FAU-Erlangen System Description for the 1st VoicePrivacy Attacker Challenge

Ünal Ege Gaznepoglu*, Anna Leschanowsky†, Ahmad Aloradi‡, Prachi Singh†
Daniel Tenbrinck‡, Emanuël Habets*, Nils Peters§
*International Audio Laboratories, Friedrich-Alexander-University Erlangen-Nürnberg, Germany
†Fraunhofer IIS, Erlangen, Germany
‡Department of Data Science, Friedrich-Alexander-University Erlangen-Nürnberg, Germany
§Department of Electronic & Electrical Engineering, Trinity College Dublin, Ireland
ege.gaznepoglu [at] audiolabs-erlangen.de

***Abstract*—We present two novel attacks against speaker anonymization systems. The first exploits the linguistic content, based on a language model. The second one targets the so-called AdMixture anonymization systems. We observe that, relying only on the textual content, our first method manages to de-anonymize certain speakers in the Voice Privacy Challenge (VPC) datasets, hence, raising questions on the effectiveness of the evaluation methodology, in particular, the datasets. The second method provides a simple fix to the baseline by exploiting the number of possible anonymization systems.**

***Index Terms*—speaker anonymization, automated speaker verification systems, language models**

I. Introduction

Recent technological advances in speech processing and the accompanying risks have led to the development of speaker anonymization techniques. The aim is to develop speech processing systems that hide the speaker's identity while retaining useful information for certain downstream tasks, such as automated speech recognition (ASR) and emotion recognition [1], [2]. Through Voice Privacy Challenges (VPCs), numerous systems emerged despite the ever-increasing requirements. For the first time, a VoicePrivacy Attacker Challenge is held, where the task is to develop systems compromising privacy to raise the bar for the speaker anonymization systems further [3].

The attack models, visualized in Fig. 1, define how much information is available to the attacker. The *unprotected case* is meant as a reference, measuring how identifiable the enrollment speakers are without any anonymization attempt. The *ignorant attack model* has access to the anonymized enrollment utterances, as well as original trial utterances. The *lazy-informed attack model* assumes that the attacker can invoke the employed anonymization system, and uses it to anonymize the trial utterances. The aforementioned attack models use a pre-trained ASV_{eval} on unprocessed speech. The *semi-informed attack model* further exploits the knowledge of the anonymization system by using anonymized speech to train a custom ASV_{eval}^{anon} and is currently considered the strongest attack model known to date [2].

The literature on the analysis of attacker scores is rather limited. In [4], Williams et al. explored speaker-level distributions of ASV scores obtained through ignorant and lazy-informed attack models. Other than these, there is also a study on pathological speech [5] using the ignorant attack model. They found the standard deviation of equal error rates (EERs) attained across speakers to be very low, e.g., $32.26\% \pm 0.31$. To the best of our knowledge, there is no study on semi-informed attacker scores on the speaker level.

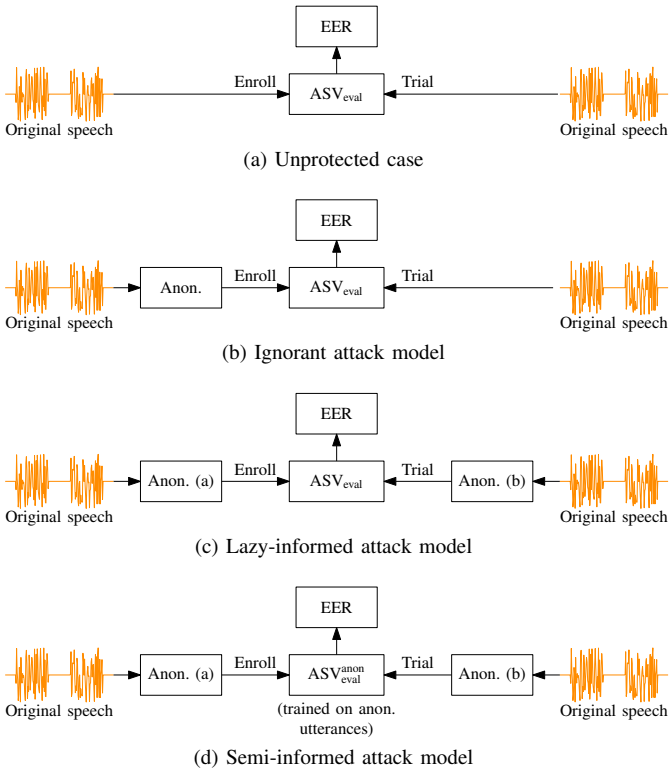


Fig. 1: VPC attack models [6].

