Comprehensive Layer-wise Analysis of SSL Models for Audio Deepfake Detection

Anonymous ACL submission

Abstract

This paper conducts a comprehensive layer-wise analysis of self-supervised learning (SSL) models for audio deepfake detection across diverse contexts, including multilingual datasets (English, Chinese, Spanish), partial, song, and scene-based deepfake scenarios. By systematically evaluating the contributions of different transformer layers, we uncover critical insights into model behavior and performance. Our findings reveal that lower layers consistently provide the most discriminative features, while higher layers capture less relevant information. Notably, all models achieve competitive equal error rate (EER) scores even when employing a reduced number of layers. This indicates that we can reduce computational cost and increase inference speed of detecting deepfakes by utilizing only few lower layers. This work enhances our understanding of SSL models in deepfake detection, offering valuable insights applicable across varied linguistic and contextual settings.

1 Introduction

Recent advancements in speech synthesis and voice conversion have resulted in high-fidelity synthetic audio, which can convincingly mimic real human voices (Kumar et al., 2023; Huang et al., 2023; Yi et al., 2023b). As a result, distinguishing between authentic speech and sophisticated deepfake audio has become increasingly challenging (Khan et al., 2022). This poses a significant threat, particularly for Automatic Speaker Verification (ASV) systems which are widely being used for authenticating in access control, telephone banking, and forensic investigations (Anjum and Swamy, 2017; Li et al., 2024). The potential for social and economic harm is evident, highlighted by many incidents like the $243,000 scam, which exploited voice mimicry to deceive a CEO (Damiani, 2019). Such threats emphasize the urgent need for effective audio deepfake detectors to ensure the security and trustworthiness of ASV systems.

In light of these challenges, SSL has emerged as a promising approach in speech processing. They are capable of extracting rich features from vast amounts of unlabeled audio data, which have enhanced the accuracy in speaker verification and speech recognition tasks (Mohamed et al., 2022; Borgholt et al., 2022; Yang et al., 2021; Tsai et al., 2022; Shon et al., 2022). Many studies show that features derived from SSL methods consistently outperform traditional acoustic features across various tasks (Babu et al., 2021; Xie et al., 2021; Martín-Doñas and Álvarez, 2022; Wang and Yamagishi, 2021). This suggests a new direction for enhancing the resilience of speech systems against deepfake threats.

While previous studies have highlighted the potential of SSL models in deepfake detection, a thorough layer-wise analysis of SSL models remain largely unexplored. Most current research concentrates on deepfakes created exclusively in English within full-utterance audio scenarios, leaving several important questions unanswered: (1) How effectively do SSL models detect deepfakes across diverse scenarios, including multiple languages, partial, song, and scene deepfakes? (2) How do different layers of SSL models behave in these varying conditions, and is their performance consistent across different setups? (3) Which layers yield the most discriminative features for differentiating real audio from deepfakes?

In this paper, we address these gaps by undertaking a comprehensive analysis of SSL models across various settings. This includes (1) full speech utterance deepfake detection in English (En), Chinese (Zh), and Spanish (Es); (2) partial speech utterance detection in English and Chinese; and (3) detection of songs and scene-based (acoustic environment) deepfakes. By examining layer-wise contributions, we aim to provide insights into the effectiveness of SSL models in multiple languages and contexts.

Our main contributions in this paper are as fol-

low:

- We show that lower layers of the SSL models (Wav2Vec2and WavLM) are more important than upper layers in detecting audio deepfakes. This is consistent across multiple languages and setups (full vs partial deepfakes, song, scene) using different back-end classifiers.

- We demonstrate that utilizing only the lower layers of SSL models—4 layers for Small models and 10 layers for Large models—achieves performance comparable to, or even better than, the full model while maintaining generalizability across diverse datasets and setups.

This research represents a pioneering effort to conduct a comprehensive layer-wise analysis of SSL models in the context of deepfake detection, encompassing a wide range of languages and settings.

2 Background

Traditional audio deepfake detection methods use hand-crafted acoustic features like MFCC, LFCC, or CQCC (Chakroborty et al., 2008; Alegre et al., 2013; Li et al., 2023; Sahidullah et al., 2015; Todisco et al., 2017). These methods require expert knowledge and may overlook essential discriminative information (Tak et al., 2020). As a result, although it is effective on some datasets, such methods face challenges in generalizing to new or unseen spoofing attacks due to the static nature of the features. This limitation leaves systems vulnerable to emerging spoofing techniques. Therefore, there is a need for more robust and flexible anti-spoofing strategies that can automatically learn and extract relevant features from raw speech data.

With advancements in deep neural network (DNN) approaches, DNN-based methods have generally been adopted in two ways. First, they can be used as back-end models with traditional hand-crafted features as front-ends (Zhang12 et al., 2021). In light of this, light convolutional neural network (LCNN) (Lavrentyeva et al., 2017; Tomilov et al., 2021), residual neural network (ResNet) (Lai et al., 2019), and siamese convolutional neural network (Lei et al., 2020), have been successfully applied to anti-spoofing systems using MFCC, LFCC, or CQCC features. On the other hand, end-to-end DNN modeling techniques has been used to extract learnable embeddings directly from raw audio and automatically extract features relevant to anti-spoofing techniques. For instance, in (Tak et al., 2021), SincNet (Ravanelli and Bengio, 2018) is utilized to directly extract front-end features from raw audio, with fixed cut-off frequencies. Other structures as well, as explored in (Bartusiak and Delp, 2022; Shim et al., 2022; Khan et al., 2024), are applied directly to raw audio data or combined with hand-crafted features to construct deep embeddings through supervised training.

SSL models are large pre-trained frameworks that serve as the backbone for various speech tasks. These models excel in capturing high-quality representations that can be fine-tuned for specific downstream applications. In the field of speech deepfake detection, this approach has been widely adopted, leading to state-of-the-art results. Studies have shown that fine-tuning SSL models like Wav2Vec2 (Baevski et al., 2020) with a classifier, using features from the final layer, can significantly improve detection performance (Tak et al., 2022). Alternatively, fine-tuning the XLS-R (Babu et al., 2021) model and using features from the fifth layer's hidden states instead of the last layer has also proven effective (Lee et al., 2022). Research highlights that utilizing hidden state features from various layers of pre-trained models can be highly beneficial for deepfake detection (Martín-Doñas and Álvarez, 2022).

Consequently, recent studies have focused on leveraging multi-layer features from SSL models to further enhance detection performance. Methods such as multi-fusion attentive classifiers (Guo et al., 2024), attentive merging techniques (Pan et al., 2024) with 10 & 12 layers, and their improved versions (Guragain et al., 2024) have been explored. Additionally, expert fusion techniques, such as the Mixture of Experts method, have been proposed to extract and integrate relevant features for fake audio detection from multiple layers of SSL models (Wang et al., 2024).

