Ontology-aware Learning and Evaluation for Audio Tagging

Haohe Liu[1], *Qiuqiang Kong*[2], *Xubo Liu*[1], *Xinhao Mei*[1], *Wenwu Wang*[1], *Mark D. Plumbley*[1]

[1]Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK
[2]Speech, Audio, and Music Intelligence (SAMI) Group, ByteDance, China

haohe.liu@surrey.ac.uk

Abstract

This study defines a new evaluation metric for audio tagging tasks to alleviate the limitation of the mean average precision (mAP) metric. The mAP metric treats different kinds of sound as independent classes without considering their relations. The proposed metric, ontology-aware mean average precision (OmAP), addresses the weaknesses of mAP by utilizing additional ontology during evaluation. Specifically, we reweight the false positive events in the model prediction based on the AudioSet ontology graph distance to the target classes. The OmAP also provides insights into model performance by evaluating different coarse-grained levels in the ontology graph. We conduct a human assessment and show that OmAP is more consistent with human perception than mAP. We also propose an ontology-based loss function (OBCE) that reweights binary cross entropy (BCE) loss based on the ontology distance. Our experiment shows that OBCE can improve both mAP and OmAP metrics on the AudioSet tagging task.

Index Terms: machine learning, audio tagging, ontology, evaluation metric

1. Introduction

Audio tagging is a task that tags an audio clip with one or more labels. Audio tagging has attracted increasing interest from researchers in recent years [1, 2], with the increasing number of papers in the Detection and Classification of Acoustic Scenes and Events (DCASE) data challenges [3, 4, 5]. Audio tagging has several applications such as urban noise control [6], audio retrieval [7], and audio monitoring [8].

Most evaluation metrics for audio tagging systems are based on a confusion matrix [9]. Early works [10, 11] employ metrics such as the equal error rate (EER), sensitivity index [12], and F-score [13]. Many recent studies adopt the mean average precision (mAP) as the evaluation metric for audio tagging [14, 1, 15], which measures the area under the precision-recall curve. The mAP is preferable over other metrics on datasets with unbalanced class distribution [16], such as AudioSet [17].

Recently, a number of large-scale datasets for audio tagging have been proposed, such as AudioSet [17], and FSD50K [18]. There are 527 classes in AudioSet and 200 classes in FSD50K, with an unbalanced distribution of total duration in each class. To address the class imbalance issues [2], the primary evaluation metric on these datasets is the class-wise mAP, in which the average precision (AP) scores are calculated on different classes, and their mean value is the final mAP. When calculating the mAP, if a predicted sound event does not appear in the target labels, the prediction will be considered false positive (FP) [1]. Otherwise, it will be counted as a true positive (TP). However, calculating FP in this way has the following problems:

Missing labels in the dataset: The labels in audio tagging datasets are not always correct and may contain missing labels [19, 20]. For example, more than 50% of the labels for around 30% of the classes in AudioSet are estimated to be incorrect. In the evaluation set of AudioSet, there are 4895 files containing the label *Speech*, while only 2.1% of them have label *Male Speech* or *Female Speech*. In this case, even if the model learned to estimate gender on all files with speech, 97.9% of the gender labels will be considered to be false positive. In fact, compared to the *Speech* class with an average precision of 0.80, the experiment in [2] shows that the average precisions for *Male Speech* and *Female Speech* are only 0.07 and 0.09, respectively.

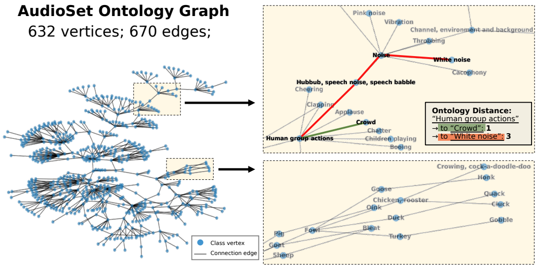


Figure 1: *The ontology graph of the AudioSet [17] classes.*

The non-exclusive nature of sound classes: Sound classes are not always mutually independent. There are also inclusive (e.g., *Music* and *Guitar*) or intersective (e.g., *Shout* and *Yell*) relations between different sounds. Therefore, we believe FPs should be reweighted by their "seriousness". For example, if the target label is the *Giggle* sound, the intuition is that an FP prediction *Laughter* is less "serious" than an FP prediction *Guitar*, because *Giggle* is semantically closer to *Laughter*. Previous evaluation methods fail to consider these class-level relations and may not ideally reflect the model performance.