II. Motivation

A. Speaker-level breakdown of the VPC2024 results

We started by analyzing the status quo by visualizing and inspecting the score distributions across several variables (see

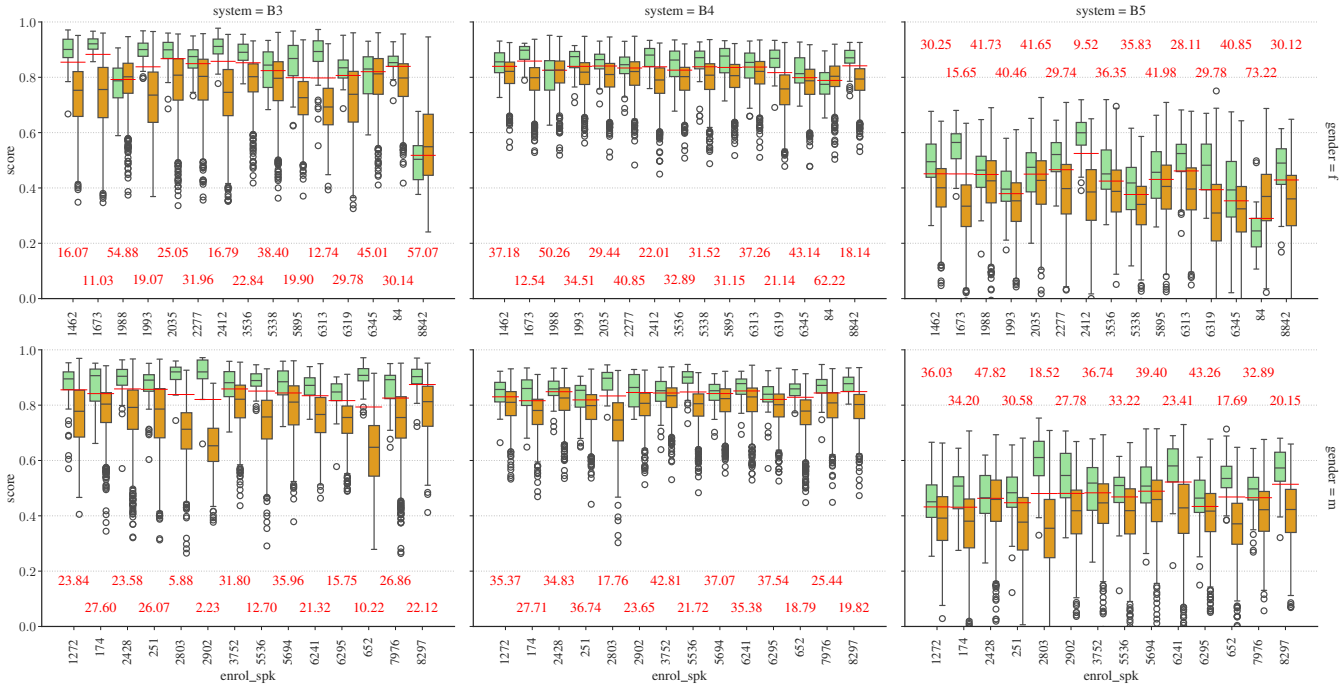


Fig. 2: Speaker-level breakdown of the semi-informed attack scores on the `libri-dev` dataset. The top row contains the female speakers, whereas the bottom row contains the male speakers. Each column displays the scores obtained from ASV_{eval}^{anon} corresponding to the indicated system (left: B3, middle: B4, right: B5). The bar plots denote the cosine distance score distributions, where light green bars indicate the positive pairs and orange bars indicate the negative pairs. The red lines denote the threshold where the difference between the Type 1 and Type 2 errors are minimal for each speaker, and the corresponding EERs (in %) are shown by the red text.

Fig. 2). As shown in the figure, there is a notable variation in the EERs across different speakers for each system, going as low as 2% even though the mean-aggregated EERs are between 25% and 35%.

