

Zero-Shot Audio Classification via Semantic Embeddings

Huang Xie, *Student Member, IEEE*, Tuomas Virtanen, *Senior Member, IEEE*

Abstract—In this paper, we study zero-shot learning in audio classification via semantic embeddings extracted from textual labels and sentence descriptions of sound classes. Our goal is to obtain a classifier that is capable of recognizing audio instances of sound classes that have no available training samples, but only semantic side information. We employ a bilinear compatibility framework to learn an acoustic-semantic projection between intermediate-level representations of audio instances and sound classes, i.e., acoustic embeddings and semantic embeddings. We use VGGish to extract deep acoustic embeddings from audio clips, and pre-trained language models (Word2Vec, GloVe, BERT) to generate either label embeddings from textual labels or sentence embeddings from sentence descriptions of sound classes. Audio classification is performed by a linear compatibility function that measures how compatible an acoustic embedding and a semantic embedding are. We evaluate the proposed method on a small balanced dataset ESC-50 and a large-scale unbalanced audio subset of AudioSet. The experimental results show that classification performance is significantly improved by involving sound classes that are semantically close to the test classes in training. Meanwhile, we demonstrate that both label embeddings and sentence embeddings are useful for zero-shot learning. Classification performance is improved by concatenating label/sentence embeddings generated with different language models. With their hybrid concatenations, the results are improved further.

Index Terms—audio classification, semantic embedding, zero-shot learning.

I. INTRODUCTION

ZERO-SHOT learning (ZSL), which was first coined in [1], refers to problems that aim at recognizing instances of classes that have no available training samples but only class side information (e.g., textual descriptions). In contrast to supervised classification, samples from only predefined classes are used to train a classifier while new classes get involved at testing or usage stage. In this paper, these new classes are referred to as the zero-shot classes.

Supervised learning has been well-studied for audio classification for decades. To obtain classifiers with satisfactory performance, conventional supervised learning methods require large amounts of annotated training samples from target sound classes. Due to the high cost of data collection and manual

annotation, most of the existing audio datasets [2] have limited numbers of audio samples and sound classes, such as ESC-50 [3] (2,000 labeled audio clips from 50 environmental sound classes) and UrbanSound8K [4] (8,732 labeled audio clips from 10 urban sound classes). In recent years, several research works have been conducted to construct large audio datasets with increased number of sound classes, including AudioSet [5] (over 2 million weakly labeled audio clips covering 527 sound classes), Freesound Dataset [6] (over 290,000 labeled audio clips covering 632 sound classes). However, with the increasing number of observed sound classes, it becomes even more challenging for humans to manually collect sufficient annotated samples for all possible sound classes. To tackle the lack of adequate training data, recent works [7]–[10] in the audio literature mainly apply data augmentation [11], meta learning [12] and few-shot learning [13] methods. However, a considerable amount of representative training samples from target classes is still indispensable to make these methods work. Furthermore, to classify audio instances from new sound classes, supervised learning classifiers require retraining and exhaustive parameter tuning for the new sound classes, which can be time-consuming. With the emergence of zero-shot learning techniques, there is the potential to develop audio classifiers for new sound classes with existing audio datasets [1], [3]–[5].

In recent years, zero-shot learning has received increasing attention in computer vision. The key idea of zero-shot learning is to transfer knowledge from training classes to zero-shot classes. Due to the lack of training samples from zero-shot classes, side information (e.g., textual descriptions) is required for exploring relationship between training and zero-shot classes to make zero-shot learning possible. Fu et al. [14] reviewed the commonly used side information of visual classes for zero-shot recognition by categorizing them into two sets: semantic attributes and beyond. Semantic attributes refer to the intrinsic properties of classes, such as human-defined attributes [15], [16]. Side information beyond semantic attributes includes concept ontology [17], and textual descriptions [18], [19], etc. Based on this side information, both training and zero-shot classes can be projected into the same representation space, which enables knowledge transfer between training and zero-shot classes. Xian et al. [20] reviewed three main kinds of methods for zero-shot image recognition. A simple method [15] was to learn individual classifiers for independent visual attributes and perform zero-shot classification by combining the predicted attributes of the learned individual classifiers. Some sophisticated methods [18], [19] were developed by leveraging intermediate-level representations for

Manuscript created July 16, 2020; revised November 19, 2020 and February 11, 2020. This work was supported by grants from the European Research Council under the European Union’s H2020 Framework Program through ERC Grant Agreement 637422 EVERYSOUND. (*Corresponding author: Huang Xie*).

The results of this manuscript have been partly presented in the MSc. thesis *Zero-Shot Learning for Audio Classification* of Huang Xie (accepted in November 2020), but previously not published in a peer-reviewed publication.

H. Xie and T. Virtanen are with the Faculty of Information Technology and Communication Sciences, Tampere University, Tampere, 33720, Finland (e-mail: huang.xie@tuni.fi; tuomas.virtanen@tuni.fi).

images and classes. In these methods, a linear/non-linear compatibility function was learned to associate images with classes through their intermediate-level representations. For instance, Akata et al. [18] proposed a bilinear compatibility framework to associate images with classes by learning a linear compatibility function. Xian et al. [19] developed this method to explore non-linear compatibilities between images and classes. The other methods [21], [22], mentioned as hybrid models, fall between independent attribute classifiers and compatibility learning frameworks.