We therefore propose a metric: ontology-aware mean average precision (OmAP) metric to address the above problems. OmAP reweights the FP predictions based on the ontology node proximity between the prediction and the target labels (see Figure 1). In this way, the FPs can be reweighted based on the relation between the predicted and the target labels. By grouping the classes based on the node proximity, we also show that OmAP can be calculated at different coarse-grained levels, reflecting a more thorough view of the model performance. We conduct human evaluations to study the consistency of different objective metrics to human perceptions and observe higher consistency of the proposed metric than the conventional mAP. Motivated by the ontology-based structure in [21, 22], we also propose a novel ontology-aware binary cross entropy (OBCE) loss function to train audio tagging systems. OBCE reweights the binary cross entropy loss (BCE) based on the class ontology. Our experiments show that the OBCE loss can not only improve the OmAP, but also improve mAP, which further indicates that the ontology information is useful for model optimization.

2. Problem formulation

Audio tagging Let $\mathbb{D}_{\text{train}}$=$\langle \mathbf{X}'_{N\times T}, \mathbf{Y}'_{N\times C}\rangle$ denote an audio tagging training dataset, where $\mathbf{X}'$ and $\mathbf{Y}'$ denote N audio samples and their labels. The audio sample length and the total number of classes are denoted by T and C, respectively. We define class as a particular type of sound and label(s) as the class(es) that appeared in an audio sample. Each audio sample can have one or more labels. The label matrix $\mathbf{Y}'$ only has elements with values zero and one. If the element $\mathbf{Y}'_{n,i}$ is equal one, the n-th sample is labeled with the i-th class. The label of the n-th sample is given by L_n= $\{i \mid \mathbf{Y}'_{n,i} = 1\}$. The audio tagging model $F(\cdot)$ aims to estimate $\hat{\mathbf{Y}}' = F(\mathbf{X}')$, where $\hat{\mathbf{Y}}' \in [0,1]_{N\times C}$ is the estimation of $\mathbf{Y}'$. The performance of the audio tagging model $F(\cdot)$ is calculated on the evaluation dataset $\mathbb{D}_{\text{eval}}$=$\langle\mathbf{X},\mathbf{Y}\rangle$, formulated as $z = \text{Eval}(F(\mathbf{X}), \mathbf{Y})$, where $\text{Eval}(\cdot)$ is an evaluation metric.

Mean average precision Mean average precision (mAP) is a metric that has been widely used in audio tagging [1, 2] and image object detection [23] tasks. z denotes the mAP value. The mAP is the average AP of each class c, and is given by

$$z = \sum_{c=1}^{C} \frac{z_c}{C}, \quad z_c = \mathcal{P}(\mathbf{Y}_{:,c}, \hat{\mathbf{Y}}_{:,c}) = \mathcal{A}(\mathbf{P}_{:,c}, \mathbf{R}_{:,c}), \qquad (1)$$

where z_c is the AP for class c, $\mathcal{P}(\cdot)$ is the function for calculating AP, and $\mathcal{A}(\cdot)$ denotes the function that calculates the area under curve [16]. We use $\mathbf{P}$ and $\mathbf{R}$ to denote the precision and recall matrix. The shape of $\mathbf{P}$ and $\mathbf{R}$ is $N \times C$ because we calculate the precision and recall on N different thresholds and C classes. The N thresholds for a class c are the N values in the label estimation $\hat{\mathbf{Y}}_{:,c}$ [1, 2]. The AP for class c is calculated by the area under the precision-recall curve formed by N pairs of precision and recall coordinates $(\mathbf{P}_{:,c}, \mathbf{R}_{:,c}) = (P_{n,c}, R_{n,c})_{n=1,2,\ldots,N}$. Given a threshold $\gamma = \hat{\mathbf{Y}}_{n,c}$, the coordinates are calculated by