Closer inspection of the values indicates an odd behavior: privacy of some speakers (`1673` and `652`) cannot be preserved by any of the systems, shown by the EER values less than 20% regardless of the employed anonymization system. This is quite counter-intuitive, given that the considered systems have very different strategies. B3 performs any-to-any voice conversion, where a non-existent target pseudo-identity is generated by a Wasserstein GAN. In contrast, B4 and B5 perform voice conversion to real speaker voices, precisely, any-to-few and any-to-one, respectively. What could have caused these speakers to be de-anonymized by the semi-informed attack in all considered cases?

B. Probabilistic Mixture of Anonymizers

One of the VPC 2024 participants explored probabilistic mixture of different anonymization systems, called AdMixture [7]. With a certain probability p a selection is made between two anonymization systems. This submission achieved one of the highest privacy scores when evaluated with the VPC 2024 semi-informed attacker, even though at least one of the systems were shown not to be as competitive.

Two design choices in the VPC 2024 semi-informed attacker could have caused this outcome, 1) During attacker training, ECAPA-TDNN trying to bin two anonymization systems together in the classifier layer and 2) During enrollment, the enrollment utterances are averaged, even though they are expected to be (at least) bimodal.

III. Proposed Attack(s)

A. Text-based attack based on BERT

A plausible hypothesis is that the semi-informed attacker ASV_{eval}^{anon} has accidentally learned to exploit the similarity of the linguistic content uttered by these speakers. To test this hypothesis, we have constructed a novel attacker system that operates only on text labels instead of on audio files. We build our attack based on the HuggingFace implementation [8] of the $\text{BERT}_{\text{BASE}}$ architecture, first introduced by [9]. Figure 3 outlines the training, enrollment, and trial phases, which are designed to be similar to how ASV_{eval}^{anon} works. We use `BertForSequenceClassification` as our starting point and override the loss function according to additive angular margin (AAM)-Softmax implementation of [10]. **This attack uses pre-trained BERT weights, which are not listed among the approved training resources.** We fine-tune on `train-clean-360` and validate on `libri-dev`. Training hyperparameters are summarized in Table I.

Hyperparameter	Value
Num. epochs	20
Batch size	32
Optimizer	AdamW
Learning rate	1e−4
Train-validation split	90%, 10%
AAM Margin	0.2
AAM Scale	30

TABLE I: Training hyperparameters

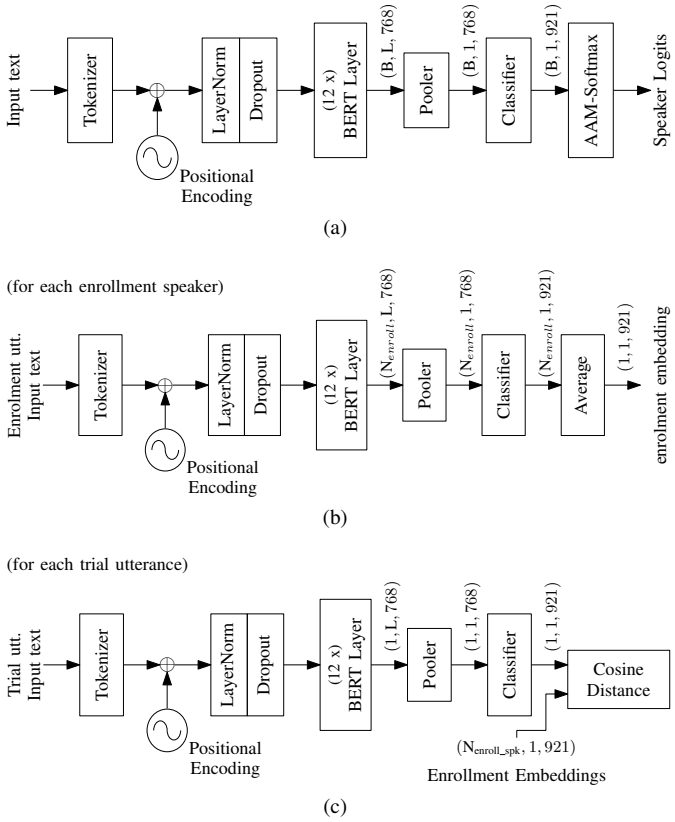


Fig. 3: The (a) training, (b) enrollment, and (c) trial phases of the proposed attack.