In contrast to the steady growth of zero-shot learning methods in computer vision, only limited work has been done in the audio field. Prior works [23]–[25] studied compatibility functions for zero-shot learning in audio classification. They mapped audio signals into a low-dimensional acoustic space through an acoustic embedding learning module, such as siamese network [26] or VGGish [27]. Sound classes were represented by word embeddings in a semantic space, which were learned from their semantic side information (e.g., textual labels) with pre-trained language embedding models, such as Word2Vec [28]. Then, a compatibility learning module was used to associate acoustic embeddings with semantic embeddings. For instance, Islam et al. [23] employed a two-layer fully-connected neural network to model a nonlinear compatibility function. In our previous work [25], we adapted the bilinear compatibility framework [18] from computer vision for zero-shot learning in audio classification. Compared with [23], [25], Choi et al. [24] integrated the acoustic embedding learning module into the compatibility learning module to optimize them holistically. Thus, a nonlinear compatibility function was inherently built into their method.

The previous studies [23], [25] used small audio datasets with 2–10 zero-shot classes for evaluation, which would lead to a limited evaluation of zero-shot learning in audio classification. In contrast, Choi et al. [24] studied zero-shot learning for music classification and tagging with large music datasets. As a complement, it would be necessary to investigate zero-shot learning in large-scale general audio classification. Secondly, only textual labels consisting of 1–3 words were used as the semantic side information of sound classes in [23]–[25]. Compared with textual labels, long pieces of textual descriptions (e.g., sentence descriptions and documents) would convey richer semantic information, which could be useful for improving zero-shot learning performance. Thirdly, a pre-trained acoustic embedding model was used for generating acoustic embeddings of audio instances in [25]. Since a pre-trained acoustic embedding model could have already embedded acoustic information about zero-shot classes, using it to generate acoustic embeddings at training stage would lead to a biased evaluation of zero-shot learning.

In this paper, the present study is extended from our previous work [25]. The main contributions of this work are the following:

- We evaluate zero-shot learning in-depth on both a small environmental dataset ESC-50 [3] and a large general dataset AudioSet [5].
- We consider textual labels and additional sentence descriptions as the side information of sound classes for

zero-shot learning in audio classification. Three kinds of semantic representations of sound classes are studied: label embeddings, sentence embeddings, and their concatenations.

- We investigate generating semantic embeddings with different pre-trained language models: Word2Vec [28], GloVe [29], and BERT [30].

The remainder of this paper is organized as follows. Section II presents an overview of the proposed method for zero-shot learning in audio classification. Section III introduces the implementation details of generating acoustic embeddings and semantic embeddings. Section IV describes the bilinear compatibility learning framework used for associating acoustic embeddings with semantic embeddings. Section V describes the evaluation experiments, and Section VI discusses the results. Section VII concludes this paper.

II. PROPOSED METHOD

In this section, we present an overview of the proposed method for zero-shot learning in audio classification, as illustrated in Fig.1. Sound classes are represented by semantic embeddings, which are learned from their semantic side information with a language embedding module. Acoustic embeddings are extracted from audio instances through an acoustic embedding module, and then projected onto semantic embeddings through an acoustic-semantic projection. A compatibility function is defined to measure how similar/compatible an projected acoustic embedding and a semantic embedding are. During training, the acoustic-semantic projection is optimized with training data. For prediction, an audio instance is classified into a sound class, the semantic embedding of which has the maximal compatibility with its projected acoustic embedding.

We denote by X an audio sample space, Y a set of training classes, and Z a set of zero-shot classes. Note that $Y \cap Z = \emptyset$. We define t_y, t_z as the semantic side information of sound classes $y \in Y$ and $z \in Z$, respectively. We are given a set of training samples $S_{tr} = \{(x_n, y_n) \in X \times Y | n = 1, \dots, N\}$, where x_n is an annotated audio sample belonging to a training class y_n . Our goal is to obtain an audio classifier $f : X \rightarrow Z$ that can recognize zero-shot classes Z , which have no available training samples but only semantic side information. In the proposed method, zero-shot learning is done by leveraging intermediate-level representations of audio instances and sound classes, i.e., acoustic embeddings and semantic embeddings. We denote by $\theta(x) \in \mathbb{R}^{d_a}$ the acoustic embedding of an audio instance $x \in X$. We denote by $\phi(y) \in \mathbb{R}^{d_s}$, $\phi(z) \in \mathbb{R}^{d_s}$ the semantic embedding extracted from semantic side information t_y and t_z , respectively. With these embeddings, an acoustic-semantic projection $T : \mathbb{R}^{d_a} \rightarrow \mathbb{R}^{d_s}$ is exploited to associate audio instances with sound classes.

