Improved Representation Learning for Acoustic Event Classification using Tree-Structured Ontology

Arman Zharmagambetov, Qingming Tang, Chieh-Chi Kao, Qin Zhang, Ming Sun, Viktor Rozgic, Jasha Droppo and Chao Wang

IEEE ICASSP 2022

amazon

Motivation: real world data have some structure

AudioSet contains sound events from 527 classes organized in a hierarchy. We expect representations of semantically similar audio events to be similar. Can we embed such structural information to obtain better representations?

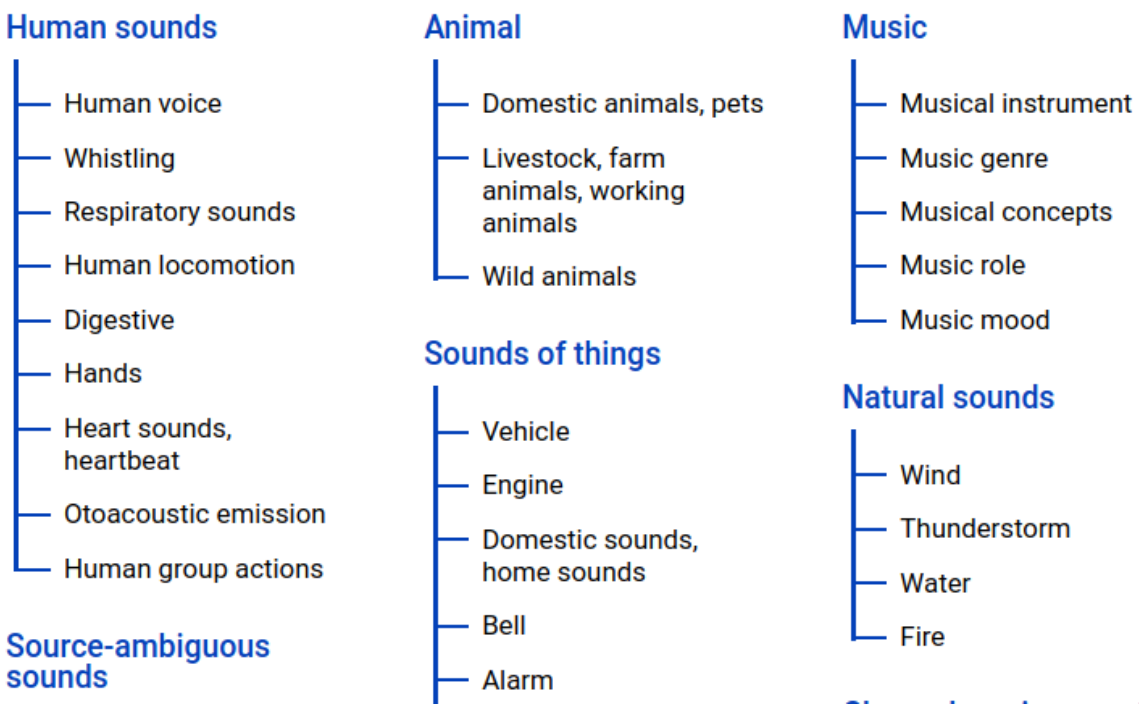
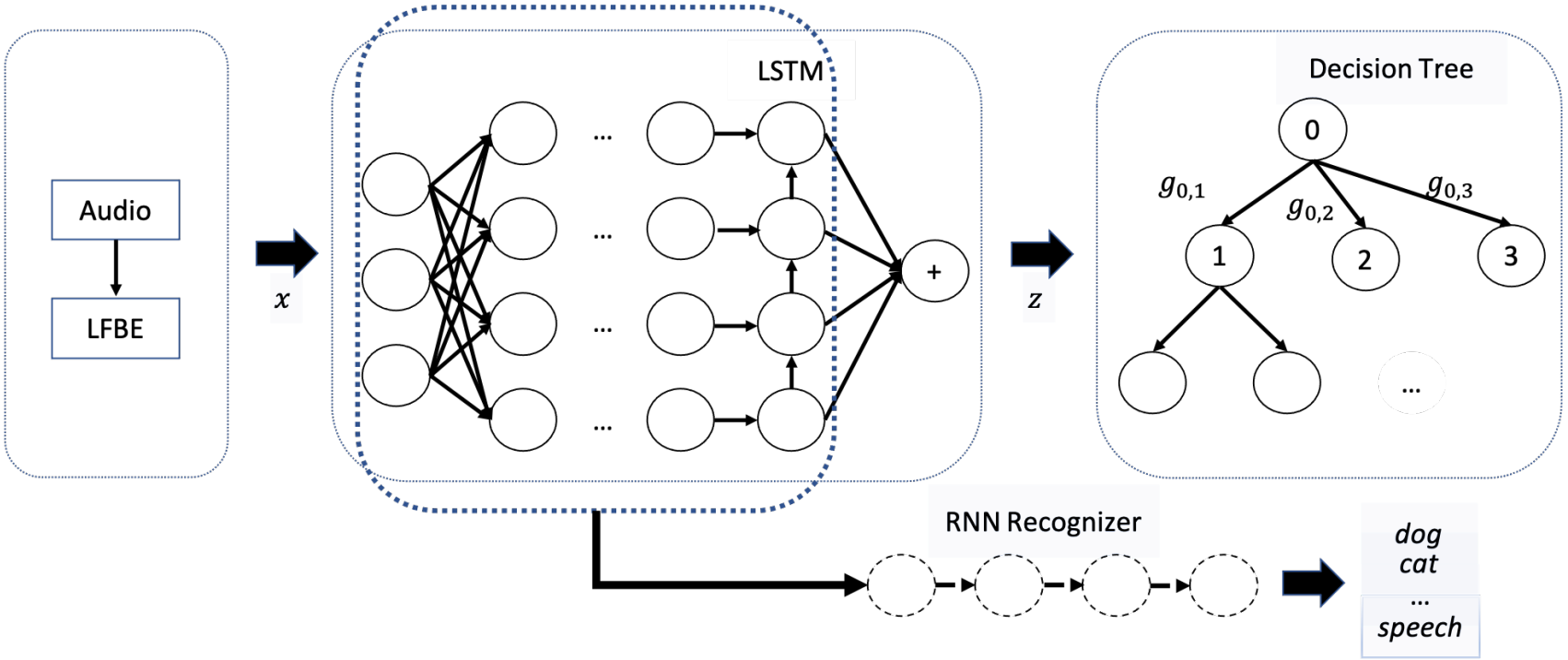