B. Cherry-picking attack

We propose a novel *cherry-picking* attack, by applying minor modifications to the ASV_{eval}^{anon}.

To address the training, we double the classifier unit count (921 to 1842). During training, we cherry-pick either the odd or even unit, depending on which has a higher activation. In code, it looks like this:

```
from speechbrain.nnet.losses import
    LogSoftmaxWrapper
class MultiLabelLogSoftmaxWrapper(
    LogSoftmaxWrapper):
    def __init__(self, loss_fn, k: int):
        super().__init__(loss_fn)
        self.k = k

    def forward(
        self,
        outputs,
        targets,
        length=None
    ):
        '''
        Arguments
        ---------
        outputs : torch.Tensor
            Network output tensor, of shape
            [batch, 1, K*n_spk]
        targets: torch.Tensor
            Target tensor, of shape [batch, 1]
        length: torch.Tensor
            The lengths of the corresponding
                inputs.
        '''
        # cherry-pick the best prediction
        # for each speaker
        N_spk = outputs.size(-1) // self.k
        outputs = outputs.view(-1, self.k, N_spk)
        # use max, not amax for gradient flow
        outputs = outputs.max(dim=1, keepdim=True)
            .values
        # the rest is the same as
            LogSoftmaxWrapper
        return super().forward(outputs, targets,
            length)
```

To remedy the averaging problem, we use the k-nearest neighbors (k = 1) to pick the enrollment vector. For each trial and for each candidate speaker, we select the embedding that yields the highest cosine similarity. To train this system, we use VPC 2024 `eval_post.yaml` without any further modification (same datasets, same seeds, same hyperparameters), staying within the challenge constraints.

IV. Results

A. Text-Based Attack

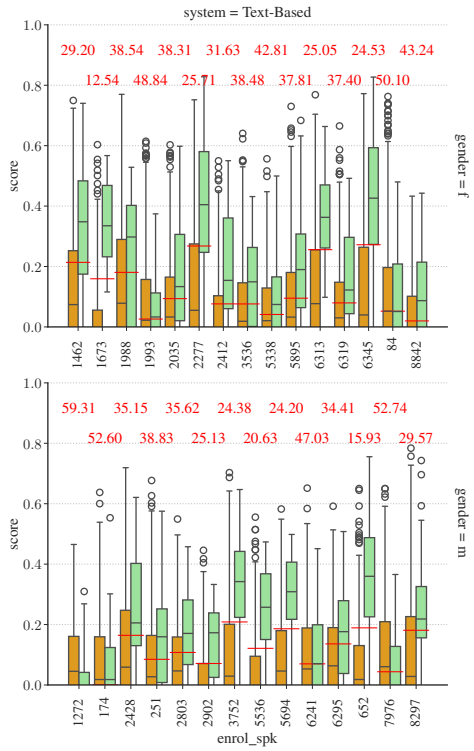
The score distributions of our attack are visualized in Fig. 4. Previously identified speakers, namely, `1673` and `652`, achieve an EER of 12.54% and 15.93%, showing that the text-based attack managed to compromise the anonymity of these speakers. An efficient way of comparing the attacks is to use radar plots, also called spider plots, which are presented in Figure 5. The vulnerable speakers cause an inward dent.

B. Cherry-picking attack

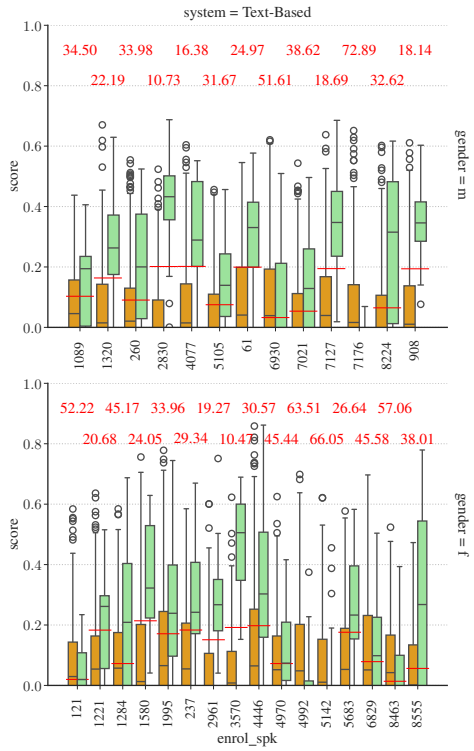
We show the results in Fig. 6.