Given an audio instance $x \in X$ belonging to a sound class $z \in Z$, we assume that the projected acoustic embedding $T(\theta(x))$ in the semantic embedding space is more similar to the semantic embedding $\phi(z)$ rather than those of other sound classes. A similarity scoring function $F : \mathbb{R}^{d_s} \times \mathbb{R}^{d_s} \rightarrow \mathbb{R}$, as known as the compatibility function, is then defined to measure how similar/compatible an projected acoustic embedding and

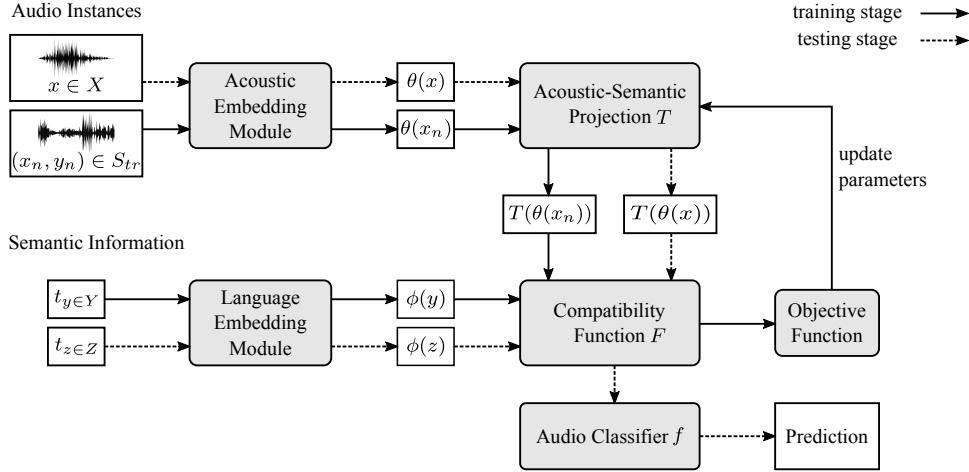


Fig. 1. Zero-shot learning via semantic information in audio classification: sound classes $y \in Y$ and $z \in Z$ are represented by semantic embeddings extracted from their respective semantic side information t_y and t_z . 1) At training stage (solid line), an acoustic-semantic projection T is optimized with every audio sample $(x_n, y_n) \in S_{tr}$. 2) At testing stage (dashed line), an audio classifier f will classify an audio instance $x \in X$ into a sound class $z \in Z$ that has the maximal compatibility value $F(T(\theta(x)), \phi(z))$.

a semantic embedding are. Possible choices of F could be Euclidean distance [23], cosine similarity [24], etc. Therefore, the audio classifier $f : X \rightarrow Z$ is formulated as¹

$$f(x) = \underset{z \in Z}{\operatorname{argmax}} F(T(\theta(x)), \phi(z)). \quad (1)$$

At training stage, f is trained with every audio sample $(x_n, y_n) \in S_{tr}$, and is expected to generalize to new classes Z as in (1) at testing stage.

III. INTERMEDIATE-LEVEL REPRESENTATIONS

In this section, we describe our approaches of generating intermediate-level representations of audio instances and sound classes, i.e., acoustic embeddings and semantic embeddings.

A. Semantic Embeddings

In this work, we use three pre-trained language embedding models to extract semantic embeddings from semantic side information of sound classes. Specifically, we adopt Word2Vec [28] and GloVe [29] as word embedding models to tackle textual labels, and we employ BERT [30] as a sentence embedding model to process sentence descriptions.

Word2Vec (Skip-gram model) is a two-layer fully-connected neural network, which is capable of learning word embeddings that are useful for predicting surrounding words in a sentence or a document [28], [31]. For the sake of simplicity, we use a publicly available pre-trained Word2Vec [32], which is trained on Google News dataset. It consists of 300-dimensional word embeddings for 3 million case-sensitive English words and phrases. To represent sound classes, we take the average of word embeddings of individual words/phrases contained in their textual labels.

GloVe is a statistical language model, which learns word embeddings based on the statistical information of global

word-word co-occurrence and local context of words [29]. We adopt a publicly available pre-trained GloVe [33], which is trained on documents from Common Crawl dataset. It contains 300-dimensional word embeddings for roughly 2.2 million case-sensitive English words. Similarly, we average GloVe word embeddings to obtain semantic embeddings for sound classes.

BERT [30], which stands for Bidirectional Encoder Representations from Transformers, is a contextual language understanding model that is capable of learning deep semantic embeddings of contiguous texts, such as sentences and documents. It obtains state-of-the-art results on a broad set of natural language processing tasks [30]. In this work, we use a pre-trained BERT model [34] to produce semantic embeddings of sound classes from their sentence descriptions. The pre-trained model consists of 24 layers, 1024 hidden states and 16 heads, and it is trained on a large lowercased text corpus (Wikipedia + BookCorpus). At usage stage, it outputs a 1024-dimensional embedding for a sentence description.

B. Acoustic Embeddings

In this work, we use VGGish [27] to generate acoustic embeddings from audio clips. VGGish [27] is a convolutional neural network, which is derived from an VGGNet with Configuration A [35] by adopting the following changes:

- The input size is changed to 96×64 for audio log mel spectrograms.
- The last group of convolutional and max-pool layers is dropped.
- The last fully connected layer is replaced with a 128-wide fully connected layer.

The authors in [27] used VGGish to perform feature extraction for large-scale audio classification tasks and defined the output embeddings of the last 128-wide fully connected layer as acoustic features. It was shown that audio classification performance was improved by using these acoustic embeddings

¹Note that the argmin operation should be considered for other compatibility functions, such as Euclidean distance, etc.

instead of traditional hand-crafted features (e.g., log mel spectrogram) [27].