Recent advancements in multi-layer fusion techniques underscore the importance of a deeper examination of layer-wise contributions in SSL models. While previous research has largely focused on specific SSL models, particular classifiers, and predominantly English datasets (Lee et al., 2022; Pan et al., 2024), our study offers a broader perspective. Our methodology uniquely extends beyond these limitations by comprehensively assessing deepfake detection tasks across various SSL models, languages, and scenarios. This approach facilitates a more thorough analysis of layer-wise contribu-

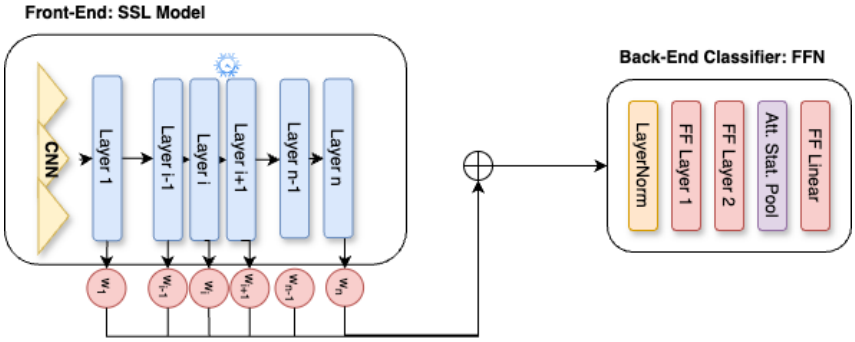


Figure 1: Layer-wise Contribution Framework. The framework consists of SSL models as front-ends to extract features and a back-end classifier. The front-end SSL models remain frozen during the experiments to evaluate the layer-wise feature contribution.

tions. Our detailed analysis reveals the most critical layers for audio deepfake detection, enabling the development of computationally efficient models that maintain effectiveness across diverse datasets.

3 Methodology

Figure 1 shows our Layer-wise Contribution Framework. This framework helps us to understand the layer-wise contribution of different transformer layers in detecting deepfakes. We use SSL models as front-ends to extract features and a back-end classifier to perform classification (real vs deepfake). We freeze the SSL model and systematically investigate the contribution of each transformer layer to the final detection performance.

Let $\mathbf{x}$ represent the input audio signal, which is passed through an SSL model. This model is composed of L transformer layers, where the output of each layer l can be denoted as $\mathbf{h}_l$. The features $\mathbf{h}_l$ extracted from each layer $l \in \{1, 2, ..., L\}$ are then weighted by a layer-wise importance factor w_l.

We follow a weighted aggregation of the extracted features, where the final representation $\mathbf{h}_{\text{final}}$ used for classification is given by:

$$\mathbf{h}_{\text{final}} = \sum_{l=1}^{L} w_l \mathbf{h}_l \tag{1}$$

The weights w_l are learnable parameters, and their values are constrained to sum to one by applying a softmax function:

$$w_l = \frac{\exp(w_l)}{\sum_{k=1}^{L} \exp(w_k)} \tag{2}$$

This ensures that the contributions from different layers are normalized. The back-end classifier receives $\mathbf{h}_{\text{final}}$ as input and generates the prediction $\hat{y}$ for the detection task. This setup allows us to investigate which layers contribute most to the detection of audio deepfakes across various settings.

3.1 Front-Ends: SSL Models

We experiment with 4 different front-end SSL models: small (base) and large versions of Wav2Vec2 and WavLM (Chen et al., 2022). These models are the most commonly used SSL models for audio deepfake detection. Wav2Vec2 and WavLM also differ in their pre-training objectives. We do not include HuBERT-based models as previous research shows that HuBERT (Hsu et al., 2021) performs similarly to WavLM in various speech tasks (Pasad et al., 2021) due to their shared pre-training objective.

Wav2Vec2.0 Wav2Vec2 is a contrastive SSL model designed to learn speech representations by predicting latent features from masked audio regions. The architecture consists of two main components: a feature extractor and a stack of transformer layers. The feature extractor is a convolutional network (CNN) with 512 channels, designed to process audio segments of approximately 25ms from 16kHz sampled audio. The CNN compresses the input with strides and kernel sizes optimized to operate every 20ms. The model comes commonly in 2 configurations: the base version with 12 transformer layers and a hidden state dimension of 768, and the large version with 24 layers and a dimension of 1024. Wav2Vec2 is trained using Contrastive Predictive Coding (CPC) loss, which distinguishes between positive and negative samples by learning contextualized audio representations. This contrastive approach enables it to effectively model the underlying structure of speech

data.

WavLM WavLM, like Wav2Vec2.0, consists of a feature extractor and transformer layers. However, it employs a different pre-training objective known as Masked Prediction Loss (MPL). MPL is a predictive task that attempts to reconstruct masked segments of audio using cross-entropy (CE) loss. This makes WavLM a predictive SSL model, optimized not only for general speech tasks but also for speaker verification, separation, and other speaker-related tasks, providing enhanced performance in scenarios involving speaker variability. WavLM is common in two versions: small with 12 transformer layers and large with 24 layers, mirroring the architecture of Wav2Vec2 but differing in its training methodology.

Model Selection and Multilingual Setup For our experiments, we select publicly available pre-trained models for both Wav2Vec2 and WavLM. Specifically, we utilize multilingual versions of WavLM (both small and large)[1] and Wav2Vec2 Large[2]. For the Wav2Vec2 small model, we opt for the English pre-trained variant due to the absence of a small multilingual pre-trained version.

3.2 Back-End Classifiers: FFN *vs* AASIST

We explore two different back-end models: (1) a simple lightweight feedforward neural network (FFN), and (2) the state-of-the-art AASIST (Jung et al., 2022) model. We experiment with these two back-end models to demonstrate the generalizability of layer-wise contributions across different classifier architectures. This ensures that our findings are not confined to a specific back-end structure.

3.2.1 FFN

We use the lightweight back-end classifier approach from (Martín-Doñas and Álvarez, 2022) for our deepfake detection task. Let $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_T]$ represent the weighted sequence of feature vectors extracted from the front-end SSL model, as defined in Equation 1. Each vector $\mathbf{h}_t \in \mathbf{R}^d$, where $d = 768$ for SSL base models & $d = 1024$ for SSL large models, is first passed through a layer normalization layer. The normalized vectors are then processed by two feedforward (FF) layers with a hidden dimension of 128, using the SeLU activation function and dropout for regularization. Next, the processed vectors $\mathbf{h}_t$ are aggregated using an attentive statistical pooling layer (Okabe et al., 2018), which computes a weighted mean and standard deviation over the time steps, yielding a fixed-length representation for the entire utterance. This attentive pooling condenses the variable-length input sequence into a single vector of size 256. Finally, the pooled representation is projected into 2 classes.

3.2.2 AASIST

AASIST is a state-of-the-art model that utilizes graph-based attention mechanisms to capture both spectral and temporal audio features. Its key components include: (1) Graph Attention Layer (GAT), which computes attention maps for spectral and temporal features using linear layers, batch normalization, and SELU activation; (2) Heterogeneous Graph Attention Layer (HtrgGAT), which processes and refines both spectral and temporal nodes; (3) a graph pooling layer, which selects the top-k nodes based on attention scores; (4) residual blocks, which apply convolutional layers and SELU activation to process features; and (5) an attention mechanism that extracts spectral and temporal features from encoded data.