C. Limitations and Future Work

This research has raised many questions that need further investigation, such as what exactly our attack has learned, and how `train-clean-360` could be sanitized such that the semi-informed attackers do not learn to exploit linguistic content similarity. In future work, we plan to address these questions. Furthermore, we were able to evaluate our cherry-picking attack only on T8-5, we plan to perform a more comprehensive evaluation.



(a)



(b)

Fig. 4: The results for the proposed text-based attack, on (a) `libri-dev` and (b) `libri-test`. Please refer to the caption of Fig. 2 for interpretation.

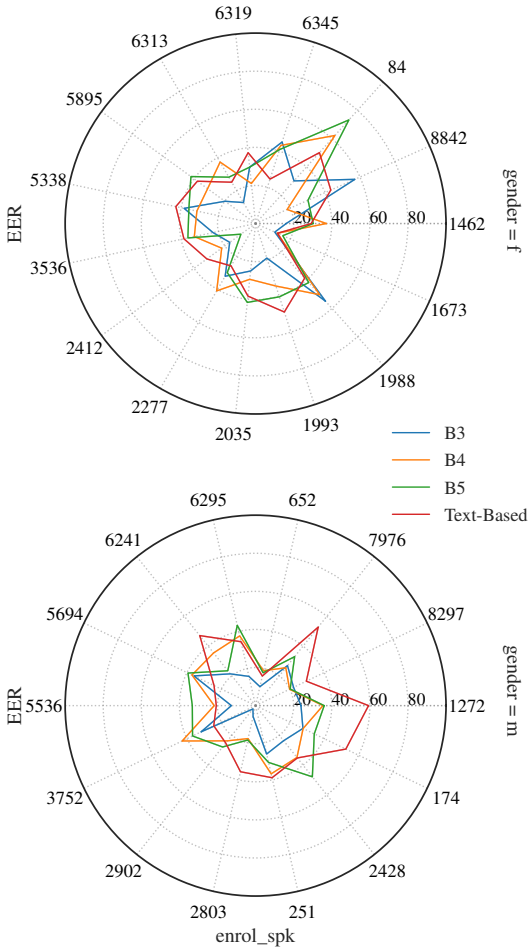


Fig. 5: Comparison of the EERs in various attack scenarios. `Text-Based` denotes our attack, and other entries denote the semi-informed attack on three speaker anonymization systems.