$$(P_{n,c}, R_{n,c}) = \left(\frac{\text{TP}_{n,c}}{\text{TP}_{n,c} + \text{FP}_{n,c}}, \frac{\text{TP}_{n,c}}{\text{TP}_{n,c} + \text{FN}_{n,c}}\right) \qquad (2)$$

where $\text{TP}_{n,c} = |\{i \mid \hat{\mathbf{Y}}_{i,c} > \gamma, \mathbf{Y}_{i,c}=1\}|$, $\text{FP}_{n,c} = |\{i \mid \hat{\mathbf{Y}}_{i,c} > \gamma, \mathbf{Y}_{i,c} = 0\}|$, and $\text{FN}_{n,c} = |\{i \mid \hat{\mathbf{Y}}_{i,c} < \gamma, \mathbf{Y}_{i,c}=1\}|$, respectively, where $|\cdot|$ denotes the size of a set. In Equation (2), the denominator of $R_{n,c}$, $\text{TP}_{n,c} + \text{FN}_{n,c}$, is a constant and equal to the total number of positive labels, $\sum_{i=}^{N} \mathbf{Y}_{i,c}$, for threshold γ and class c.

Audio class ontology The C audio classes can be represented by an undirected complete graph $\mathcal{G} = (V, E)$, where V and E denote sets for vertices and edges, respectively. We use $v_c \in V$ to denote the vertex for class c. We define the node proximity between two vertices v_i and v_j, $\mathbf{D}_{i,j}$, as the smallest number of edges to connect v_i and v_j, given by $\mathbf{D}_{i,j} = \text{Dist}(v_i, v_j)$, where $\text{Dist}(\cdot)$ is the proximity calculation function (e.g., Dijkstra's algorithm). The proximity matrix $\mathbf{D} \in \mathbb{Z}^{+}_{C\times C}$ is symmetric with shape $C \times C$. We also refer to the graph $\mathcal{G}$ as the ontology. One of the most comprehensive audio class ontologies is proposed by AudioSet [17].

3. Ontology-aware mean average precision

As discussed in Section 1, the evaluation of audio tagging system tends to be affected by the missing label problem, and mAP as an evaluation metric does not fully accommodate the relations between classes. Our proposed OmAP addresses these problems by incorporating the ontology graph into the evaluation process. Motivated by [6], we design OmAP to evaluate model performance on multiple coarse-grained levels λ to gain more insights into the model performance. The parameter λ represents the number of nodes or hops in the ontology graph, which covers the area surrounding a class within which all the nodes are treated as the same class. The final OmAP z' is the mean value of ontology-aware average precision (OAP) on different class c and λ, defined by

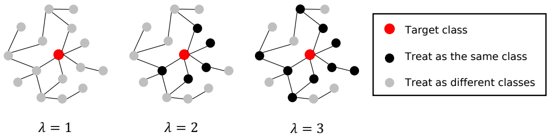


Figure 2: *Calculate false positive on different coarse levels λ on the node highlighted with red. The black nodes will be treated as the same class as the target class without being considered as false positives for a coarse-level evaluation.*

$$z' = \sum_{\lambda=0}^{D_{\text{m}}} \sum_{c=1}^{C} \frac{z'_{\lambda,c}}{D_{\text{m}} C}, \quad z'_{\lambda,c} = \mathcal{P}'(\mathbf{Y}_{:,c}, \hat{\mathbf{Y}}_{:,c}, \lambda, \mathcal{G}) = \mathcal{A}(\mathbf{P}_{:,c}, \mathbf{R}_{:,c}) \qquad (3)$$

where $\mathcal{P}'(\cdot)$ denotes the OAP evaluation function, and D_{m}= $\max(\mathbf{D})$ is the maximum value of proximity between two arbitrary vertices in $\mathcal{G}$, representing the coarsest level of evaluation. We will introduce the detail of multi-level coarse-grained evaluation in Equation (5). In a similar way as Equation (2), for each class c with N thresholds $(\hat{\mathbf{Y}}_{n,c})_{n=1,2,\ldots,N}$, we calculate the N coordinates of the OAP precision-recall curve by

