THE VICOMTECH AUDIO DEEPFAKE DETECTION SYSTEM BASED ON WAV2VEC2 FOR THE 2022 ADD CHALLENGE

Juan M. Martín-Doñ as, Aitor Álvarez
Index

- Audio Deep synthesis Detection (ADD) Challenge 2022
- Wav2Vec2-based proposed system
- Experimental framework
- Experimental results
- Conclusions and future work
Audio Deep synthesis Detection (ADD) Challenge 2022

- Recent growth of deep learning based text-to-speech (TTS) synthesis and voice conversion (VC) technologies
- Generation of deepfake speech. Malicious use: foolish human or even automatic speaker verification systems
- Need reliable countermeasures. Audio deepfake detection systems.
  - Example: ASVspoof series (2015-2021)
- Great improvements achieved, interest in more challenging scenarios:
  - Noisy and reverberant scenarios
  - Speech modified through different channels, codecs or compression algorithms
  - Partially spoofed audio
2022 Audio Deep synthesis Detection (ADD) Challenge: Detection of deep synthesis and manipulated audios in different scenarios

- **Track 1**: Low-quality fake audio detection
- **Track 2**: Partially fake audio detection
- **Track 3**: Audio fake game

Vicomtech proposed audio deepfake detection system for Tracks 1 and 2:

- **Wav2Vec2** pre-trained feature extractor
- Downstream classifier model trained for deepfake detection
- **Data augmentation** techniques to adapt the classifier
- **Winners of Track 1** and fourth position in Track 2
- Competitive results in ASVspoof 2021 Challenge
Wav2Vec2-based proposed system

- **W2V2 Feature Extractor:**
  - Cross-lingual Large models (XLS) 300M parameters
    - 53 and 128 languages
  - **Self-supervised** learning with contrastive loss
    - Masked encoded features
    - Predict quantized representations from contextualized ones
  - **Pre-trained** model
    - Freeze W2V2 parameters
    - Finetuning downstream classifier
Wav2Vec2-based proposed system

▶ Classification Model:
  - Representations from $L = 25$ transformer layers
    - $o_t = \sum_{l=0}^{L} \alpha_l h_{t,l}$
  - **Attentive** statistical temporal pooling (mean and std dev.)
  - Compute embedding $e$
  - **Cosine scoring** layer
    - $S = \cos(w, e) \in [-1, 1]$
  - **One-class** softmax loss function

<table>
<thead>
<tr>
<th>Layer name</th>
<th>Output size</th>
</tr>
</thead>
<tbody>
<tr>
<td>W2V2 features</td>
<td>$N \times T \times 1024 \times 25$</td>
</tr>
<tr>
<td>Temp. Norm. + Layer weight.</td>
<td>$N \times T \times 1024$</td>
</tr>
<tr>
<td>FF Layer (1 and 2)</td>
<td>$N \times T \times 128$</td>
</tr>
<tr>
<td>FF Linear</td>
<td>$N \times 128$</td>
</tr>
<tr>
<td>Cosine Layer</td>
<td>$N$</td>
</tr>
</tbody>
</table>
Experimental framework

- **ADD 2022 database:**
  - Genuine and TTS/VC speech from AISHELL-3 speech corpus
  - Training and development clean speech (≈ 28K utt. each)
  - Adaptation (≈ 1K) and test (≈ 100K) sets for each track
    - **Track 1:** Real-world noises and background music
    - **Track 2:** Partial fake manipulation using real or synthesized audios

- **ASVspoof 2021 database:**
  - Train and development sets from ASVspoof 2019 LA (TTS/VC)
  - Logical Access (LA): Transmission through real telephonic systems
  - Speech Deepfake (DF): Processed speech with commercial audio codecs

- **Adaptation and data augmentation techniques:**
  - Low-pass FIR filtering (narrowband and wideband): Frequency masking
  - ADD 2022: Training using train and adaptation sets
  - Track 2 (ADD): Generating new partial deepfakes by audio overlapping
Experimental results

Results on ADD 2022:
- XLS-128 outperforms XLS-53
- Few adaptation data help (main improvements)
- Generated partial deepfakes in Track 2 improve further the model performance
- Narrowband FIR filtering reduce 1% EER in both tracks
- Competitive system:
  - T1: 21.7% EER (1st)
  - T2: 16.6% EER (4th)