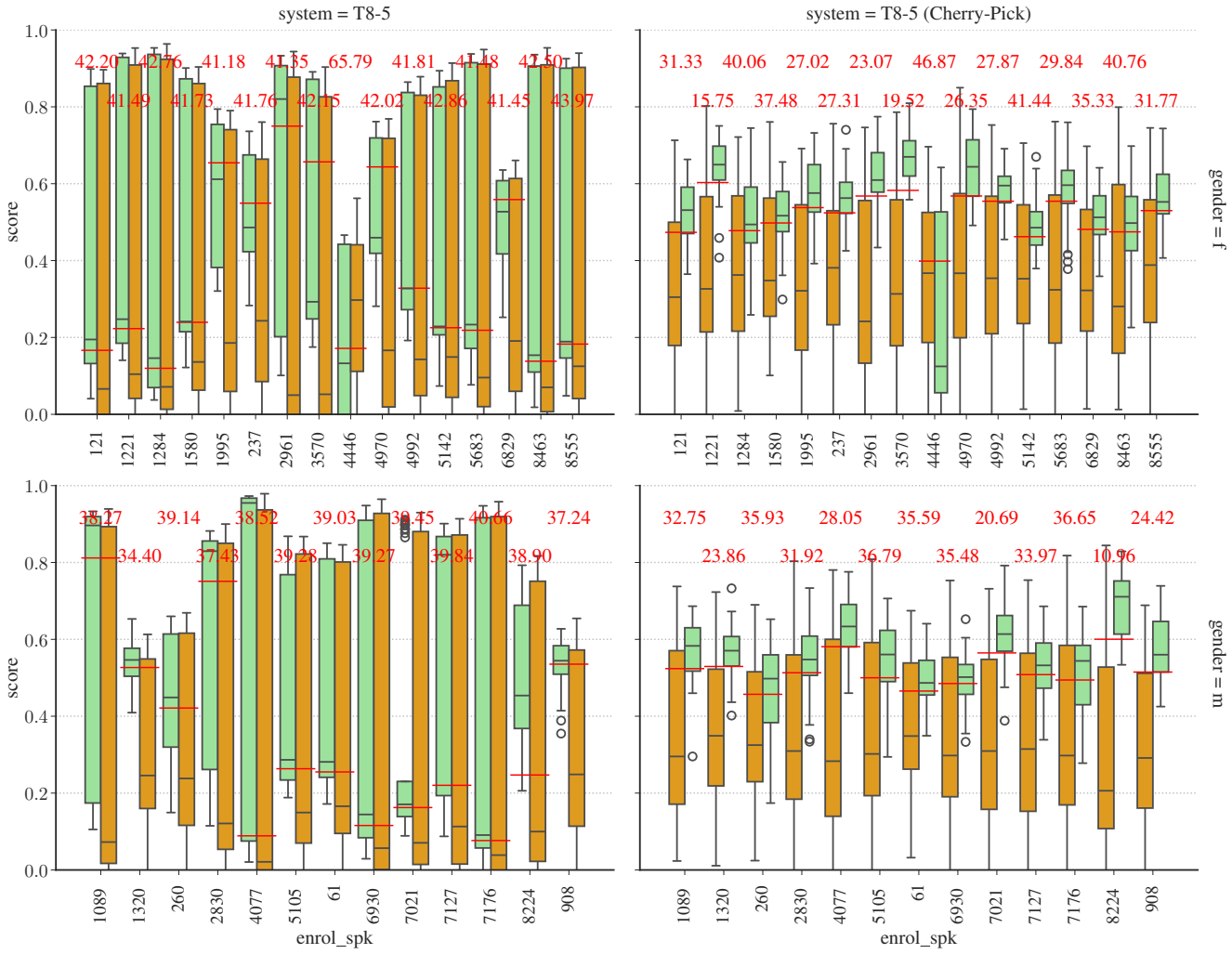


Fig. 6: The results for the VPC2024 semi-informed attack (left) and the proposed cherry-picking attack (right), on `libri-test`, applied to T8-5. Please refer to the caption of Fig. 2 for interpretation.

V. References

[1] N. Tomashenko *et al.*, "Introducing the VoicePrivacy Initiative," in *Proc. Interspeech Conf.*, 2020.

[2] Pierre Champion *et al.*, *3rd VoicePrivacy Challenge Evaluation Plan (Version 2.0)*, 2024.

[3] N. Tomashenko, X. Miao, E. Vincent, J. Yamagishi, and N. Evans, *The first VoicePrivacy Attacker Challenge evaluation plan (version 2.2)*, 2024.

[4] J. Williams, K. Pizzi, N. Tomashenko, and S. Das, "Anonymizing Speaker Voices: Easy to Imitate, Difficult to Recognize?" In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[5] S. Tayebi Arasteh *et al.*, "Addressing challenges in speaker anonymization to maintain utility while ensuring privacy of pathological speech," *Nature Communications Medicine*, vol. 4, no. 1, 2024.

[6] P. Champion, "Anonymizing Speech: Evaluating and Designing Speaker Anonymization Techniques," PhD thesis, Universite de Lorraine, 2024.

[7] H. L. Xinyuan *et al.*, "HLTCOE JHU Submission to the Voice Privacy Challenge 2024," 2024.

[8] T. Wolf *et al.*, "Transformers: State-of-the-Art Natural Language Processing," in *Proc. Conf. Empirical Methods in Natural Language Processing: System Demonstrations*, Q. Liu and D. Schlangen, Eds., 2020.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.

[10] M. Ravanelli *et al.*, "Open-Source Conversational AI with SpeechBrain 1.0," *Journal of Machine Learning Research*, vol. 25, no. 333, 2024.