A Study on the Use of wav2vec Representations for Multiclass Audio Segmentation

Pablo Gimeno, Alfonso Ortega, Antonio Miguel, Eduardo Lleida
ViVoLab, Aragón Institute for Engineering Research (I3A), University of Zaragoza
{pablog, ortega, amiguel, lleida}@unizar.es

Abstract
This paper presents a study on the use of new unsupervised representations through wav2vec models seeking to jointly model speech and music fragments of audio signals in a multiclass audio segmentation task. Previous studies have already described the capabilities of deep neural networks in binary and multiclass audio segmentation tasks. Particularly, the separation of speech, music and noise signals through audio segmentation shows competitive results using a combination of perceptual and musical features as input to a neural network. Wav2vec representations have been successfully applied to several speech processing applications. In this study, they are considered for the multiclass audio segmentation task presented in the Albayzin 2010 evaluation. We compare the use of different representations obtained through unsupervised learning with our previous results in this database using a traditional set of features under different conditions. Experimental results show that wav2vec representations can improve the performance of audio segmentation systems for classes containing speech, while showing a degradation in the segmentation of isolated music. This trend is consistent among all experiments developed. On average, the use of unsupervised representation learning leads to a relative improvement close to 6.8% on the segmentation task.

Index Terms: multiclass audio segmentation, unsupervised representation learning, wav2vec

1. Introduction
The goal of a generic audio segmentation system is to provide a sequence of labels that isolates different regions in an input audio signal according to the characteristics described in a predefined set of classes, e.g., speech, music or noise. So far, a number of different methods have been applied to audio segmentation. Literature reviews usually make a separation between two kind of systems: those based on the segmentation & classification approach, and those based on the segmentation by classification approach [1]. The first approach is usually known as distance-based due to the use of a distance metric to obtain class boundaries. Then, segments obtained are classified in a posterior stage. In the segmentation by classification approach, systems obtain its segmentation hypotheses as a direct sequence of decisions over the input audio. Traditionally statistical approaches have been applied to audio segmentation tasks [2] [3], however current state-of-the-art is based on deep learning models [4] [5]. Audio segmentation systems described in this paper are based on the segmentation by classification paradigm, relying mainly on deep learning solutions.

Supervised learning algorithms have played a crucial role in the evolution of deep learning applications, however the process of obtaining labelled data is very costly, and these models are strongly dependent on the amount of data they were trained on. Under these circumstances, unsupervised learning methods seek to reduce the need for labelled data by presenting a model with large amounts of unlabelled data. The model then learns to characterise input data by predicting different aspects of it. First occurrences of this idea happened in discrete data such as text, forcing a model to generate the next items [6]. In the case of real valued signals, the initial approach was based on the minimisation of the reconstruction error of the signal [7]. Nevertheless, recent works have demonstrated that the use of pretext tasks, where the objective of the system is to solve a prediction as a classification, can lead to better representations. In many works, this target is to select an unseen fragment of the signal among other randomly selected distractor fragments [8].

Unsupervised learning representation has become an active research field in the audio and speech processing community, with different approaches and techniques being applied in different tasks [9] [10]. This strategy has been successfully developed with competitive performance in tasks such as speech recognition [11] [12], or speaker verification [13]. Concerning audio segmentation applications, we can find examples of these methods being applied successfully in binary audio segmentation tasks, such as speech activity detection [14], showing relevant results specially in scenarios with mismatched train and evaluation data. Another example can be found in [15], that applies contrastive learning to a phoneme segmentation task.

Our previous experience in multiclass audio segmentation already demonstrated the capabilities of deep neural networks in the Albayzin 2010 dataset, seeking to separate speech, music, voice and a combination of them [16]. This work explored a number of neural architectures based on the use of bidirectional LSTM layers, successfully introducing a temporal pooling mechanism that improved the performance of the system without increasing the number of parameters. This paper builds on the knowledge acquired from these previous experiences, aiming to introduce the recent advances in unsupervised representation learning in the audio segmentation task. While all the previous experiments relied on the use of a combination of different sets of features, namely perceptual features such as log Mel filterbanks, or musical features such as chroma, the approach presented in this paper aims to replace these features with representations obtained through unsupervised learning.