$$(P_{n,c}, R_{n,c}) = \left(\frac{\text{TP}_{n,c}}{\text{TP}_{n,c} + \text{FP}_{n,c}\,\mathbf{W}_{n,c}}, \frac{\text{TP}_{n,c}}{\text{TP}_{n,c} + \text{FN}_{n,c}}\right), \qquad (4)$$

in which the calculation of FN, FP, and TP are the same as Equation (2), and the only difference is the reweight matrix $\mathbf{W}_{n,c}$, which represents how "serious" is the mistake if class c appears as an FP on the n-th samples. The shape of the reweighting matrix $\mathbf{W}$ is $N \times C$. The value of $\mathbf{W}_{n,c}$ will be small if $\text{FP}_{n,c}$ represents only a minor mistake. The seriousness of $\text{FP}_{n,c}$ is quantified with the ontology graph based on the assumption that a label prediction that is further away from the target label is a more "serious" mistake. To calculate $\mathbf{W}$, we first quantify the ontology proximity $\mathbf{D}$ by

$$\mathbf{D}_{i,j} \begin{cases} d_{i,j}, & \text{if } d_{i,j} > \lambda \\ 0, & \text{otherwise} \end{cases}, \quad d_{i,j} = \text{Dist}(v_i, v_j) \qquad (5)$$

As illustrated in Figure 2, OmAP is calculated with multiple coarse-grained levels λ from 0 to D_m, $\lambda \in \mathbb{Z}_{\geq 0}$, where D_{m} is the maximum proximity between two arbitrary vertices in $\mathcal{G}$. Evaluation with different λ can alleviate the missing label problem because the missed labels, which are more likely to be closer to target labels, will be omitted at certain coarse levels. For example, if $\lambda = 2$, the FP on classes that have a minimum proximity smaller or equal than two (e.g., *Female Speech*) to the target classes (e.g., *Speech*) will not be taken into account. With the proximity matrix, we can calculate $\mathbf{W}_{n,c}$ by

$$\mathbf{W}_{n,c} = \frac{1}{\mu} \min\{\mathbf{D}_{c,k} \mid k \in L_n\}, \quad \mu = \text{mean}(\mathbf{D}), \qquad (6)$$

Algorithm 1: Calculate OBCE loss weight

```
Inputs : Ontology $\mathcal{G}$, label for the $n$-th sample $L_n$, total
         number of classes $C$, proximity power factor $\beta$.
Output : Loss weight vector $\boldsymbol{r}$ with length $C$.
1 for $c$ in $[1, 2, ..., C]$ do
2     $\boldsymbol{r}_c = \min\{d^\beta \mid d = \mathrm{Dist}(v_c, v_k), k \in L_n\}$;
3 $\boldsymbol{r} \leftarrow \boldsymbol{r} / \max(\boldsymbol{r})$;                ▷ Preparation for line 4.
4 $\boldsymbol{r}_k \leftarrow 1.0,\ k \in L_n$;   ▷ Target labels have the highest weight.
5 for $c$ in $[1, 2, ..., C]$ do
6     $\boldsymbol{r}_c \leftarrow \boldsymbol{r}_c / \mathrm{mean}(\boldsymbol{r})$;
7     ▷ Let mean $\bar{\boldsymbol{r}}$=1. For a fair comparison with the BCE loss.
```

where L_n is the label for the n-th sample, function $\mathrm{mean}(\cdot)$ calculates the mean value of all the elements in a matrix, and μ is the mean value of $\mathbf{D}$. We divide $\mathbf{W}$ by μ to ensure the value of OmAP can have a similar scale as mAP. The reweighting matrix $\mathbf{W}$ is dependent on $\mathbf{D}$, which is calculated with different λ, thus $\mathbf{W}$ also has different values on different λ. Finally, $\mathbf{W}_{n,c}$ can be utilized in Equation (4) and (3) to calculate OmAP at different coarse-grained levels.