4 Experimental Design

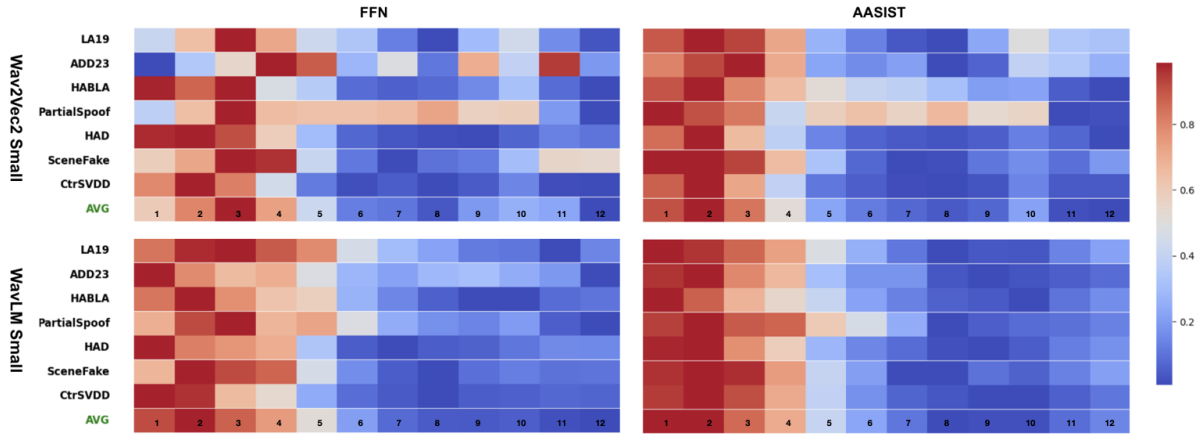
4.1 Datasets

Table 1 summarizes datasets used in this study. These datasets are very diverse including both full utterance deep fakes (**Full Fake**) and partial utterance deepfakes (**Partial Fake**) in multiple languages: English (En), Chinese (Zh), and Spanish (Es). Additionally, **Scene**-based and **Song** deepfake setups are included. All datasets leverage cutting-edge TTS and VC generation techniques for creating audio deepfakes. Additional details on the different setups can be found in Appendix A, while Appendix B provides further information on the datasets.

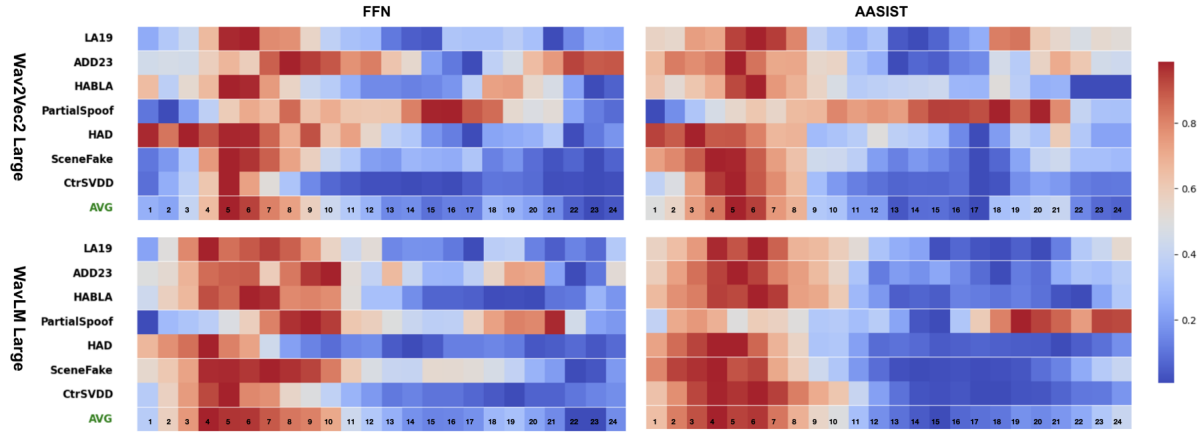
4.2 Training setup

We employ the cross-entropy (CE) loss function and the Adam optimizer, setting the learning rate to $1e^{-4}$ and a dropout rate of 0.2, with a batch size of 16 for training our models. During training, we freeze the SSL model parameters and update only the back-end classifier's parameters with the weights in Equation 1 initialized to ones. Similar to Zhang et al. (2024a), we train our models for 50 epochs with early stopping patience set to 3 epochs.

[1] unilm/blob/master/wavlm
[2] fairseq/blob/main/examples/wav2vec/xlsr



(a) Small SSL Models



(b) Large SSL Models

Figure 2: Heatmap of Normalized Layer-wise weights Across Various Datasets. AVG row representing the average weights across datasets, highlighted in green.

Dataset	Language	Split
Full Fake		
ASVspoof 2019 (19LA) (Nautsch et al., 2021)	En	Train & Eval
ASVspoof 2021 (21LA)* (Yamagishi et al., 2021)	En	Eval
ASVspoof 2021 (21DF)* (Yamagishi et al., 2021)	En	Eval
ADD23 (Track 1.2)+ (Yi et al., 2023a)	Zh	Train & Eval
HABLA (Flórez et al.)	Es	Train & Eval
Partial Fake		
PartialSpoof (Zhang et al., 2021)	En	Train & Eval
Half-Truth (HAD) (Yi et al., 2021)	Zh	Train & Eval
Song		
CtrSVDD (Zhang et al., 2024b)	Multilingual	Train & Eval
Scene		
SceneFake (Yi et al., 2024)	En	Train & Eval

Table 1: Summary of datasets used in this study. *Evaluated on models trained with LA19; + focused on Track 1.2 (Full Fake) with two eval sets, ADD23.1.2.1 and ADD23.1.2.2.

We use a single NVIDIA H100 GPU to run our experiments. We set the random seed to 42 to ensure reproducibility. We either crop or concatenate the audio data to create segments of approximately 4 seconds in duration (64,600 samples)[3]. We evaluate the model performance using the standard equal error rate (EER) metric.

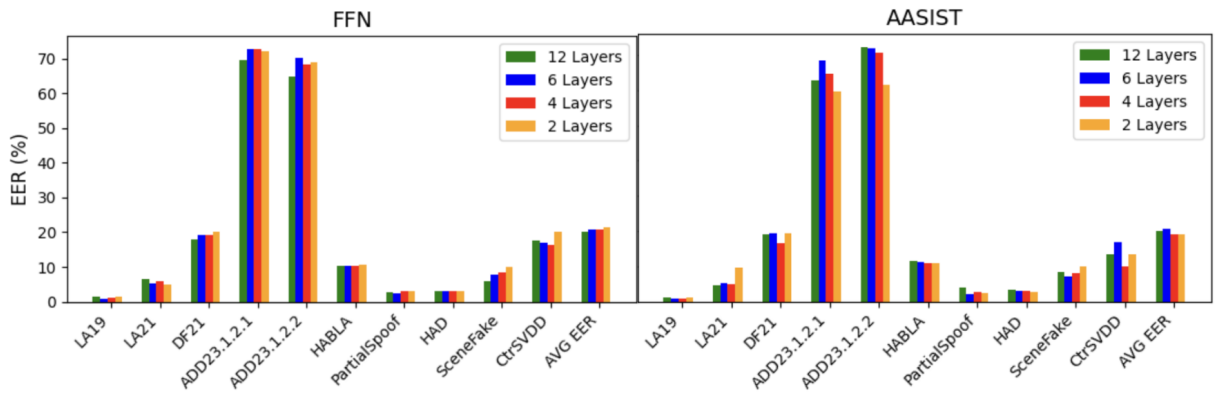
[3]For **Partial Fake**, the cropped segments are guaranteed to contain both real and deepfake portions.