The remainder of the paper is organised as follows: section 2 provides an overview of the full audio segmentation system, from the wav2vec representation learning framework to the neural network classifiers. Section 3 describes the experimental setup for this paper, focusing on the datasets and the metrics for evaluation. Results are presented and commented in section 4. Finally, an overview of the paper and some conclusions are provided in section 5.
2. System description

2.1. wav2vec representation learning

The original wav2vec system was presented in [11] in the context of speech recognition, following the ideas described in [8] about contrastive predictive coding. Our implementation is based on those works but with some variations that we explain in this section. A general overview of the system can be observed in the block diagram depicted in Figure 1. As it can be seen, the system is made of two different parts: a convolutional neural network (CNN) encoder and a recurrent neural network (RNN). The CNN encoder is composed of 10 strided 1D convolutional layers whose task is to map the input sequence \( X_t \) into a latent space \( Z_t \). This encoder works on top of the 16 kHz raw waveform, providing an output feature vector every 10 ms, as the total downsampling factor of the encoder is 160. The second main part of the system is a RNN, which in our implementation is a single layer gated recurrent unit (GRU) [17] with a hidden state of dimension 512. The output of the GRU layer for every temporal instant serves as the context embedding \( C_t \) that the system uses to predict 8 timesteps via a contrastive loss.

BERT text models [18] and other works such as Decoar [19] already proposed the use of bidirectional embeddings in order to estimate previous and posterior timesteps, resulting in a loss term for each direction. Our wav2vec system follows this approach using a bidirectional GRU and computing the final loss as the sum of right and left embedding predictions. Thereby, our context embedding is a 1024-dimensional array that, after training, is extracted and used as the learned features. In the classifier training stage, the wav2vec model is frozen, so no fine tuning is performed.

Our approach to prediction differs slightly from the one presented in [11]. We use a single head with a single hidden layer [20] in order to predict future and past frames. Some of the different wav2vec models evaluated in this work use data augmentation techniques in training time. In those cases, augmented versions of the context embedding \( C_t \) serves as input to predict an unmodified reference \( Z_t \). We add noises from MUSAN [21] database sampling a uniform distribution in the range \((3,15) \text{ dB}\) to obtain the signal to noise ratio (SNR). Furthermore, a variety of room impulse responses (RIR) are simulated using the gprRIR toolkit [22]. The influence of data augmentation is evaluated by training different wav2vec models with or without this augmentation pipeline.

2.2. Neural network classifiers

Inspired by our previous work on the Albayzín 2010 audio segmentation task [16], the neural network architectures used in this work are based on the use of BLSTM layers. Specifically, we evaluate three different variants with increasing complexity that have been proven to show relevant results and that we describe in the following lines:

- **1BLSTM**: a single BLSTM layer is used to process the input. Final classification scores are obtained through a linear layer. One segmentation label is emitted for every frame at the input, this is one each 10 milliseconds.
- **2BLSTM pool**: a BLSTM layer is followed by an average pooling mechanism on the temporal dimension as a way to generate a smoother output score. The output of the average pooling is then fed to another BLSTM layer working at one tenth of the original frame rate. As in the previous architecture, final classification scores are obtained with a linear layer. Due to the average pooling, this model produces a segmentation label every 100 milliseconds.
- **2BLSTM pool + mixup**: this alternative incorporates mixup augmentation [23] on top of the 2BLSTM pool architecture. Mixup is a data agnostic augmentation technique that generates new virtual examples through linear combinations of examples from training data. Weights for this combination are usually sampled from a beta distribution \( X \sim \text{Beta}(\alpha,\beta) \) with parameters \( \alpha = \beta \).

Note that the wav2vec model is frozen during the classifier training stage and, therefore, is used only as a feature extractor. All classifier models have been trained using Adam optimiser, with a learning rate that decays exponentially from \( 10^{-3} \) to \( 10^{-4} \) during the 20 epochs that data is presented. Cross entropy loss is used as training objective as usually done in classification tasks.

3. Experimental setup

The following section describes the datasets and metrics used in this work. For the data description we make a separation between unlabeled data used to train the wav2vec system and the annotated data used to train the neural network classifiers. Note that unlike our previous work in [16], this work does not follow Albayzín 2010 evaluation conditions because we included several external datasets apart from the one provided for the original challenge.

3.1. Data for unsupervised feature representation

We distinguish three different wav2vec models depending on how they have been trained. Particularly, we differentiate if data augmentation has been used or not, and consider two training conditions: a first one trained with a smaller amount of data, and a second one using a larger amount of data in training. With these considerations three different models were evaluated:

- **wav2vec base**: this model was trained using mainly audio data in Spanish. Data from Albayzin 2018 and 2020 evaluations [24] and from the Spanish partition of the Commonvoice 2 dataset was used. This makes a total of around 1200 hours of unsupervised audio for training.
- **wav2vec base no aug**: this model was trained on the exact same training data as the wav2vec base model with
the only difference that no data augmentation of any kind was applied in training time.