4. Ontology-aware binary cross entropy loss

We propose an OBCE loss, $\mathcal{L}_{\text{obce}}$, to explore if the ontology information is beneficial for model optimization. The intuition behind $\mathcal{L}_{\text{obce}}$ is similar to OmAP, alleviating the missing-label problem, and treating each class differently according to its proximity to the target classes. The proposed OBCE loss is built upon the traditional BCE loss. Given the target and label prediction $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ of an audio sample, the BCE loss can be formulated as

$$\mathcal{L}_{\text{bce}} = \mathrm{mean}(\boldsymbol{y} \odot \log(\hat{\boldsymbol{y}}) + (1 - \boldsymbol{y}) \odot \log(1 - \hat{\boldsymbol{y}})) \quad (7)$$

where $\odot$ means the Hadamard product, log means element-wise log, and $\boldsymbol{y}$ is the label vector with elements of ones and zeros. Compared to $\mathcal{L}_{\text{bce}}$, the OBCE loss reweights the loss function for each class c based on the node proximity of the predictions to the target labels. Based on the similar motivation discussed in Section 3, OBCE loss is designed to assign a smaller weight to false predictions that are closer to the target labels. Assigning weight to false prediction can also alleviate the missing-label problem. As shown in Algorithm 1, we calculate the loss weight of class c, $\boldsymbol{r}_c$, based on the minimum node proximity between the vertex of class c and vertices of the target label set L_n. With the loss weight $\boldsymbol{r}$, the OBCE loss can be formulated as

$$\mathcal{L}_{\text{obce}} = \mathrm{mean}(\boldsymbol{r} \odot (\boldsymbol{y} \odot \log(\hat{\boldsymbol{y}}) + (1 - \boldsymbol{y}) \odot \log(1 - \hat{\boldsymbol{y}}))) \quad (8)$$

Note that in OmAP, we calculate vertex proximity in $\mathcal{G}$ by simply calculating the number of edges (see Equation (5)). However, this assumption is not necessarily optimal for model optimization using OBCE. Hence, in Algorithm 1, we raise the proximity d to the power of a proximity power factor β to explore the effect of non-linear proximity. We empirically observe that β can affect the model performance on mAP and OmAP. Since the OBCE loss has a higher weight on classes that are further away from the target labels, the OBCE loss tends to emphasize more on coarse-grained classification. Therefore, we use the BCE loss with the OBCE loss to ensure the model still is sufficiently optimized for fine-grained classification. The final loss function $\mathcal{L}$ is the combination of the BCE and OBCE losses, given by $\mathcal{L} = (\mathcal{L}_{\text{bce}} + \mathcal{L}_{\text{obce}})/2$, in which the division of these two is to ensure a similar scale of $\mathcal{L}$ with $\mathcal{L}_{\text{bce}}$ for fair comparisons in the experiments.

Model	Params	mAP	OmAP	OmAP$_0$
PANN [1]	42 M	43.3	76.7	54.3
PSLA [2]	14 M	43.7	77.6	55.3
AST [14]	88 M	45.6	**78.5**	57.0
HTS-AT [24]	31 M	**46.4**	**78.5**	**57.7**

Table 1: *The performance of state-of-the-art methods on AudioSet. We also report the OmAP at the finest-grained level when λ=0, denoted by OmAP$_0$. All the metrics are reported in the percentage format.*

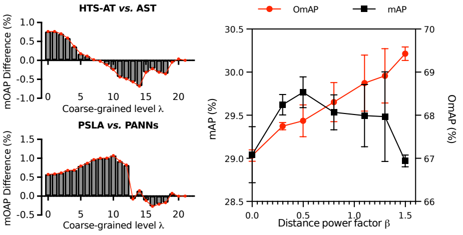
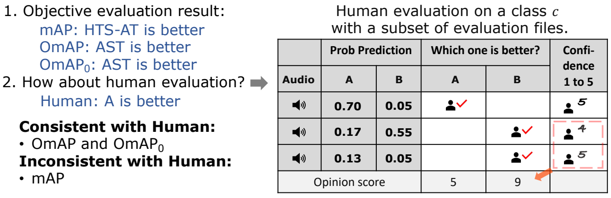