5 Results and Discussion

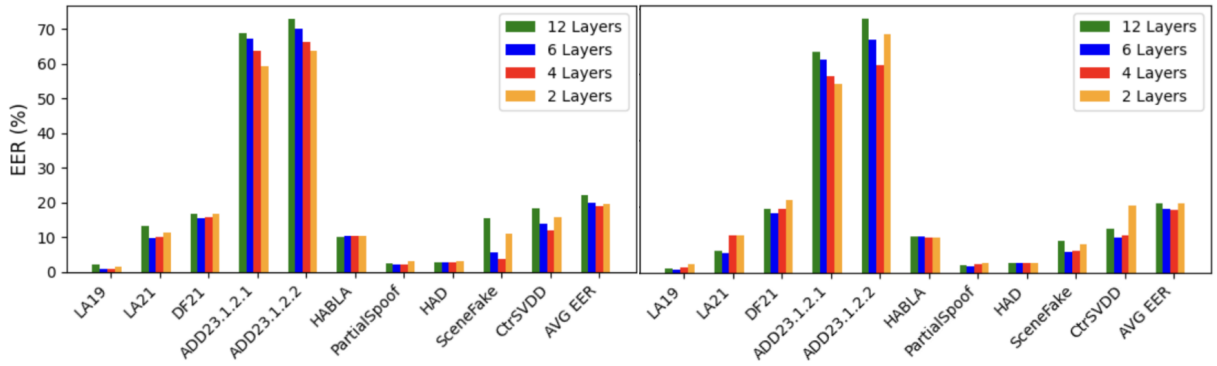
5.1 Layer-wise Contribution Analysis

After training SSL Model + Back-end classifiers with different datasets, we extract the layer-wise weights as outlined in Equation 2 to study the contribution of each layer to the task. For analysis, we first normalize these weights and then visualize them as heatmaps, as illustrated in Figure 2.

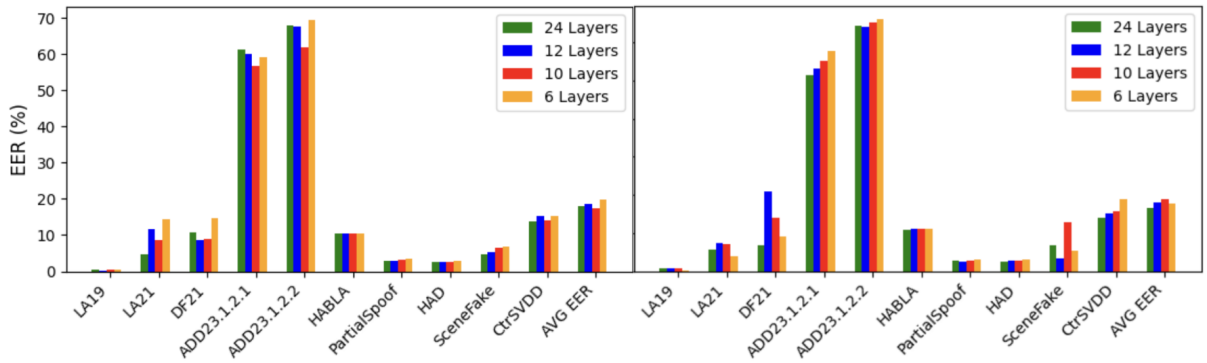
Small SSL Models Figure 2(a) presents the normalized layer-wise weights after training Wav2Vec2-Small and WavLM-Small front-end models, using FFN and AASIST back-ends across various datasets. The results consistently show that the highest contributions come from lower layers (1-4) across all datasets and setups (speech, scene, and song), indicating that these layers capture the most discriminative features for the deepfake detection task. This pattern holds regardless of the back-end classifier used, suggesting that classifier choice does not significantly impact the task reliance on lower layers. However, slight variations are observed in certain cases, such as PartialSpoof and ADD23, particularly with the pair Wav2Vec2-



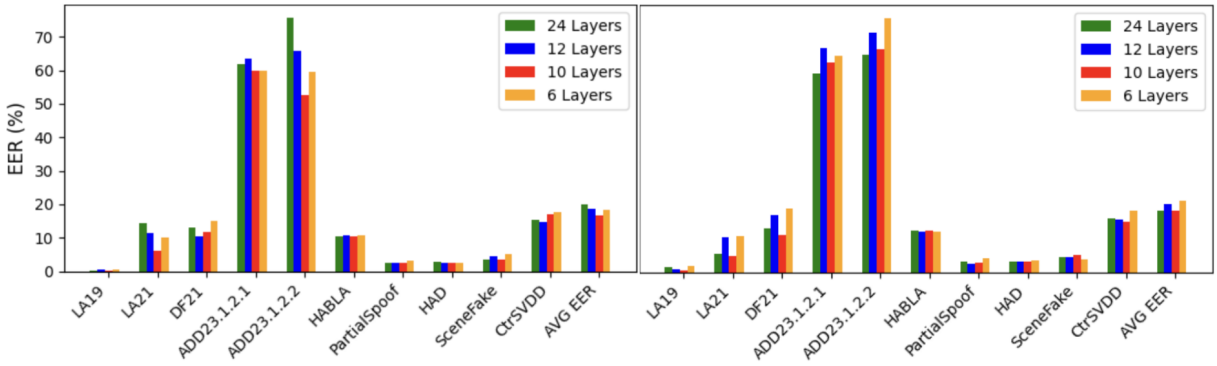
(a) Wav2Vec2 Small



(b) WavLM Small



(c) Wav2Vec2 Large



(d) WavLM Large

Figure 3: Comparison of Wav2Vec2Small (a) and WavLM Small (b) results across datasets and setups. The lower EER the better the system.

Small+FFN, where intermediate and upper layers contribute more significantly.
Looking at the average (AVG) layer-wise weights across datasets, the trend becomes even clearer, with lower layers (1-4), especially the 2nd and 3rd, consistently showing the highest contributions. This confirms the dominant role of lower layers in the deepfake performance of small SSL models.

Large SSL Models Figure 2(b) shows the normalized layer-wise weights for Wav2Vec2-Large and WavLM-Large front-end models. Unlike the concentrated lower-layer contributions seen with small SSL models, large SSL models display a lesser concentrated pattern of layer importance. For Wav2Vec2-Large, lower layers (3-10) still contribute significantly across most datasets when using both FFN and AASIST back-ends. However, a shift towards middle layers (12-21) is observed in more complex datasets like PartialSpoof and ADD23. In contrast, WavLM-Large exhibits more consistent importance in the lower layers (2-10) across datasets. When looking at the average (AVG) layer-wise weights, this trend remains stable across SSL models and back-end classifiers, with the highest contributions emerging in lower layers (1-10), particularly around the 4th, 5th, and 6th layers.

Particularly, The scarcity of layer-wise weights in Wav2Vec2-Large, compared to WavLM-Large, can be attributed to their distinct training objectives, which influence how they capture local and semantic features across the layers. Wav2Vec2-Large shows a broader distribution of relevant features over more layers compared to WavLM-Large, which tends to encapsulate local features primarily in the lower layers. This distinction is a direct consequence of their respective pre-training tasks where Wav2Vec2 models employ a CPC loss that encourages to capture acoustic features across a wider range of layers. In contrast, WavLM-Large is designed to predict discrete units learned in an intermediate layer, leading to a concentration of local feature extraction in the lower layers and more semantic information in the upper layers (Pasad et al., 2023).