<table>
<thead>
<tr>
<th>W2V2</th>
<th>Sets</th>
<th>DA</th>
<th>Track1</th>
<th>Track2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>-</td>
<td>32.96</td>
<td>38.09</td>
</tr>
<tr>
<td></td>
<td>Tr.+Adap.</td>
<td>-</td>
<td>23.70</td>
<td>33.73</td>
</tr>
<tr>
<td>XLS-53</td>
<td>Train</td>
<td>-</td>
<td>32.20</td>
<td>45.88</td>
</tr>
<tr>
<td></td>
<td>Tr.+Adap.</td>
<td>-</td>
<td>22.62</td>
<td>30.35</td>
</tr>
<tr>
<td>XLS-128</td>
<td>Tr.+Adap.</td>
<td>FIR</td>
<td>21.71</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Tr.+Adap.</td>
<td>partial</td>
<td>-</td>
<td>17.58</td>
</tr>
<tr>
<td></td>
<td>Tr.+Adap.</td>
<td>FIR+part.</td>
<td>-</td>
<td>16.59</td>
</tr>
</tbody>
</table>
Experimental results

- Results on **ASVspoof 2021**:
  - XLS-128 also outperforms XLS-53 model
  - Narrowband FIR for LA
    - 8 kHz bandwidth telephone channel
  - Wideband FIR for DF
    - Emulate general audio codecs

<table>
<thead>
<tr>
<th>W2V2 model</th>
<th>Data augmentation</th>
<th>LA</th>
<th>DF</th>
</tr>
</thead>
<tbody>
<tr>
<td>XLS-53</td>
<td>FIR-NB</td>
<td>4.34</td>
<td>11.27</td>
</tr>
<tr>
<td></td>
<td>FIR-WB</td>
<td>4.98</td>
<td>6.99</td>
</tr>
<tr>
<td>XLS-128</td>
<td>FIR-NB</td>
<td>3.54</td>
<td>6.18</td>
</tr>
<tr>
<td></td>
<td>FIR-WB</td>
<td>7.08</td>
<td>4.98</td>
</tr>
</tbody>
</table>
Experimental results

- Previous ASVspoof 2021 results:
  - *Ensemble* classifiers with robust neural models (LCNN, ECAPA, ResNet)
    - Poor generalization on DF set (other speech databases)
  - Previous W2V2 only used representations from *last layer* (need finetuning)
  - Our proposal uses general *W2V2 feature extractor* with specialized downstream model (using data augmentation)

<table>
<thead>
<tr>
<th>System</th>
<th>LA</th>
<th>DF</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCNN+ResNet+RawNet</td>
<td>1.32</td>
<td>15.64</td>
</tr>
<tr>
<td>GMM+LCNN (Ensemble)</td>
<td>3.62</td>
<td>18.30</td>
</tr>
<tr>
<td>ECAPA-TDNN (Ensemble)</td>
<td>5.46</td>
<td>20.33</td>
</tr>
<tr>
<td>ResNet (Ensemble)</td>
<td>3.21</td>
<td>16.05</td>
</tr>
<tr>
<td>W2V2 (fixed)+LCNN+BLSTM</td>
<td>10.97</td>
<td>7.14</td>
</tr>
<tr>
<td>W2V2 (finetuned)+LCNN+BLSTM</td>
<td>7.18</td>
<td>5.44</td>
</tr>
<tr>
<td><em>Proposed system</em></td>
<td>3.54</td>
<td>4.98</td>
</tr>
</tbody>
</table>
Experimental results

- **Weight values $\alpha_l$ for the transformer layers:**
  - The information from different layers is used for deepfake detection
  - Different layer weights depending on the scenario
  - Example: For DF, special focus around layers 6 and 19
Conclusions and future work

- Our approach effectively exploits the contextualized representation from the different transformer layers of a pre-trained Wav2Vec2 model.
- The downstream classifier can be finetuned using these representations and adapted through adequate data augmentation techniques.
- Our system shows competitive results in both ASVspoof 2021 (especially in DF task) and ADD 2022 challenges (winner of Track 1).
- **Future work**: Test other self-supervised models and additional data augmentation techniques.
Thank you!

Contact: Juan M. Martin (jmmartin@vicomtech.org)