- **wav2vec extended no aug**: A larger amount of data was added to the datasets already described in the base model. Several well known English corpora were included in the training data: LibriSpeech, LibriLight, Tedlium and VoxCeleb 1&2. This makes a total of around 60000 hours of speech audio. Additionally, a dataset of music scrapped from different sources, and presenting around 1700 hours of audio, was also considered, seeking to improve the modelling capabilities of the system in classes containing music. As done in the previous model, no data augmentation was applied in training time.

### 3.2. Data for multiclass audio segmentation

The multiclass audio segmentation task described in this paper was firstly proposed in the Albayzin 2010 dataset [25]. The full database includes 87 hours of audio sampled at 16 KHz and belonging to the broadcast news domain. The database was separated into two parts, using two thirds of the data for training and the remaining third for testing. The database features five different acoustic classes with the following unbalanced distribution: 37% for clean speech (sp), 5% for music (mu), 15% for speech over music (sm), 40% for speech over noise (sn) and 3% for others (ot). The class “others” is not evaluated in the final test.

### 3.3. Metric & evaluation protocol

In order to evaluate our results we follow the same metric as the one proposed in the original Albayzin 2010 evaluation. As shown in the following equation, the metric represents the average error obtained over all the acoustic classes:

$$\text{Avg error} = \frac{1}{|C|} \sum_{i \in C} \frac{\text{dur}(\text{miss}_i) + \text{dur}(\text{fa}_i)}{\text{dur}(\text{ref})}, \quad (1)$$

where $C$ is the set of acoustic classes defined in the evaluation, $\text{dur}(\text{miss}_i)$ is the total duration of all miss errors for the $i$th acoustic class, $\text{dur}(\text{fa}_i)$ is the total duration of all false alarm errors for the $i$th acoustic class, and $\text{dur}(\text{ref})$ is the total duration of the $i$th acoustic class according to the reference. Seeking to avoid uncertainty in class transitions, a 1 second collar is not scored around each reference boundary.

### 4. Results

As a comparison to the results obtained using the presented wav2vec representations, we also evaluate the neural network classifiers using a set of traditional features that combines log Mel filterbank energies and chroma features. Namely, we concatenate 80 log Mel coefficients and the log energy of the frame with 12 chroma coefficients. Furthermore, first and second derivatives are computed and concatenated to the input. Experimental results showed that this frontend can lead to competitive results in the multiclass audio segmentation task [16].

In a first approximation, seeking to perform an exploratory analysis of the wav2vec representations capabilities, a t-SNE [26] dimensionality reduction of the best performing wav2vec system is computed in order to obtain a 2D visualisation. These features are also compared to the traditional frontend based on log Mel and chroma features. The mentioned representation extracted for the validation partition of the Albayzin 2010 dataset is presented in Figure 2. It can be observed that the use of a wav2vec representation provides a 2D t-SNE plane that is much more easily separable. The t-SNE plane obtained for the log Mel and chroma features shows only a limited number of clusters that can be identified, e.g., for the class others in orange. On the other hand, the wav2vec representation results in a plane where separation between classes is much more distinguishable, and that shows no significant overlap between classes such as speech and music , clean speech and speech and noise. In the case of the wav2vec system, the music class seems to obtain the case of the wav2vec system, the music class seems to obtain the best performance.

<table>
<thead>
<tr>
<th>Features</th>
<th>Class err.(%)</th>
<th>Avg err.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>log Mel + chroma</td>
<td>16.28</td>
<td>28.82</td>
</tr>
<tr>
<td>log Mel + chroma + deriv</td>
<td>21.46</td>
<td>28.09</td>
</tr>
<tr>
<td>wav2vec</td>
<td>25.84</td>
<td></td>
</tr>
<tr>
<td>wav2vec base</td>
<td>26.72</td>
<td></td>
</tr>
<tr>
<td>wav2vec base no aug</td>
<td>26.32</td>
<td></td>
</tr>
<tr>
<td>wav2vec extended no aug</td>
<td>30.35</td>
<td></td>
</tr>
<tr>
<td>wav2vec extended</td>
<td>23.72</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Error per class and average class error for the 1BLSTM classifier on the Albayzin 2010 test partition for different frontend configurations.
mainly motivated by a large increase in the music class classification error. Furthermore, class error in the rest of classes remains similar, making that the average error increases due to both facts mentioned. In the case of the wav2vec base no aug model, which used no data augmentation in the pretraining stage of the wav2vec system, the error in the music and speech with music classes decreases compared to the model using data augmentation. Even though these differences are significant, they do not affect in a major way the average error observed, that remains similar to one from the version using data augmentation. Best results are observed when incorporating more data in the pretraining phase of the wav2vec system. The wav2vec extended noaug version yields a relative improvement of 5.2%, 16.03% and 6.7% respectively in the classes speech, speech with music and speech with music compared to the baseline frontend configuration. The music class error rate still remains high compared to the performance of the baseline system, limiting the overall performance of the wav2vec representations to a 5.3% relative improvement in the average error metric.