In zero-shot learning, it is assumed that there is no available acoustic data from zero-shot classes at training stage. In [25], we extracted acoustic embeddings from audio clips with a pre-trained VGGish. We realize that the pre-trained VGGish may have already embedded acoustic information about zero-shot classes and using it to generate acoustic embeddings at training stage can lead to a biased evaluation of zero-shot learning. In this work, we train VGGish from scratch and ensure that test classes are excluded from training data.

To obtain acoustic embeddings with VGGish, we follow the processing in [27]. At first, all the audio clips are resampled to 16 kHz mono. Each mono clip is split into non-overlapping segments of 960 ms. For each segment, short-time Fourier transform is then computed on 25 ms Hanning windowed frames with a step size of 10 ms. After that, the power spectrogram is aggregated into 64 mel bands. Finally, a 96×64 log mel spectrogram is generated and fed into VGGish to obtain a 128-dimensional embedding vector for every audio segment. To generate a clip-level acoustic embedding for an audio clip, we simply average the 128-dimensional embedding vectors of its 960 ms segments.

IV. BILINEAR COMPATIBILITY LEARNING FRAMEWORK

In this section, we describe the bilinear compatibility learning framework adapted from computer vision [18] to tackle zero-shot learning in audio classification in this work.

A. Bilinear Compatibility

As illustrated in Fig.1, acoustic embeddings are extracted from audio instances through an acoustic embedding module. Similarly, sound classes are represented by semantic embeddings learned from their semantic side information with a language embedding module. After that, acoustic embeddings are projected onto semantic embeddings through a simple linear projection T :

$$T(\theta(x)) = W^T \theta(x), \quad (2)$$

where W is the projection matrix to be learned. For compatibility function F , a natural parameterization is the dot product between $T(\theta(x))$ and $\phi(z)$. Therefore, F is written as

$$F(T(\theta(x)), \phi(z)) = T(\theta(x))^T \phi(z). \quad (3)$$

By substituting (2) into (3), F is reformulated as

$$F(T(\theta(x)), \phi(z)) = \theta(x)^T W \phi(z), \quad (4)$$

which is bilinear with respect to acoustic/semantic embeddings. For zero-shot learning, the audio classifier $f : X \rightarrow Z$ is then defined as

$$f(x) = \operatorname{argmax}_{z \in Z} \theta(x)^T W \phi(z). \quad (5)$$

B. Training Algorithm

In this section, we introduce the algorithm for optimizing W in (2) with training data S_{tr} . Given an audio sample $(x_n, y_n) \in S_{tr}$, we consider the task of sorting sound classes $y \in Y$ in descending order according to their compatibility values $F(T(\theta(x_n)), \phi(y))$. Our objective is to optimize W so that y_n will be ranked at top of the sorted class list, i.e., having the maximal compatibility value for x_n .

Let r_{y_n} be the rank of sound class y_n , which refers to the number of incorrect classes placed before y_n . Usunier et al. [36] proposed a ranking error function that transformed rank r_{y_n} into loss $\beta(r_{y_n})$:

$$\beta(r_{y_n}) = \sum_{i=1}^{r_{y_n}} \alpha_i, \quad (6)$$

with $\alpha_1 \geq \alpha_2 \geq \dots \geq 0$ and $\beta(0) = 0$. Specifically, $\beta(r_{y_n})$ defines a ranking penalty for y_n , and α_i measures the penalty incurred by losing a rank from $i-1$ to i . In this work, we follow previous work in [18], [36] and choose $\alpha_i = 1/i$.

Inspired by [36], Weston et al. [37] introduced the hinge loss l into (6) to add a margin and make it continuous, and then proposed the so-called weighted approximate-rank pairwise loss function

$$\frac{1}{N} \sum_{n=1}^N \frac{\beta(r_{y_n})}{r_{y_n}} \sum_{y \in Y} \max\{0, l(x_n, y_n, y)\}, \quad (7)$$

with the convention $0/0 = 0$ when $r_{y_n} = 0$, i.e., y_n was top-ranked. We define the hinge loss $l(x_n, y_n, y)$ as

$$l(x_n, y_n, y) = \Delta(y_n, y) + F(T(\theta(x_n)), \phi(y)) - F(T(\theta(x_n)), \phi(y_n)), \quad (8)$$

where $\Delta(y_n, y) = 0$ if $y_n = y$ and 1 otherwise.

The loss function (7) is convex and can be optimized through stochastic gradient descent. By minimizing (7), the correct class of an audio sample will be top-ranked, i.e., having the maximal compatibility. To prevent over-fitting, we regularize (7) with the squared Frobenius norm of W . The final objective function is

$$\frac{1}{N} \sum_{n=1}^N \frac{\beta(r_{y_n})}{r_{y_n}} \sum_{y \in Y} \max\{0, l(x_n, y_n, y)\} + \lambda \|W\|^2, \quad (9)$$

where λ is the coefficient of the regularization term. In this work, we select λ from $\{0, 0.01, 1, 10\}$ with a validation dataset at training stage.

V. EXPERIMENTS

In this section, we describe the evaluation experiments on two audio datasets: ESC-50 [3] and AudioSet [5].