Figure 3: *The left two figures show the differences of OmAP at each coarse-grained level between two groups of models. Although HTS-AT and AST have the same OmAP (see Table 1), HTS-AT is better on smaller coarse-grained levels. The right figure shows the OmAP and mAP performance with different β settings in the OBCE loss.*

5. Experiments

We conduct experiments on the AudioSet balanced subset (AudioSet-20K), full AudioSet (AudioSet-2M) [17], and the FSD50K dataset [18]. All the datasets are resampled into a sampling rate of 16 kHz following [1, 25]. The AudioSet ontology is a complete graph, in which the maximum proximity between two nodes is D_m=21. The FSD50K datasets use part of the AudioSet ontology with 200 classes. We use the same backbone as [2], which is an ImageNet pretrained EfficientNet-B2 [26] with a four-head attention block. The detailed setup of hyper-parameters is the same as Liu et al. [25]. For the proximity power factor β, we use 1.0 by default, except for the experiments in Figure 3. We also report the OmAP when λ = 0, denoted by mAP$_0$, which is the most fine-grained evaluation level. Note that the calculation of OmAP$_0$ is not affected by the value of λ and will consider all false positives made by the model. The false prediction will be reweighted by OmAP$_0$ based on proximity to the target labels.

OmAP of state-of-the-art methods As shown in Table 1, we evaluate several SOTA methods for audio tagging with both mAP and our proposed OmAP metrics. The evaluation is performed on the open-sourced pretrained versions of these four methods. In Table 1 we see that the OmAP score of AST and HTS-AT are the same, while the OmAP of AST when λ=0 is 0.7% lower than HTS-AT, which indicates that HTS-AT and AST do not perform the same on different λ. We further visualize the difference between HTS-AT and AST in Figure 3, which shows that HTS-AT performs better on smaller λ while AST performs better on larger λ. This indicates that the hierarchical structure and shifted window attention in HTS-AT [24] might benefit fine-grained classifications. Although all three evaluation metrics show that PSLA is better than PANN, the comparison between PSLA and PANN in Figure 3 shows that PANN performs better at higher coarse levels, which indicates PANN makes fewer false predictions on classes far from the target classes. The result in this section shows OmAP can provide more detailed evaluation results on different coarse-grained levels, and can better guide



1. Objective evaluation result:
mAP: HTS-AT is better
OmAP: AST is better
OmAP_0: AST is better
2. How about human evaluation?
Human: A is better

Consistent with Human:
- OmAP and OmAP_0

Inconsistent with Human:
- mAP

Human evaluation on a class c with a subset of evaluation files.

	Prob Prediction		Which one is better?		Confi- dence
Audio	A	B	A	B	1 to 5
🔊	0.70	0.05	✓		5
🔊	0.17	0.55		✓	4
🔊	0.13	0.05		✓	5
Opinion score			5	9	

Figure 4: *Comparing the objective evaluation result with the human evaluation score. The procedure in this figure will be performed multiple times with different class c and subsets of evaluation files. Finally, for each objective metric, we calculate the statistic of agreement and disagreement with the human evaluation score to measure its consistency with human perception.*

Evaluation metric	mAP	OmAP	OmAP_0
Consistency with human opinions	10.0%	**82.5%**	62.5%

Table 2: *The percent of agreement between human annotations and each objective evaluation metrics in* 20 *different trails.*

model comparison and performance analysis than mAP.

Which metric is closer to human perception? By randomly sampling a class c and a random subset of evaluation files to evaluate HTS-AT [24] and AST [14], we observe 17% of the results are inconsistent between mAP and OmAP on deciding which model is better. So, we design human evaluations on the inconsistent results as a reference to find out which metric is better. As illustrated in Figure 4, after listening to an audio clip, the participant needs to choose which model makes a better prediction and his/her confidence (1 to 5). We anonymize the file name and model name during the evaluation. In our experiment, we found 94% of the answers are marked with the highest confidence. We perform human annotation on 20 different random classes and subsets of evaluation clips. For each class c we randomly sample 30 audio clips on the AudioSet evaluation subset both with and without the label of class c. We ensure there are at least 5 audio clips with the label c. We also ensure for the clips without label c, at least half of them have model probability estimations greater than 0.1 on class c. We set these two constraints to ensure the majority of 30 audio clips are relevant to class c, and have a reasonable proportion of positive and negative labels. We recruit four participants with audio-processing backgrounds to perform this test. For each of the 20 evaluation subsets, the participants are asked to determine which model is better, model A or B, using the method shown in Figure 4. On a subset of evaluation files, if an objective metric has the same result as the participant, we call this metric consistent with human perceptions. The averaged consistency results on four participants are shown in Table 2. Both OmAP and OmAP_0 show better consistency with human evaluation than mAP.