Tables 2 and 3 describe the results obtained using the 2BLSTM pool and 2BLSTM pool + mixup classifiers respectively on the Albayzín 2010 test partition comparing traditional features and the best performing wav2vec representation.

### Table 2: Error per class and average class error for the 2BLSTM pool classifier on the Albayzín 2010 test partition comparing traditional features and the best performing wav2vec representation.

<table>
<thead>
<tr>
<th>Features</th>
<th>Class err.(%)</th>
<th>Avg err.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>mu sp sm sn</td>
<td></td>
</tr>
<tr>
<td>Mel,chr, deriv</td>
<td>15.55 29.16 24.34 30.51</td>
<td>24.89</td>
</tr>
<tr>
<td>w2v extended</td>
<td>19.29 24.75 21.96 26.69</td>
<td>23.24</td>
</tr>
</tbody>
</table>

Concerning experiments depicted in Table 3, we evaluate the mixup augmentation strategy under the same conditions as described in [16]: applied on the feature space, and using 3 different $\alpha$ values 0.1, 0.2 and 0.3. The best performance is obtained using a setup with $\alpha = 0.2$ for both the traditional set of features and the wav2vec representation. Furthermore, all setups showed an improvement when using mixup augmentation and the wav2vec representation. The best result in this paper using a wav2vec system is obtained with the 2BLSTM pool architecture, mixup augmentation and $\alpha = 0.2$, achieving a relative improvement close to 6.8% when compared to the same setup using log Mel and chroma features. Even though wav2vec models show better performance than traditional features in classes containing speech, experimental results suggest that one limitation of the wav2vec representations evaluated come from its capacity to model music fragments, underperforming traditional features in all the experiments shown. This fact strongly limits the boost in average performance observed, that could be improved by decreasing music class error.

### Table 3: Error per class and average class error for the 2BLSTM pool + mixup classifier on the Albayzín 2010 test partition comparing traditional features and the best performing wav2vec representation.

<table>
<thead>
<tr>
<th>Features</th>
<th>$\alpha$</th>
<th>Class err.(%)</th>
<th>Avg err.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>mu sp sm sn</td>
<td></td>
</tr>
<tr>
<td>Mel,chr, deriv</td>
<td>0.1</td>
<td>15.21 27.99 23.05 29.34</td>
<td>23.90</td>
</tr>
<tr>
<td>w2v extended</td>
<td>0.2</td>
<td>14.64 28.20 22.01 29.01</td>
<td>23.56</td>
</tr>
<tr>
<td>na</td>
<td></td>
<td>17.41 24.58 20.01 26.23</td>
<td>22.06</td>
</tr>
<tr>
<td>Mel,chr, deriv</td>
<td>0.3</td>
<td>16.03 26.36 23.89 28.82</td>
<td>23.62</td>
</tr>
<tr>
<td>w2v extended</td>
<td>0.3</td>
<td>18.66 25.70 20.78 26.65</td>
<td>22.95</td>
</tr>
</tbody>
</table>

5. Conclusions

In this paper we have presented a study on the applications of self supervised learned features to the multiclass audio segmentation task. Namely, we have evaluated different wav2vec models seeking to jointly discriminate audio information belonging to speech, music and noise classes in the Albayzín 2010 dataset. Following previous studies that showed competitive performance using a combination of log Mel and chroma features, we apply the same neural network classifier models using the new wav2vec features seeking to obtain a comparable experimental setup. The task is evaluated using the average error among all acoustic classes as done in the original evaluation.

Experimental results demonstrate that the use of wav2vec representations can lead to a significant improvement in the performance of audio segmentation systems for classes containing speech. However, a degradation in performance is observed in the segmentation of isolated music when compared to traditional features. This limits the overall improvement observed to a relative improvement close to 6.8% on the Albayzín 2010 segmentation task. Further research should investigate the learning of more robust audio representations, capable of dealing with speech and music simultaneously. Some research is already working towards that direction [27] [28].

6. Acknowledgements

This work was supported in part by the European Union’s Horizon 2020 research and innovation programme under Marie Skłodowska-Curie Grant 101007666; in part by MCIIN/AEI/10.13039/501100011033 and by the European Union “NextGenerationEU”/PRTR under Grant PDC2021-120846-C41, and in part by the Government of Aragón (Grant Group T36_20R).
7. References


