Improved Masking Strategies for Self-Supervised Speech Representation Learning

Anuj Diwan, David Harwath, Eunsol Choi
Department of Computer Science,
University of Texas at Austin
Background

- **Self-supervised** speech representation learning involves training a model with lots of unlabeled speech data to generate powerful latent representations.
- In this work, we focus on HuBERT and Masked Language Modelling approaches for speech.
HuBERT

- Involves **masking** a random subset of the input (speech frames) sequence and tasking the model with **predicting** a *discretized* version of the masked input
- Similar approach to Masked Language Models like BERT
RandomFrameSpan Masking

- We can call the masking strategy used by HuBERT as *RandomFrameSpan Masking*
- This masking strategy is fairly simple:
  - randomly sample a proportion $p$ of all input timesteps to be start indices of masking spans
  - mask a fixed number $M$ of timesteps starting from each start index
  - In HuBERT, $p = 0.08$, $M = 10$

![RandomFrameSpan Masking Diagram](image)
Better Masking Strategies in textual NLP

- **Random-Token Masking**: Original BERT; pick subword tokens randomly
- **Random-Span Masking**: Sample a span length and span start index randomly, mask entire span (consisting of multiple words)
- **Knowledge Masking** and **Salient Span Masking**: Use parsers to identify meaningful entities/phrases and mask these.

*Can we propose improved masking strategies for speech as well?*
RandomPhoneme Masking

- We propose a **new masking strategy** where we mask **spans of entire phonemes**
- Given a list of phoneme boundaries for each utterance i.e. the start and end frame indices for each phoneme in the utterance.
  - fix a **proportion** $q$ of total frames we intend to mask and a phoneme **span length** $m$
  - randomly sample a phoneme index $i$ and mask out all phonemes from $i$ to $i+m-1$. Repeatedly do this until the total frames masked hit the proportion $q$
  - $q=0.56$ is chosen such that the number of frames masked is approximately the same as RandomFrameSpan masking
- This is a more linguistically-driven masking strategy that intuitively should result in a harder pretraining loss

*How to obtain a list of phoneme boundaries for unlabelled data?*
Phoneme Segmentation

- The phoneme segmentation task involves segmenting a given input speech utterance into its constituent phonemes i.e. outputting a sequence of phoneme boundaries
Since we are developing a masking strategy for SSL, we cannot assume access to ground truth text data.

We use an unsupervised phoneme segmentation strategy proposed by Kreuk et. al. [6]. This approach neither requires labels for training nor for inference.

The algorithm uses contrastive loss to learn latent frame representations that distinguish adjacent frame pairs from non-adjacent frame pairs.

During inference, a similarity score is computed for each adjacent frame pair and the lowest similarity score pairs are identified as phoneme boundaries.
 Supervised Phoneme Segmentation

- If one has access to the *ground truth text* of the speech utterance as well as a phonetic dictionary that maps words to phonemes (like in the traditional ASR pipeline), one can easily extract phoneme boundaries by:
  - Training a traditional HMM-based HCLG ASR system on labelled speech data
  - For a given speech utterance, run forced Viterbi alignment using the trained model on the speech and the corresponding ground truth phoneme sequence. This will time-align the ground truth phoneme sequence, giving phoneme boundaries as desired
- This requires access to ground truth text data for both the training set used to train the ASR system and the test set whose boundaries need to be found
- Kaldi [5] has off-the-shelf scripts to run forced Viterbi alignment
- This is an 'oracle' experiment; uses near-perfect phoneme boundaries
Evaluation

- The SUPERB [4] benchmark consists of a set of downstream speech tasks that can be used to evaluate pretrained speech models.
- We focus on ASR (Automatic Speech Recognition), PR (Phoneme Recognition), KS (Keyword Spotting). The metrics are WER (lower is better), PER (lower is better), Accuracy (higher is better) respectively.
- For each task, the model parameters are frozen. Then,
  - All layers of the model are summed (with learnable weights for each layer) to generate the final representation
  - A task-specific head is placed on top of this representation for generating the task output.
Experimental Setup - Pretraining and Eval

• **Datasets**
  ○ We use the 960-hr Librispeech data for pretraining the model

• **Model**
  ○ We use the HuBERT Base model for all our experiments. We train our k-means clusters using the 6th layer of the Facebook-provided pretrained checkpoint.
  ○ We modify the HuBERT dataloader and training code from fairseq [8] to support phoneme-based masking.
  ○ We initialize our pretraining expts using the Facebook-provided pretrained checkpoint rather than pretraining from scratch. We train for 40k additional steps.

• **Evaluation**
  ○ We use the provided SUPERB scripts in the s3prl toolkit.
Experimental Setup - Phoneme Segm.

- **Unsupervised Approach**
  - Off-the-shelf phoneme segmentation model released by Kreuk. et. al. trained on Buckeye [7] corpus and train-other-500 set of Librispeech. We run the entire Librispeech corpus through the model to generate phoneme segmentations

- **Supervised Approach**
  - We train a TDNN-HMM ASR model using the standard Kaldi Librispeech ASR recipe on 960-hr Librispeech train set
  - We run forced Viterbi alignment using the align_fmllr.sh Kaldi script on the Librispeech corpus
## Experimental Results

<table>
<thead>
<tr>
<th>Masking Strategy</th>
<th>Span Length</th>
<th>Boundary Type</th>
<th>ASR (WER)</th>
<th>PR (PER)</th>
<th>KS (Acc)</th>
</tr>
</thead>
<tbody>
<tr>
<td>None (original ckpt)</td>
<td>-</td>
<td>-</td>
<td>7.09</td>
<td>6.10</td>
<td>96.55</td>
</tr>
<tr>
<td>RandFrameSpan</td>
<td>-</td>
<td>-</td>
<td>7.18</td>
<td>5.61</td>
<td>96.62</td>
</tr>
<tr>
<td>RandPhoneme</td>
<td>1</td>
<td>Supervised</td>
<td>7.13</td>
<td>5.57</td>
<td>96.72</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Supervised</td>
<td>7.11</td>
<td>5.53</td>
<td><strong>96.88</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Unsupervised</td>
<td><strong>7.05</strong></td>
<td><strong>5.51</strong></td>
<td>96.52</td>
</tr>
</tbody>
</table>
Future Work

- Training for more steps to (hopefully) demonstrate larger gains
- Data-driven analysis (like PMI Masking) to find spans that are potentially even more meaningful than phonemes/resemble phonemes
- Reducing dependency on external tools like phoneme segmentation using the above
- Using phoneme-based ideas to change the discretization strategy itself (force all frames within a phoneme to have the same discrete index, for example)
References


References