Improving audio tagging with OBCE loss We performed repeated experiments with different random seeds on both AudioSet-20K and AudioSet-2M with and without the OBCE loss. Our experimental results are shown in Figure 5, in which the experiments with the same seed are connected. We perform a paired t-test [27] on the observed model performance with and without the OBCE loss. The OmAP improvements with OBCE loss on the AudioSet-20K and AudioSet-2M are both statistically significant at more than 99% confidence (p<0.0001 and p=0.0005). This is expected because OmAP and OBCE are designed with similar motivations. Surprisingly, we observe mAP can be improved on both datasets with 95% confidence, which suggests reweighting the loss function based on ontology proximity can also benefit model optimization and help the model make more accurate predictions. This might be because the reweighting in OBCE loss helps the model to learn the relation of classes, such as which classes are more (dis)similar to the target classes, and learn a better decision boundary. We also conduct experiments on FSD50K and perform the same paired t-test. The result shows that the OBCE loss can improve the OmAP on the FSD50K tagging task with 95% confidence ($p < 0.05$), while our experiments do not show high confidence in improving mAP (p=0.73). This might be because the audio clips in FSD50K are more exhaustively labeled than AudioSet [18], hence fewer labels are missing. Nevertheless, the improvements of mAP and OmAP with the OBCE loss presented in Figure 5 indicate that ontology is beneficial for audio tagging, which also suggests the proposed OmAP metric is preferable.

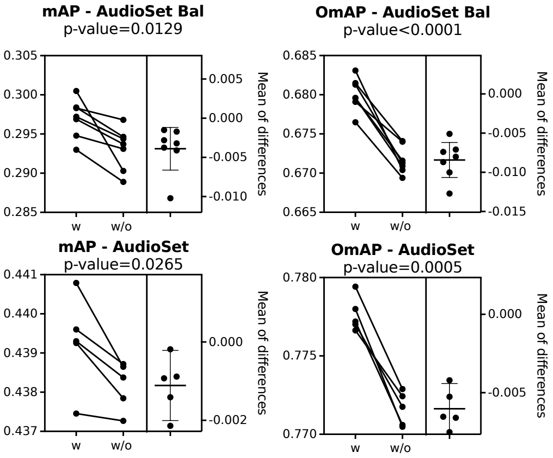


Figure 5: *The mAP and OmAP without (w/o) and with (w) the OBCE loss. Both mAP and OmAP on AudioSet show improvement with more than* 95% *confidence. The right half of each subfigure shows the improvements in the repeated experiments.*

As discussed in Section 4, we introduce a proximity power factor β in the OBCE loss to explore the effect of non-linear proximity between nodes. The parameter β raises the elements of the proximity matrix to power and will affect model optimization. For example, a higher β will make the difference between small and large values more prominent, which in turn makes the classes further from the target classes on the ontology have larger loss weight. When $\beta = 0$, the proximity matrix becomes an all-one matrix, and the OBCE loss is reduced to the conventional BCE loss. The effect of β on the AudioSet-20K is shown in Figure 3. The OmAP improves roughly linearly with the increase of β, while mAP shows roughly a quadratic relation with β. This indicates the OBCE loss might need a proper tuning of β to achieve the best performance on OmAP and mAP.

6. Conclusions

In this paper, we proposed a new evaluation metric, ontology-aware mean average precision (OmAP), which can evaluate model performance based on an intuitive class ontology. The multi-level coarse-grained evaluation scheme in OmAP provides more angles on model evaluation. Our human evaluation shows that OmAP is more consistent with human perceptions. We also proposed a loss function, ontology-aware binary cross entropy (OBCE) loss, that shows high confidence in improving both mAP and OmAP on AudioSet. The success of our proposed OBCE loss also supports our claim that OmAP is preferable to mAP as the audio tagging evaluation metric. Future work will be evaluating the OBCE loss on more SOTA models.