Focus on Local Features Across Models Across all models, there is a consistent reliance on the lower layers, particularly layers 1-4 in Small SSL models and layers 1-10 in Large SSL models. This trend suggests that the models focus on local features, which aligns with how audio deepfake detectors identify artifacts left by VC and TTS algorithms. These artifacts often reside in specific frequency sub-bands or temporal segments, highlighting the significance of local feature extraction (Sriskandaraja et al., 2016; Tak et al., 2020).
Particularly, in our investigation of the PartialSpoof data, we observed significant disfluency in certain speech segments, indicating that the model tends to focus more on the transitions between fake and real segments as shown in (Liu et al., 2024). This may explain the model's increased reliance on intermediate and upper SSL layers, which are more effective at capturing disfluent speech patterns, as demonstrated in (Shih et al., 2024).

5.2 Performance of Reduced-Layer SSL Models

This section investigates the capabilities of SSL models when limited to a decreased number of transformer layers. We evaluate Wav2Vec2-Small and WavLM-Small using 2, 4, and 6 layers, as well as Wav2Vec2-Large and WavLM-Large using 6, 10, and 12 layers. These configurations are guided by our earlier layer-wise analysis. The reduced-layer models are trained under the same conditions as the full models described in Section 4.2, and their performance is assessed across the same datasets and setups. We introduce a notation where "SSLModel-X" represents a model with X layers (e.g., Wav2Vec2-Small-4 denotes Wav2Vec2-Small with 4 layers).

Performance Comparison Across Deepfake Scenarios Our analysis reveals the effectiveness of reduced-layer SSL models across various deepfake detection scenarios. Detailed results are provided in Table 3 in Appendix C, while Figure 3 illustrates the performance across different reduced-SSL configurations. Across all datasets and settings, reduced-layer SSL models demonstrate comparable or superior performance to their full-layer counterparts. Figure 3 demonstrates that neither full SSL models nor reduced-SSL counterparts consistently outperform one another. Performance varies across datasets and back-end classifiers, with full SSL models sometimes yielding the best results and reduced-SSL models excelling in other cases. However, the differences between the two configurations are generally marginal. As highlighted in Table 3, for instance Wav2Vec2-Small-6+FFN achieves a lower EER of **0.84%** on LA19 dataset compared to Wav2Vec2-Small

	Wav2Vec2-Small			WavLM-Small			AVG_{out}
Back-End/#L	**FFN**	**AASIST**	$\textbf{AVG}_{in}$	**FFN**	**AASIST**	$\textbf{AVG}_{in}$	
12	**20.01**	20.02	20.01	22.26	20.97	21.61	20.81
6	20.86	20.59	20.72	19.80	19.22	19.51	20.12
4	20.80	**19.16**	**19.98**	**18.82**	**18.81**	**18.81**	**19.40**
2	21.45	19.02	20.23	19.53	20.95	20.24	20.24
	Wav2Vec2-Large			**WavLM-Large**			AVG_{out}
Back-End/#L	**FFN**	**AASIST**	$\textbf{AVG}_{in}$	**FFN**	**AASIST**	$\textbf{AVG}_{in}$	
24	17.92	**16.53**	**17.23**	19.99	**16.71**	18.35	17.79
12	18.48	18.11	18.30	18.64	18.62	18.63	18.46
10	**17.30**	18.74	18.02	**16.67**	16.77	**16.72**	**17.37**
6	19.69	17.83	18.76	18.43	19.47	18.95	18.86

Table 2: Reported average EER across datasets using different Wav2Vec2 and WavLM SSL models with different back-end layers. AVG_{in} represents the average EER across datasets for each SSL model, calculated by first averaging the EER for different back-ends (FFN and AASIST) within the same reduced-layer model. AVG_{out} indicates the overall average of AVG_{in}, calculated separately for SSL Small and SSL Large models. #L indicates the number of transformer layers.

model's **1.28%**. Similarly, on LA21 dataset, Wav2Vec2-Small-2 results in an EER of **5.03%**, outperforming the full model's **6.48%**. WavLM-Small models also exhibit marginal performance differences when paired with AASIST. WavLM-Small-6 configuration yields an EER of **9.66%** on LA21, compared to **13.23%** for the full model, indicating the effectiveness of reduced layers. On ADD23.1.2.2 dataset, the WavLM-Small-4+ASSIST model scores **62.65%**, outperforming the full model's **76.90%**. For Wav2Vec-Large-6+AASIST, it achieves an EER of **3.38%** on the SceneFake dataset, significantly lower than the full model's **6.86%**. Additionally, Wav2Vec-Large-10 shows marginal improvements with an EER of **13.65%** on CtrSVDD dataset compared to **14.73%** for the full SSL model.

5.3 Optimal Layer Configuration Across Datasets

Our findings indicate that using 4 layers for Wav2Vec-Small and WavLM-Small, and 10 layers for Wav2Vec2-Large and WavLM-Large consistently yields promising average results across datasets. As shown in Table 2, Wav2Vec2-Small-4 and WavLM-Small-4 achieve an $\textbf{AVG}_{out}$ of **19.40%**, outperforming full SSL and other reduced-SSL settings across datasets. Similarly, the 10-layer configuration for Large models also shows optimal performance with an $\textbf{AVG}_{out}$ of **17.37%**, demonstrating its robustness across multiple datasets. However, there are specific cases where the 10-layer configuration is not the optimal choice for Wav2Vec2-Large. For instance, when analyzing performance on full Wav2Vec2-Large, we observe a slightly better $\textbf{AVG}_{in}$ of **17.23%**, which is marginally lower than the Wav2Vec2-Large-10 configuration. Despite this, the difference remains small (**0.79%**), indicating that the 10-layer configuration still offers a competitive trade-off between performance and computational efficiency.

6 Small *vs* Large SSL Models

When comparing Small and Large SSL models across various datasets, no significant performance difference is observed in detecting less challenging deepfake manipulations. However, for more complex tasks, such as those in the DF21, ADD23.1.2.1, and ADD23.1.2.2 datasets, larger architectures become essential for capturing subtle deepfake artifacts. As shown in Table 3, a **7.87%** difference in EER is noted when comparing Wav2Vec-Small's best performance on the ADD23.1.2.1 with Wav2Vec-Large. Additionally, a **10.03%** difference in EER is observed between WavLM-Small and WavLM-Large on ADD23.1.2.2. In contrast, across other datasets such as HABLA, PartialSpoof, HAD, and SceneFake, the performance remains relatively consistent between Small and Large SSL models.

7 Conclusion

This study provides a comprehensive layer-wise analysis of SSL models for audio deepfake detection, covering multilingual datasets and a range of deepfake types, including Full Fake, Partial Fake, and Song- and Scene-based scenarios. Our results show that using only first 4 layers for Small models and 10 layers for Large models consistently delivers optimal performance across datasets, reducing parameters by over 66% in Small models and 58% in Large models. This significantly lowers computational costs while maintaining competitive EER scores.

8 Limitations

Our study provides a comprehensive layer-wise analysis of SSL models for deepfake detection across languages and contexts, but it has some limitations. We did not investigate the latest Audio Language Models and Speech tokenizers used for audio deepfake generation (Xie et al., 2024), which might lead to different artifacts and model behavior. Additionally, we focused solely on speech-based SSL models, excluding acoustic SSL models. Future work could explore these models, especially for tasks involving non-speech artifacts, like those in the SceneFake dataset. We also plan to investigate alternative fusion strategies, such as attentional fusion, beyond a simple weighted-sum layer-wise combination.