A. Datasets

ESC-50 [3] is a small balanced audio dataset, which includes 2,000 single-label 5-second audio clips covering 50 environmental sound classes, as shown in Table I. These sound classes are arranged into five high-level sound categories with 10 classes per category: animal sounds, natural soundscapes &

TABLE I
SOUND CATEGORY GROUPS IN ESC-50

Sound Category	Sound Classes
Animal sounds	dog, rooster, pig, cow, frog, cat, hen, insects, sheep, crow
Natural sounds	rain, sea waves, crackling fire, crickets, chirping birds, water drops, wind, pouring water, toilet flush, thunderstorm
Human sounds	crying baby, sneezing, clapping, breathing, coughing, footsteps, laughing, brushing teeth, snoring, drinking sipping
Interior/domestic sounds	door wood knock, mouse click, keyboard typing, door wood creaks, can opening, washing machine, vacuum cleaner, clock alarm, clock tick, glass breaking
Exterior/urban noises	helicopter, chainsaw, siren, car horn, engine, train, church bells, airplane, fireworks, hand saw

water sounds, human (non-speech) sounds, interior/domestic sounds, and exterior/urban noises. Each class is described using a textual class label, such as “dog”, “door wood knock”.

AudioSet [5] is an unbalanced large general audio dataset, which contains roughly 2 million multi-label audio clips covering over 527 sound classes. For the sake of simplicity, we consider only single-label classification in this work and select single-label audio clips from AudioSet. To make a trade-off between least populated classes (i.e., have no more than 50 audio samples) and most populated classes (i.e., have more than 1,000 audio samples), we randomly under-sample the most populated classes until each of them has at most 1,500 audio samples. The final audio subset extracted from AudioSet contains 112,774 single-label 10-second audio clips and 521 sound classes. Each of these classes is defined by a textual label. Meanwhile, AudioSet [5] provides an additional sentence description for every sound class as an explanation of its meaning and characteristics.

B. Experiments on ESC-50

In this section, we introduce our experimental setups on ESC-50 [3].

1) *Dataset Splits*: We conduct zero-shot learning with 5-fold cross-validation on ESC-50 sound classes. First, we split sound classes into five disjoint class folds with two partition strategies: based on sound categories and randomly. In the category-based strategy, sound classes belonging to the same high-level category are organized as one class fold, which is shown as “Sound Category” in Table I. In the random-based strategy, we group sound classes at random into five class folds, as shown in Table II. For each partition strategy, we apply 5-fold cross-validation. Four class folds (40 sound classes with 1,600 audio samples) are used to train the bilinear compatibility framework (in Section IV). After that, zero-shot learning is conducted on the remaining class fold (10 sound classes with 400 audio samples).

2) *VGGish Training*: We train VGGish from scratch with the same class folds that are used for training the bilinear compatibility framework. In total, there are 1,600 audio samples

TABLE II
RANDOM-BASED CLASS FOLDS IN ESC-50

Class Fold	Sound Classes
Fold0	brushing teeth, church bells, clock tick, cow, drinking sipping, fireworks, helicopter, mouse click, pig, washing machine
Fold1	clapping, crickets, glass breaking, hand saw, keyboard typing, laughing, siren, sneezing, thunderstorm, vacuum cleaner
Fold2	breathing, chainsaw, chirping birds, coughing, door wood creaks, door wood knock, frog, pouring water, rain, train
Fold3	airplane, can opening, crying baby, engine, footsteps, hen, insects, rooster, snoring, toilet flush
Fold4	car horn, cat, clock alarm, crackling fire, crow, dog, sea waves, sheep, water drops, wind

of 40 sound classes from four class folds. Each audio sample is split into five 960 ms segments annotated with its sound class. We obtain 8,000 audio segments, and then randomly divide them into two sets with a class-specific proportion of 80/20: 6,400 segments for training and 1,600 segments for parameter validation. During the 5-fold cross-validation, a separate VGGish is trained from scratch at each step, using Adam [38] optimizer with a learning rate of $1e-4$. Training is terminated by early stopping, i.e., once performance stops increasing on the validation partition.

3) *Class Semantic Embeddings*: There are only textual labels of sound classes available in ESC-50. Hence, we consider zero-shot learning via class label embeddings on ESC-50. With the pre-trained Word2Vec [32] and GloVe [33] (in Section III-A), we obtain three sets of class label embeddings: Word2Vec label embeddings (WLE), GloVe label embeddings (GLE), and their concatenations (WLE+GLE).

C. Experiments on AudioSet

In this section, we introduce our experimental setups on the extracted subset of AudioSet [5].

1) *Dataset Splits*: We split the extracted subset (in Section V-A) into five disjoint class folds, which have similar number of sound classes and audio samples, as shown in Table III. First, all sound classes are arranged into nine class bins according to their number of audio samples. Fig.2 illustrates the number of sound classes and total audio samples in the nine class bins. Then, each class bin is randomly split into five groups of sound classes with equal size. After that, we randomly select one group from each class bin without replacement, and merge the nine selected groups into one class fold.

