doi.org/10.48550/arXiv.2106.15561

[29] Alaa Tharwat. 2021. Classification Assessment Methods. *Applied Computing and Informatics* 17, 1 (December 2021), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

[30] Massimiliano Todisco, Héctor Delgado, and Nicholas Evans. 2017. Constant Q Cepstral Coefficients: A Spoofing Countermeasure for Automatic Speaker Verification. *Computer Speech & Language* 45 (September 2017), 516–535. https://doi.org/10.1016/j.csl.2017.01.001

[31] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. WaveNet: A Generative Model for Raw Audio. *Proceedings of the ISCA Workshop on Speech Synthesis Workshop* (September 2016), 125. https://doi.org/10.48550/arXiv.1609.03499 Sunnyvale, USA.

[32] Zhizheng Wu, Xiong Xiao, Eng Siong Chng, and Haizhou Li. 2013. Synthetic speech detection using temporal modulation feature. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (May 2013), 7234–7238. https://doi.org/10.1109/ICASSP.2013.6639067 Vancouver, Canada.

[33] Amit Kumar Singh Yadav, Emily R. Bartusiak, Kratika Bhagtani, and Edward J. Delp. 2023. Synthetic Speech Attribution using Self Supervised Audio Spectrogram Transformer. *Proceedings of the IS&T Media Watermarking, Security, and Forensics Conference, Electronic Imaging Symposium* (January 2023), 372–1 – 372–11. https://doi.org/10.2352/EI.2023.35.4.MWSF-372 San Francisco, USA.

[34] Amit Kumar Singh Yadav, Kratika Bhagtani, Ziyue Xiang, Paolo Bestagini, Stefano Tubaro, and Edward J. Delp. 2023. DSVAE: Interpretable Disentangled Representation for Synthetic Speech Detection. *arXiv:2304.03323* (April 2023). https://doi.org/10.48550/arXiv.2304.03323

[35] Junichi Yamagishi, Massimiliano Todisco, Md. Sahidullah, Héctor Delgado, Xin Wang, Nicolas Evans, Tomi Kinnunen, Kong Aik Lee, Ville Vestman, and Andreas Nautsch. 2019. ASVspoof 2019: The 3rd Automatic Speaker Verification Spoofing and Countermeasures Challenge database. *University of Edinburgh. The Centre for Speech Technology Research* (March 2019). https://doi.org/10.7488/ds/2555

[36] You Zhang, Fei Jiang, and Zhiyao Duan. 2021. One-class Learning Towards Synthetic Voice Spoofing Detection. *IEEE Signal Processing Letters* 28 (April 2021), 937–941. https://doi.org/10.1109/LSP.2021.3076358