References

Federico Alegre, Asmaa Amehraye, and Nicholas Evans. 2013. A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns. In *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–8. IEEE.

Z Khamar Anjum and R Kumara Swamy. 2017. Spoofing and countermeasures for speaker verification: A review. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 467–471. IEEE.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick Von Platen, Yatharth Saraf, Juan Pino, et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Emily R Bartusiak and Edward J Delp. 2022. Transformer-based speech synthesizer attribution in an open set scenario. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 329–336. IEEE.

Lasse Borgholt, Jakob Drachmann Havtorn, Joakim Edin, Lars Maaløe, and Christian Igel. 2022. A brief overview of unsupervised neural speech representation learning. *arXiv preprint arXiv:2203.01829*.

Sandipan Chakroborty, Anindya Roy, and Goutam Saha. 2008. Improved closed set text-independent speaker identification by combining mfcc with evidence from flipped filter banks. *International Journal of Electronics and Communication Engineering*, 2(11):2554–2561.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Jesse Damiani. 2019. A voice deepfake was used to scam a ceo out of 243,000. *Forbes Magazine*.

Pablo Andrés Tamayo Flórez, Rubén Manrique, and Bernardo Pereira Nunes. Habla: A dataset of latin american spanish accents for voice anti-spoofing.

Yinlin Guo, Haofan Huang, Xi Chen, He Zhao, and Yuehai Wang. 2024. Audio deepfake detection with self-supervised wavlm and multi-fusion attentive classifier. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12702–12706. IEEE.

Anmol Guragain, Tianchi Liu, Zihan Pan, Hardik B Sailor, and Qiongqiong Wang. 2024. Speech foundation model ensembles for the controlled singing voice deepfake detection (ctrsvdd) challenge 2024. *arXiv preprint arXiv:2409.02302*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.

Wen-Chin Huang, Lester Phillip Violeta, Songxiang Liu, Jiatong Shi, and Tomoki Toda. 2023. The singing voice conversion challenge 2023. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Jee-weon Jung, Hee-Soo Heo, Hemlata Tak, Hye-jin Shim, Joon Son Chung, Bong-Jin Lee, Ha-Jin Yu, and Nicholas Evans. 2022. Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks. In *ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 6367–6371. IEEE.

Awais Khan, Khalid Mahmood Malik, and Shah Nawaz. 2024. Frame-to-utterance convergence: A spectra-temporal approach for unified spoofing detection. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10761–10765. IEEE.

Awais Khan, Khalid Mahmood Malik, James Ryan, and Mikul Saravanan. 2022. Voice spoofing countermeasures: Taxonomy, state-of-the-art, experimental analysis of generalizability, open challenges, and the way forward. *arXiv preprint arXiv:2210.00417*.

Yogesh Kumar, Apeksha Koul, and Chamkaur Singh. 2023. A deep learning approaches in text-to-speech system: a systematic review and recent research perspective. *Multimedia Tools and Applications*, 82(10):15171–15197.

Cheng-I Lai, Nanxin Chen, Jesús Villalba, and Najim Dehak. 2019. Assert: Anti-spoofing with squeeze-excitation and residual networks. *arXiv preprint arXiv:1904.01120*.

Galina Lavrentyeva, Sergey Novoselov, Egor Malykh, Alexander Kozlov, Oleg Kudashev, and Vadim Shchemelinin. 2017. Audio replay attack detection with deep learning frameworks. In *Interspeech*, pages 82–86.

Jin Woo Lee, Eungbeom Kim, Junghyun Koo, and Kyogu Lee. 2022. Representation selective self-distillation and wav2vec 2.0 feature exploration for spoof-aware speaker verification. *arXiv preprint arXiv:2204.02639*.

Zhenchun Lei, Yingen Yang, Changhong Liu, and Jihua Ye. 2020. Siamese convolutional neural network using gaussian probability feature for spoofing speech detection. In *Interspeech*, pages 1116–1120.

Menglu Li, Yasaman Ahmadiadli, and Xiao-Ping Zhang. 2024. Audio anti-spoofing detection: A survey. *arXiv preprint arXiv:2404.13914*.

Wei Li, Jichen Yang, and Pei Lin. 2023. Investigation of the influence of blocks on the linear spectrum for synthetic speech detection. *Electronics Letters*, 59(9):e12797.

Tianchi Liu, Lin Zhang, Rohan Kumar Das, Yi Ma, Ruijie Tao, and Haizhou Li. 2024. How do neural spoofing countermeasures detect partially spoofed audio? *arXiv preprint arXiv:2406.02483*.

Juan M Martín-Doñas and Aitor Álvarez. 2022. The vicomtech audio deepfake detection system based on wav2vec2 for the 2022 add challenge. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9241–9245. IEEE.

Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, et al. 2022. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1179–1210.

Andreas Nautsch, Xin Wang, Nicholas Evans, Tomi H Kinnunen, Ville Vestman, Massimiliano Todisco, Héctor Delgado, Md Sahidullah, Junichi Yamagishi, and Kong Aik Lee. 2021. Asvspoof 2019: spoofing countermeasures for the detection of synthesized, converted and replayed speech. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(2):252–265.

Koji Okabe, Takafumi Koshinaka, and Koichi Shinoda. 2018. Attentive statistics pooling for deep speaker embedding. *arXiv preprint arXiv:1803.10963*.

Zihan Pan, Tianchi Liu, Hardik B Sailor, and Qiongqiong Wang. 2024. Attentive merging of hidden embeddings from pre-trained speech model for anti-spoofing detection. *arXiv preprint arXiv:2406.10283*.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. 2021. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921. IEEE.

Ankita Pasad, Bowen Shi, and Karen Livescu. 2023. Comparative layer-wise analysis of self-supervised speech models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Mirco Ravanelli and Yoshua Bengio. 2018. Speaker recognition from raw waveform with sincnet. In *2018 IEEE spoken language technology workshop (SLT)*, pages 1021–1028. IEEE.

Md. Sahidullah, Tomi Kinnunen, and Cemal Hanilçi. 2015. A comparison of features for synthetic speech detection. In *Proc. Interspeech 2015*, pages 2087–2091.

Yao Shi, Hui Bu, Xin Xu, Shaoji Zhang, and Ming Li. 2020. Aishell-3: A multi-speaker mandarin tts corpus and the baselines. *arXiv preprint arXiv:2010.11567*.

Yi-Jen Shih, Zoi Gkalitsiou, Alexandros G Dimakis, and David Harwath. 2024. Self-supervised speech models for word-level stuttered speech detection. *arXiv preprint arXiv:2409.10704*.

Hye-jin Shim, Jungwoo Heo, Jae-Han Park, Ga-Hui Lee, and Ha-Jin Yu. 2022. Graph attentive feature aggregation for text-independent speaker verification. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7972–7976. IEEE.

Suwon Shon, Ankita Pasad, Felix Wu, Pablo Brusco, Yoav Artzi, Karen Livescu, and Kyu J Han. 2022. Slue: New benchmark tasks for spoken language understanding evaluation on natural speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7927–7931. IEEE.