With the increased number of sound classes and audio samples in the extracted subset, we explore the effect of training VGGish with different class folds on generating acoustic embeddings for zero-shot learning. Table IV presents two data settings for training VGGish from scratch. In Setting 1, we train an VGGish model with class folds “Fold0” and “Fold1”, which are excluded from zero-shot learning. As a comparison, another VGGish model is trained in Setting 2 with class

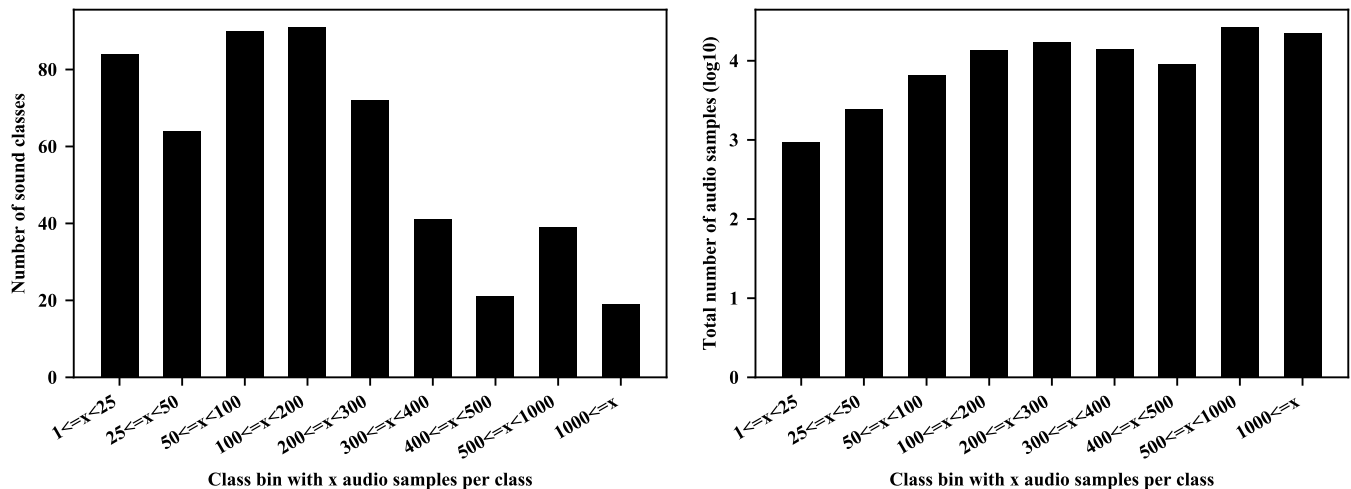


Fig. 2. Number of sound classes and total audio samples of class bins in the extracted subset of AudioSet.

TABLE III
CLASS FOLDS IN THE EXTRACTED SUBSET OF AUDIOSET

Class Fold	Total Sound Classes	Total Audio Samples
Fold0	104	23007
Fold1	104	22889
Fold2	104	22762
Fold3	104	22739
Fold4	105	21377

TABLE IV
TWO DATA SETTINGS WITH THE EXTRACTED SUBSET OF AUDIOSET

Model	Setting 1	Setting 2
VGGish training + validation	Fold0, Fold1	Fold2, Fold3
ZSL training		Fold2
ZSL validation		Fold3
ZSL test		Fold4

TABLE V
SEMANTIC EMBEDDINGS FOR AUDIOSET SOUND CLASSES

Semantic Embedding	Type	Dimensionality
WLE	Label	300
GLE	Label	300
WLE+GLE	Label	600
WSE	Sentence	300
GSE	Sentence	300
BSE	Sentence	1024
WSE+GSE	Sentence	600
WSE+BSE	Sentence	1324
GSE+BSE	Sentence	1324
WSE+GSE+BSE	Sentence	1624
WLE+WSE	Hybrid	600
GLE+GSE	Hybrid	600
WLE+BSE	Hybrid	1324
GLE+BSE	Hybrid	1324
WLE+GLE+BSE	Hybrid	1624

folds “Fold2” and “Fold3”, which are used for training and validating the bilinear compatibility framework. For test, zero-shot learning is conducted on class fold “Fold4”.

2) *VGGish Training*: We train an VGGish from scratch with two class folds in each data setting. Audio samples are randomly divided into training/validation partitions with a class-specific proportion of 75/25. As described in Section III-B, we split audio clips into non-overlapping 960 ms segments and extract a 96×64 log mel spectrogram for each segment. Then, log mel spectrograms belonging to the same audio sample is fed into VGGish as a single input batch. During training, we use Adam optimizer with a learning rate of $1e-4$. We terminate training by early stopping, i.e., once performance stops increasing on the validation partition.

3) *Class Semantic Embeddings*: In addition to textual labels, AudioSet provides an sentence description for every sound class as an explanation of its meaning and characteristics. We consider these sentence descriptions as a complement of the semantic side information of sound classes for zero-

shot learning. Therefore, sound classes have two types of semantic embeddings: label embeddings, which are extracted from their textual labels, and sentence embeddings, which are generated from their sentence descriptions. We obtain three sets of label embeddings (i.e., WLE, GLE and WLE+GLE) by using the pre-trained Word2Vec [32] and GloVe [33] (in Section III-A). For sentence embeddings, we first use the pre-trained BERT [30] to tackle sentence descriptions and obtain a set of BERT sentence embeddings (BSE). As a comparison, we experiment on generating sentence embeddings with the pre-trained Word2Vec [32] and GloVe [33]. An Word2Vec/GloVe sentence embedding (WSE/GSE) is defined as the average of Word2Vec/GloVe word embeddings of individual words in a sentence description. Since stop words usually contain useless semantic information, we exclude them from calculating WSE/GSE. Furthermore, we experiment with concatenating semantic embeddings to obtain concatenated sentence embeddings (e.g., WSE+GSE) and hybrid embeddings (e.g., WLE+BSE) as shown in Table V.