Figure: Ontology of various audio events from AudioSet. Source: `https://research.google.com/audioset/ontology/index.html`

Introduction

- We consider representation neural networks pretrained on a large amount of data (mostly unlabeled).
- Extracted representations can be used by a wide range of downstream tasks, e.g. acoustic event classification, speech recognition, etc.
- We capitalize on the idea of using a tree-structured ontology to guide the training of the representation network.
- Why Tree?
  - Availability of the human curated ontology;
  - Well suited for annotation purposes;
  - Can be applied to detect/categorize novel sounds;
  - Encouraging the audio representation manifold to align with tree ontology organizes representations in more reasonable fashion;
  - etc.

Proposed Approach



Representation network (encoder) takes log mel-filterbank energy (LFBE) $\mathbf{x}$ as input and produces an embedding vector $\mathbf{z}$ (average of the LSTM outputs). Decision tree operates on embedding space and has its own learnable parameters at each node. The encoder + decision tree is trained jointly; for downstream tasks (like RNN recognizer), the encoder is frozen.

Proposed approach

- The transformation of the encoder is denoted as $\mathbf{z} = f(\mathbf{x}; \Theta)$ which is the input to the tree module. For each tree node i we define a gating function $g_i(\cdot)$ with trainable parameters $\mathbf{W}_i, \mathbf{b}_i$:

$$g_i(\mathbf{z}; \mathbf{W}_i, \mathbf{b}_i) = \sigma(\mathbf{W}_i^T \mathbf{z} + \mathbf{b}_i) \tag{1}$$