Kaavya Sriskandaraja, Vidhyasaharan Sethu, Phu Ngoc Le, and Eliathamby Ambikairajah. 2016. Investigation of sub-band discriminative information between spoofed and genuine speech. In *Interspeech*, pages 1710–1714.

Hemlata Tak, Jose Patino, Andreas Nautsch, Nicholas Evans, and Massimiliano Todisco. 2020. An explainability study of the constant q cepstral coefficient spoofing countermeasure for automatic speaker verification. *arXiv preprint arXiv:2004.06422*.

Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher. 2021. End-to-end anti-spoofing with rawnet2. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6369–6373. IEEE.

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 preprint arXiv:2202.12233*.

Massimiliano Todisco, Héctor Delgado, and Nicholas Evans. 2017. Constant q cepstral coefficients: A spoofing countermeasure for automatic speaker verification. *Computer Speech & Language*, 45:516–535.

Anton Tomilov, Aleksei Svishchev, Marina Volkova, Artem Chirkovskiy, Alexander Kondratev, and Galina Lavrentyeva. 2021. Stc antispoofing systems for the asvspoof2021 challenge. In *Proc. ASVspoof 2021 Workshop*, pages 61–67.

Hsiang-Sheng Tsai, Heng-Jui Chang, Wen-Chin Huang, Zili Huang, Kushal Lakhotia, Shu-wen Yang, Shuyan Dong, Andy T Liu, Cheng-I Jeff Lai, Jiatong Shi, et al. 2022. Superb-sg: Enhanced speech processing universal performance benchmark for semantic and generative capabilities. *arXiv preprint arXiv:2203.06849*.

Xin Wang and Junichi Yamagishi. 2021. Investigating self-supervised front ends for speech spoofing countermeasures. *arXiv preprint arXiv:2111.07725*.

Zhiyong Wang, Ruibo Fu, Zhengqi Wen, Jianhua Tao, Xiaopeng Wang, Yuankun Xie, Xin Qi, Shuchen Shi, Yi Lu, Yukun Liu, et al. 2024. Mixture of experts fusion for fake audio detection using frozen wav2vec 2.0. *arXiv preprint arXiv:2409.11909*.

Yang Xie, Zhenchuan Zhang, and Yingchun Yang. 2021. Siamese network with wav2vec feature for spoofing speech detection. In *Interspeech*, pages 4269–4273.

Yuankun Xie, Yi Lu, Ruibo Fu, Zhengqi Wen, Zhiyong Wang, Jianhua Tao, Xin Qi, Xiaopeng Wang, Yukun Liu, Haonan Cheng, Long Ye, and Yi Sun. 2024. The codecfake dataset and countermeasures for the universally detection of deepfake audio. *Preprint*, arXiv:2405.04880.

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. In *ASVspoof 2021 Workshop-Automatic Speaker Verification and Spoofing Coutermeasures Challenge*.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.

Jiangyan Yi, Ye Bai, Jianhua Tao, Haoxin Ma, Zhengkun Tian, Chenglong Wang, Tao Wang, and Ruibo Fu. 2021. Half-truth: A partially fake audio detection dataset. *arXiv preprint arXiv:2104.03617*.

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)*, pages 9216–9220. IEEE.

Jiangyan Yi, Jianhua Tao, Ruibo Fu, Xinrui Yan, Chenglong Wang, Tao Wang, Chu Yuan Zhang, Xiaohui Zhang, Yan Zhao, Yong Ren, et al. 2023a. Add 2023: the second audio deepfake detection challenge. *arXiv preprint arXiv:2305.13774*.

Jiangyan Yi, Chenglong Wang, Jianhua Tao, Chu Yuan Zhang, Cunhang Fan, Zhengkun Tian, Haoxin Ma, and Ruibo Fu. 2024. Scenefake: An initial dataset and benchmarks for scene fake audio detection. *Pattern Recognition*, 152:110468.

Jiangyan Yi, Chenglong Wang, Jianhua Tao, Xiaohui Zhang, Chu Yuan Zhang, and Yan Zhao. 2023b. Audio deepfake detection: A survey. *arXiv preprint arXiv:2308.14970*.

Lin Zhang, Xin Wang, Erica Cooper, Junichi Yamagishi, Jose Patino, and Nicholas Evans. 2021. An initial investigation for detecting partially spoofed audio. *arXiv preprint arXiv:2104.02518*.

Qishan Zhang, Shuangbing Wen, and Tao Hu. 2024a. Audio deepfake detection with self-supervised xls-r and sls classifier. In *ACM Multimedia 2024*.

You Zhang, Yongyi Zang, Jiatong Shi, Ryuichi Yamamoto, Jionghao Han, Yuxun Tang, Tomoki Toda, and Zhiyao Duan. 2024b. Svdd challenge 2024: A singing voice deepfake detection challenge evaluation plan. *arXiv preprint arXiv:2405.05244*.

Yuxiang Zhang12, Wenchao Wang12, and Pengyuan Zhang12. 2021. The effect of silence and dual-band fusion in anti-spoofing system. In *Proc. Interspeech*.

A Setup Description

In this study, various types of audio deepfakes are utilized to evaluate the performance of SSL. These deepfakes are categorized as follows:

Full Fake: The entire utterance is generated using voice conversion (VC) or text-to-speech (TTS) technologies. No part of the original speech remains intact.

Partial Fake: Only specific segments or words from the original utterance are replaced with VC or TTS-generated content. The remainder of the utterance is real.

Song: These setups involve utterances accompanied by background music, with corresponding speech content either real or deepfaked.

Scene: In these cases, the speech itself is real, but the background environment or context has been altered (e.g., a different ambiance or noise is introduced to simulate a new scene).

B Datasets

In this section, we provide information about different datasets used in the study:

B.1 ASVSpoof19 (LA19):

Our work focuses on the LA subset of the ASVSpoof19 database, which contains both bonafide (real) and spoofed speech (deepfake) data. The spoofed data is generated using 17 different TTS and Voice VC systems. While the TTS and VC systems were trained on data from the VCTK[4] database, there is no overlap with the data in the 2019 database. Of the 17 systems, 6 are categorized as known attacks, and the remaining 11 are classified as unknown. The training and development sets consist exclusively of known attacks, while the evaluation set includes 2 known attacks and 11 unknown spoofing attacks. Among the 6 known attacks, 2 are VC systems and 4 are TTS systems.

B.2 ASVSpoof21 (LA21 & DF21):

The ASVspoof21 evaluation dataset contains bonafide and spoofed speech transmitted over various telephony systems, including VoIP and PSTN. While no additional noise is introduced, transmission introduces variability and artifacts from both spoofing and encoding. Unlike previous LA tasks, LA21 includes acoustic propagation, with all data featuring reverberation and noise. The evaluation set includes recordings from the same 48 speakers (21 male, 27 female) as ASVspoof 2019. The DF21 task simulates an attack where the attacker has access to a victim's voice data, such as audio from social media. The dataset is sourced from the VCTK corpus and additional undisclosed datasets.

B.3 ADD23 (Track1.2):

ADD23 Track 1.2 focuses on detecting fake utterances, particularly those generated in Track 1.1 (from the ADD23 challenge). The training and development sets are the same as those used in Track 3.2 of ADD 2022(Yi et al., 2022), comprising both real and fake utterances based on the AISHELL-3 dataset (Shi et al., 2020). The evaluation phase includes two distinct datasets, referred to as ADD23.1.2.1 and ADD23.1.2.2.