TABLE VI
ZERO-SHOT CLASSIFICATION ON ESC-50 WITH CATEGORY PARTITION STRATEGY

Test Class Fold	WLE		GLE		WLE+GLE	
	mAP	Top-1	mAP	Top-1	mAP	Top-1
Animal sounds	0.43	0.26	0.37	0.18	0.38	0.20
Natural sounds	0.52	0.33	0.42	0.24	0.42	0.24
Human sounds	0.33	0.18	0.35	0.20	0.34	0.18
Interior/domestic sounds	0.37	0.21	0.38	0.21	0.38	0.21
Exterior/urban noises	0.38	0.19	0.40	0.21	0.39	0.21

TABLE VII
ZERO-SHOT CLASSIFICATION ON ESC-50 WITH RANDOM PARTITION STRATEGY

Test Class Fold	WLE		GLE		WLE+GLE	
	mAP	Top-1	mAP	Top-1	mAP	Top-1
Fold0	0.47	0.28	0.45	0.26	0.46	0.26
Fold1	0.56	0.39	0.61	0.46	0.61	0.45
Fold2	0.50	0.33	0.50	0.33	0.52	0.34
Fold3	0.51	0.31	0.50	0.31	0.52	0.33
Fold4	0.50	0.29	0.47	0.26	0.46	0.27

TABLE VIII
AVERAGED PERFORMANCE ON ESC-50 WITH DIFFERENT PARTITION STRATEGIES

Semantic Embedding	Category-based Strategy		Random-based Strategy	
	mAP	Top-1	mAP	Top-1
WLE	0.41	0.23	0.51	0.32
GLE	0.38	0.21	0.51	0.32
WLE+GLE	0.38	0.21	0.52	0.33

D. Evaluation Metrics

For evaluation, we calculate the classification accuracy (Top-1) and mean average precision (mAP) over samples. Top-1 is the proportion of correctly classified samples to the total samples. mAP is the mean over samples of the average precision (AP), which is the average of precisions at all positions where correct classes are placed in a sorted class list [39]. In this work, mAP is inversely proportional to the ranks of corrected classes (in Section IV-B), and perfect classification achieves a mAP of 1.0 when all correct classes are top-ranked.

VI. RESULTS AND ANALYSIS

In this section, we discuss the experimental results of zero-shot learning via semantic embeddings on ESC-50 [3] and AudioSet [5].

A. Zero-Shot Learning on ESC-50

We report the results of the category-based strategy in Table VI, and the ones of the random-based strategy in Table VII, respectively. For 5-fold cross-validation, the averaged results of these strategies are compared in Table VIII.

Overall, we obtain better results than random guess, which should give a mAP / Top-1 of 0.29 / 0.10 in these experiments. The classification performance varies dramatically across different test class folds in both partition strategies regardless of

label embeddings, which indicates that the partition strategy of training classes and test classes has a significant influence on zero-shot learning. Furthermore, we observe that all the three sets of label embeddings have best results on the ‘‘Natural sounds’’ in the category-based strategy and the ‘‘Fold1’’ fold in the random-based strategy. Among these label embeddings, WLE achieves the best mAP / Top-1 of 0.52 / 0.33 on the ‘‘Natural sounds’’ while GLE reaches the best mAP of 0.61 and Top-1 of 0.46 on the ‘‘Fold1’’ fold.

On average, WLE achieves better performance in the category-based strategy with a mAP / Top-1 of 0.41 / 0.23 while WLE+GLE in the random-based strategy reaches a mAP / Top-1 of 0.52 / 0.33. Considering the performance on training data, we obtain an averaged training Top-1 of 0.76 with WLE+GLE, 0.64 with WLE and 0.66 with GLE in the category-based strategy. In the random-based strategy, the averaged training Top-1 is 0.81 with WLE+GLE, 0.76 with WLE and 0.83 with GLE. It indicates that over-fitting occurs with WLE+GLE in the category-based strategy and the partition strategy has an influence on the effectiveness of semantic embeddings.

On the other hand, we notice that classification performance increases significantly in the random-based strategy in contrast to the category-based strategy. In the random-based strategy, sound classes from the same high-level category are distributed across training and test data. We conclude that classification performance can be improved by including classes that are

TABLE IX
ZERO-SHOT CLASSIFICATION ON THE EXTRACTED SUBSET OF AUDIOSET VIA DIFFERENT SEMANTIC EMBEDDINGS WITH TWO DATA SETTINGS