  The above softmax function outputs probability distribution over the children nodes of i. Then, the probability of reaching a certain leaf:

$$P(y|\mathbf{z}) = g_{\mathbf{c}_0,\mathbf{c}_1}(\mathbf{z}) \cdot g_{\mathbf{c}_1,\mathbf{c}_2}(\mathbf{z}) \cdots g_{\mathbf{c}_{l-1},\mathbf{c}_l}(\mathbf{z}) \tag{2}$$

  where $g_{\mathbf{c}_{i-1},\mathbf{c}_i}$ indicates the probability of transition from node $\mathbf{c}_{i-1}$ to its child $\mathbf{c}_i$ along root-to-leaf path.
- Training: given a set (X, Y), we minimize the negative log-likelihood loss shown below:

$$L_s(\mathbf{W}, \Theta) = - \mathop{\mathbb{E}}_{\mathbf{x},y \in (X,Y)} \frac{\sum_{j=1}^{|y|} \log P(y_j|f(\mathbf{x}))}{|y|} \tag{3}$$

Leveraging unlabeled data

- In order to use unlabeled data, we apply *consistency training*. Assuming we are given a portion of data indexed in set $\{\mathbf{x}_m\}_{m=1}^{M}$, the unsupervised consistency loss is defined as:

$$L_c(\mathbf{W}, \Theta) = \mathop{\mathbb{E}}_{\mathbf{x}} \left\{ \mathop{\mathbb{E}}_{\mathbf{c} \in \mathbf{C}} \left\{ \mathcal{D}(g_c(\mathbf{z}), g_c(\hat{\mathbf{z}})) \right\} \right\} \tag{4}$$

  where $\mathbf{z}$ and $\hat{\mathbf{z}}$ are the representations of $\mathbf{x}$ and $\hat{\mathbf{x}}$ (perturbed duplicate of $\mathbf{x}$), respectively. To obtain $\hat{\mathbf{x}}$, we use common augmentation methods, such as SpecAugment which applies transformations on the LFBE surface feature.

- The final loss:

$$L(\mathbf{W}, \Theta) = L_s(\mathbf{W}, \Theta) + \lambda L_c(\mathbf{W}, \Theta) \tag{5}$$

Experiments: main results

Comparison of different methods on a downstream classification task (on a subset of AudioSet) given fixed representations. F1 score is given as the average of 5 runs.

Method	Label fraction 1%	5%	10%
	Test F1 macro		
supervised	——— 0.612 ———		
supervised (with aug.)	——— 0.638 ———		
LFBE	0.201	0.442	0.508
SimCLR+APC	0.244	0.491	0.551
SimCLR+APC → Fine-tune	0.329	0.540	0.587
SDT	0.288	0.531	0.580
SDT + consistency (SDTC)	0.410	0.557	0.600
SDTC + APC (SDTCA)	**0.417**	**0.561**	**0.609**

Experiments: accuracy at the level of super-categories

Average accuracy by levels where "Level 1" checks if a leaf node is correctly classified, parent and grandparent for "Level 2" and "Level 3". We measure the accuracy as follows: consider as an example audio clips of "dog barking", we collect all audio clips which belong to this event. We measure how many of them are correctly classified as "dog" (Level 1), as "domestic animal"(Level 2), etc.

	Seen		Unseen*	
Avg. acc.	SDT	Baseline	SDT	Baseline
Level 1	0.46	0.35	-	-
Level 2	0.61	0.44	0.59	0.41
Level 3	0.79	0.62	0.78	0.57

* here unseen means novel classes that have not been available during training.

Conclusion

- We leverage a tree-structured ontology of audio events for representation learning for acoustic event classification.
- To achieve this goal, we propose a parametric tree model which can be jointly trained with a representation encoder.
- We apply a semi-supervised learning scheme based on consistency training that can be used to handle labeled data scarcity issue.
- To the best of our knowledge, this is the first attempt to use consistency training for tree-based models in the AEC domain.
- Experimental results suggest that:
  - SDT-based semi-superivsed learning can improve AEC performance by conveying the structural information hidden in the label ontology to learned audio embeddings.
  - The proposed approach allows to more confidently classify unseen/novel events (at the level of super-categories).

Questions?

Poster Session: AUD-9: Detection and Classification of Acoustic Scenes and Events III: Losses and Training

When? Monday, 9 May, 21:00 - 21:45 (Singapore Time, UTC +8)