SLICER: Learning universal audio representations using low-resource self-supervised pre-training

Ashish Seth\textsuperscript{1,}*, Sreyan Ghosh\textsuperscript{2,}*, S.Umesh\textsuperscript{1,}*, Dinesh Manocha\textsuperscript{2}
\textsuperscript{1}Indian Institute of Technology, Madras, \textsuperscript{2}University of Maryland, College Park

**Motivation: Generalizability and Cluster Contrastive**
- Learning audio representation that can generalise across various speech and non-speech tasks in low-resource settings.
- SLICER (Symmetrical Learning of Instance and Cluster level Efficient Representation) computes contrastive loss at the instance and cluster levels to generate clustering-favourite representations.

**Proposed Augmentation Technique: K–mix**
- K–mix aims to sample audio from the queue which is further apart in Euclidean space, using k–means.
- A simple k–means clustering algorithm is trained on a 10% unlabeled AudioSet dataset to obtain k–cluster centroids.
- For a new data sample, the first step is to find the closest centroid and sort the queue in descending order based on the centroid distances.
- The next step is to randomly select a sample from the first r samples as noise.

\[ \bar{x}_i = \log \left( (1 - \lambda) \exp(x_i) + \lambda \exp(x_k) \right) \]

**Results: Comparing SLICER performance**
Models have been pre-trained on 10% of AudioSet and FSD50K and then linearly evaluated while keeping the weights frozen on the LAPE benchmark.

<table>
<thead>
<tr>
<th>Model</th>
<th>Speech Tasks</th>
<th>Non-Speech Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SC-V1</td>
<td>SC-V2 (12)</td>
</tr>
<tr>
<td>COLA</td>
<td>77.3</td>
<td>77.2</td>
</tr>
<tr>
<td>BYOL</td>
<td>87.7</td>
<td>87.2</td>
</tr>
<tr>
<td>DeLoRes-S</td>
<td>86.1</td>
<td>85.4</td>
</tr>
<tr>
<td>DeLoRes-M</td>
<td>94.0</td>
<td>93.3</td>
</tr>
<tr>
<td>SLICER</td>
<td>94.8</td>
<td>94.2</td>
</tr>
</tbody>
</table>

SLICER achieves SOTA performance with an average gain of 1.2% across all the downstream tasks in the LAPE benchmark compared to DeLoRes-M.

**Proposed Architecture for SLICER**
Introduce symmetric cross contrastive learning framework (instance-level contrastive learning) and cluster-level contrastive learning framework for momentum based student-teacher network.

**Instance–level contrastive loss:**
\[
L'(f, h) = - \log \left( \frac{\exp(f(x_i^I) \cdot h(x_k^I)/\tau)}{\sum_{j=0}^{K} \exp(f(x_j^I) \cdot h(x_k^I)/\tau)} \right)
\]

**Cluster–level contrastive loss:**
\[
L^C = - \log \left( \frac{\exp(\gamma_i^C \cdot \gamma_j^C/\tau)}{\sum_{c=0}^{K} \exp(\gamma_i^C \cdot \gamma_j^C/\tau)} \right)
\]

- SLICER is pre-trained by combining instance and cluster–level contrastive loss.
- While the instance-level contrastive loss is computed between the student and teacher network, the cluster-level contrastive loss is computed only with the student network.
- The teacher network parameters are updated using momentum update (exponential average update).