7. References

[1] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[2] Y. Gong, Y.-A. Chung, and J. Glass, "PSLA: Improving audio tagging with pretraining, sampling, labeling, and aggregation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3292–3306, 2021.

[3] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 379–393, 2017.

[4] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "DCASE 2017 challenge setup: Tasks, datasets and baseline system," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.

[5] H. Liu, X. Liu, X. Mei, Q. Kong, W. Wang, and M. D. Plumbley, "Segment-level metric learning for few-shot bioacoustic event detection," *arXiv preprint:2207.07773*, 2022.

[6] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz, and H. Doraiswamy, "SONYC: A system for monitoring, analyzing, and mitigating urban noise pollution," *Communications of the ACM*, vol. 62, no. 2, pp. 68–77, 2019.

[7] G. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 209–215, 2003.

[8] S. Ward and R. Dawes, "AudioWatch: Live audio monitoring for Autumnwatch 2021," 2021. [Online]. Available: https://www.bbc.co.uk/rd/blog/2021-11-live-audio-monitoring-autumnwatch-ai

[9] P. Branco, L. Torgo, and R. P. Ribeiro, "A survey of predictive modeling on imbalanced domains," *ACM Computing Surveys*, vol. 49, no. 2, pp. 1–50, 2016.

[10] E. Cakır, T. Heittola, and T. Virtanen, "Domestic audio tagging with convolutional neural networks," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, 2016.

[11] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "A joint detection-classification model for audio tagging of weakly labelled data," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2017, pp. 641–645.

[12] T. D. Wickens, *Elementary Signal Detection Theory*. Oxford University Press, 2001.

[13] D. C. Blair, "Information retrieval," *Journal of the American Society for Information Science*, vol. 30, no. 6, pp. 374–375, 1979.

[14] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio spectrogram transformer," *arXiv preprint:2104.01778*, 2021.

[15] X. Liu, H. Liu, Q. Kong, X. Mei, M. D. Plumbley, and W. Wang, "Simple pooling front-ends for efficient audio classification," *arXiv preprint:2210.00943*, 2022.

[16] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proceedings of the International Conference on Machine Learning*, 2006, pp. 233–240.

[17] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "AudioSet: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2017, pp. 776–780.

[18] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "FSD50K: An open dataset of human-labeled sound events," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2021.

[19] E. Fonseca, S. Hershey, M. Plakal, D. P. Ellis, A. Jansen, and R. C. Moore, "Addressing missing labels in large-scale sound event recognition using a teacher-student framework with loss masking," *IEEE Signal Processing Letters*, vol. 27, pp. 1235–1239, 2020.

[20] S. Hershey, D. P. Ellis, E. Fonseca, A. Jansen, C. Liu, R. C. Moore, and M. Plakal, "The benefit of temporally-strong labels in audio event classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2021, pp. 366–370.

[21] A. Jimenez, B. Elizalde, and B. Raj, "Sound event classification using ontology-based neural networks," in *Proceedings of the Annual Conference on Neural Information Processing Systems*, vol. 9, 2018.

[22] A. Jati, N. Kumar, R. Chen, and P. Georgiou, "Hierarchy-aware loss function on a tree structured label space for audio event detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2019, pp. 6–10.

[23] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.

[24] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2022, pp. 646–650.

[25] H. Liu, X. Liu, Q. Kong, W. Wang, and M. D. Plumbley, "Learning the spectrogram temporal resolution for audio classification," *arXiv preprint:2210.01719*, 2022.

[26] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114.

[27] X. Tan, J. Chen, H. Liu, J. Cong, C. Zhang, Y. Liu, X. Wang, Y. Leng, Y. Yi, L. He *et al.*, "NaturalSpeech: End-to-end text to speech synthesis with human-level quality," *arXiv preprint:2205.04421*, 2022.