B.4 HABLA:

The HABLA dataset is a Spanish-language anti-spoofing corpus that includes accents from Argentina, Colombia, Peru, Venezuela, and Chile. It contains more than 22,000 genuine speech samples from both male and female speakers across these five countries, alongside 58,000 spoofed samples generated using six different speech synthesis methods. We use the train/dev/eval split provided by the dataset authors.

B.5 PartialSpoof:

The PartialSpoof database is derived from the ASVspoof 2019 LA dataset, which includes 17 types of spoofed data generated by advanced speech synthesizers, voice converters, and hybrid systems. The same bonafide data from the ASVspoof19 LA corpus is used to create partially spoofed audio by following these steps: first, voice activity detection (VAD) algorithms are applied, and based on the detected boundaries, a randomly selected segment of a bonafide utterance are replaced with a spoofed segment.

B.6 Half-Truth (HAD):

The HAD (Half-Truth Audio Detection) dataset is focused on detecting partially fake audio, where only a small portion of an utterance—such as individual words—is altered using advanced TTS systems. It is based on the AISHELL-3 speech corpus, which includes multi-speaker recordings designed for TTS model training. The dataset provides the challenging task of identifying subtle manipulations in audio where most of the content remains genuine.

B.7 CtrSVDD:

CtrSVDD is a diverse dataset focused on detecting deepfake singing vocals. It consists of 47.64 hours of bona fide singing and 260.34 hours of deepfake vocals generated using state-of-the-art synthesis techniques. Deepfake songs span 14 different methods, including 7 speech synthesis and 7 voice conversion (VC) techniques. The dataset includes 164 singer identities and provides a total of 188,486 deepfake song clips and 32,312 genuine song clips, with an average clip length of 5.02 seconds. CtrSVDD offers a rich resource for studying deepfake detection in singing, making it a valuable tool for advancing research in this area.

B.8 SceneFake:

SceneFake is a dataset developed for scene fake audio detection, focusing on manipulated audio samples generated by altering the acoustic scene of real

[4] https://datashare.ed.ac.uk/handle/10283/2651

utterances using advanced speech enhancement technologies. Unlike traditional datasets that modify timbre, prosody, or linguistic content, SceneFake specifically addresses the challenge of detecting audio where the background environment has been tampered with. The dataset includes a variety of real utterances paired with their manipulated counterparts, and initial experiments indicate that existing models trained on datasets like ASVSpoof19 struggle to reliably detect scene fake utterances.

C Detailed Results

			Eval Datasets										
Front-End SSL	Back-End Model	#Transformer Layers	LA19	LA21	DF21	ADD23.1.2.1	ADD23.1.2.2	HABLA	PartialSpoof	HAD	SceneFake	CtrSVDD	AVG EER
Wav2Vec-Small	FFN	12	1.28	6.48	**17.94**	**69.76**	**65.01**	**10.29**	2.67	3.02	**5.93**	17.72	**20.01**
		6	**0.84**	5.23	19.18	72.90	70.18	10.36	**2.50**	**2.91**	7.69	16.85	20.86
		4	0.98	5.82	19.34	72.74	68.35	10.38	2.86	2.97	8.41	**16.24**	20.80
		2	1.37	**5.03**	20.09	72.24	69.19	10.51	2.96	3.00	9.94	20.20	21.45
	AASIST	12	1.18	**4.71**	19.02	62.78	72.23	11.55	3.79	3.38	8.22	13.39	20.02
		6	**0.81**	5.16	19.49	68.51	72.01	11.11	**2.18**	2.94	**7.03**	16.70	20.59
		4	0.93	4.76	**16.6**	64.47	70.72	**10.85**	2.61	2.85	7.89	**10.00**	19.16
		2	1.11	9.52	19.38	**59.57**	**61.45**	10.87	2.51	**2.73**	9.80	13.31	**19.02**
WavLM-Small	FFN	12	2.07	13.23	16.62	68.87	73.08	**10.13**	2.35	2.77	15.29	18.22	22.26
		6	0.87	**9.66**	**15.38**	67.23	70.19	10.36	**2.17**	**2.67**	5.58	13.95	19.80
		4	**0.78**	10.02	15.91	63.89	66.47	10.50	2.20	2.77	**3.78**	**11.88**	**18.82**
		2	1.41	11.28	16.56	**59.22**	**63.91**	10.38	3.00	2.93	10.93	15.77	19.53
	AASIST	12	1.29	6.63	19.21	66.86	76.90	11.04	2.24	**2.71**	9.50	13.37	20.97
		6	**0.96**	**6.00**	**17.93**	64.56	70.37	10.77	**1.95**	2.87	**6.16**	**10.70**	19.22
		4	1.58	11.29	19.38	**59.39**	**62.65**	**10.58**	2.57	2.85	6.65	11.21	**18.81**
		2	2.51	11.07	21.83	56.99	72.31	10.66	2.85	2.75	8.51	20.11	20.95
Wav2Vec-Large	FFN	24	0.51	**4.56**	10.71	61.44	68.05	**10.40**	**2.80**	**2.50**	**4.59**	**13.70**	17.92
		12	0.29	11.71	**8.6**	60.09	67.68	10.50	2.91	2.56	5.16	15.37	18.48
		10	0.33	8.59	8.86	**56.65**	**61.95**	10.44	3.15	2.60	6.41	14.08	**17.30**
		6	**0.32**	14.35	14.66	59.33	69.53	10.44	3.55	2.79	6.74	15.25	19.69
	AASIST	24	0.75	5.48	**6.65**	**51.35**	64.28	**10.81**	2.63	**2.46**	6.86	**14.07**	**16.53**
		12	0.74	7.46	20.86	53.26	**64.09**	11.02	**2.49**	2.88	**3.38**	14.99	18.11
		10	0.78	7.20	13.94	55.26	65.14	10.98	2.72	2.76	12.86	15.80	18.74
		6	**0.21**	**3.98**	9.04	57.67	66.17	10.99	3.12	3.11	5.22	18.84	17.83
WavLM-Large	FFN	24	0.30	14.49	13.00	62.01	75.77	10.50	**2.37**	2.73	**3.33**	15.48	19.99
		12	0.61	11.42	**10.28**	63.50	65.94	10.58	2.62	2.53	4.38	**14.62**	18.64
		10	**0.28**	**6.19**	11.58	**59.87**	**52.62**	**10.51**	2.46	2.59	3.50	17.14	**16.67**
		6	0.39	10.14	15.15	59.99	59.70	10.74	3.16	**2.41**	4.97	17.74	18.43
	AASIST	**24**	**1.23**	5.03	**11.91**	54.01	59.15	**11.23**	**2.86**	2.79	**4.21**	**14.73**	**16.71**
		12	0.87	9.37	15.53	60.79	65.21	**10.87**	**2.24**	2.91	4.04	14.40	18.62
		10	**0.38**	4.51	**10.21**	57.12	60.67	11.15	2.66	2.84	4.56	**13.65**	16.77
		6	1.57	9.81	17.26	58.80	69.05	11.11	3.87	3.02	**3.52**	16.76	19.47

Table 3: EER results across various datasets, using full and partial transformer layers of SSL Models. * denotes the average EER across all datasets. while # indicates the number of layers used from the front-end SSL models. Best EER results are bolded, Best AVG EER is highlighted in red