Type	Semantic Embedding	Setting 1		Setting 2	
		mAP	Top-1	mAP	Top-1
Label	WLE	0.15	0.06	0.15	0.06
	GLE	0.14	0.06	0.14	0.06
	WLE+GLE	0.17	0.08	0.19	0.08
Sentence	WSE	0.16	0.07	0.16	0.07
	GSE	0.15	0.06	0.15	0.06
	BSE	0.17	0.08	0.14	0.07
	WSE+GSE	0.16	0.08	0.17	0.09
	WSE+BSE	0.14	0.08	0.17	0.08
	GSE+BSE	0.18	0.08	0.17	0.09
	WSE+GSE+BSE	0.16	0.08	0.18	0.09
Hybrid	WLE+WSE	0.18	0.07	0.18	0.07
	GLE+GSE	0.17	0.07	0.21	0.11
	WLE+BSE	0.21	0.10	0.19	0.10
	GLE+BSE	0.20	0.11	0.20	0.11
	WLE+GLE+BSE	0.20	0.12	0.20	0.11

semantically close to the test classes at training stage. Additionally, for the category-based strategy with WLE, we used a pre-trained VGGish to generate acoustic embeddings in our previous work [25] and obtained an averaged Top-1 of 0.26, which is better than the reported result (0.23) in Table VIII. It would be possible that the pre-trained VGGish already has acoustic information about the test classes, which leads to improved performance.

B. Zero-Shot Learning on AudioSet

The results of different semantic embeddings on two data settings are presented in Table IX. Overall, we obtain better results than random guess, which should give a mAP / Top-1 of 0.05 / 0.01 in these experiments. We discuss the effectiveness of different semantic embeddings in the following paragraphs.

Comparison of label embeddings. We first compare different label embeddings. For WLE and GLE, similar mAPs (around 0.15) and accuracies (about 0.06) are obtained in both data settings. By concatenating WLE and GLE, better results are reached with a mAP / Top-1 of 0.17 / 0.08 in Setting 1 and 0.19 / 0.08 in Setting 2, respectively. It shows that classification performance is improved by concatenating individual label embeddings.

Comparison of sentence embeddings. We experiment with seven sets of sentence embeddings: individual embeddings (i.e., WSE, GSE and BSE), which are generated with individual language models, and their concatenations (i.e., WSE+GSE, WSE+BSE, GSE+BSE, and WSE+GSE+BSE). For the sake of comparing individual embeddings and their concatenations, we average the results across these embeddings. For individual embeddings, an averaged mAP / Top-1 of 0.16 / 0.07 is achieved in Setting 1, and 0.15 / 0.07 in Setting 2. In contrast, their concatenations reach an averaged mAP / Top-1 of 0.16 / 0.08 in Setting 1, and 0.17 / 0.09 in Setting 2. It shows that classification accuracy is improved by concatenating individual sentence embeddings.

Comparison of hybrid embeddings. For comparing hybrid embeddings, we divide them into two groups according to their concatenated sentence embeddings (i.e., WSE/GSE and BSE). For hybrid embeddings with WSE/GSE (i.e., WLE+WSE and GLE+GSE), an averaged mAP / Top-1 of 0.18 / 0.07 is achieved in Setting 1, and 0.20 / 0.09 in Setting 2. On the other hand, we observe that hybrid embeddings with BSE (i.e., WLE+BSE, GLE+BSE and WLE+GLE+BSE) achieve an averaged mAP / Top-1 of 0.20 / 0.11 in both data settings, which is better than those of hybrid embeddings with WSE/GSE.

Similarly to label/sentence embeddings, it shows that concatenating semantic embeddings generated with different language models helps improve classification performance. Since different language models can learn different aspects (in Section III-A) of semantic information from textual labels and sentence descriptions, concatenated embeddings will contain richer semantic information than individual embeddings, which results in an improvement on the performance.

Comparison of label/sentence/hybrid embeddings. First, we compare individual label embeddings (i.e., WLE and GLE) and their hybrid embeddings (i.e., WLE+WSE/WLE+BSE and GLE+GSE/GLE+BSE). The results in Table IX show that classification performance is improved for WLE and GLE by concatenating them with individual sentence embeddings. Secondly, we compare individual sentence embeddings (i.e., WSE, GSE and BSE) and their hybrid embeddings (i.e., WLE+WSE, GLE+GSE, and WLE+BSE/GLE+BSE). We observe that WLE+WSE reaches a better mAP (0.18) than WSE (0.16) but has the same Top-1 (0.07) in both settings. For GSE, classification performance is improved by combining it with GLE (i.e., GLE+GSE) in both settings. Particularly, GLE+GSE achieves a mAP / Top-1 of 0.21 / 0.11 in Setting 2. Similarly, we obtain better results with the hybrid embeddings containing BSE (i.e., WLE+BSE and GLE+BSE) than the individual BSE. Therefore, we conclude that classification performance can be improved by concatenating individual label embeddings

TABLE X
AVERAGED PERFORMANCE ACROSS SEMANTIC EMBEDDINGS ON TWO
DATA SETTINGS OF THE EXTRACTED SUBSET OF AUDIOSET

Type	Setting 1		Setting 2	
	mAP	Top-1	mAP	Top-1
Label	0.15	0.07	0.16	0.07
Sentence	0.16	0.08	0.16	0.08
Hybrid	0.19	0.09	0.20	0.10

TABLE XI
CONTINGENCY TABLE FOR SEMANTIC EMBEDDINGS WLE+GLE+BSE
AND GLE+BSE IN AUDIOSET

Correct / Incorrect Predictions	WLE+GLE+BSE		
	correct	incorrect	
GLE+BSE	correct	1854	381
	incorrect	609	18533

and sentence embeddings.

Furthermore, we study the overall effectiveness of these three types of semantic embeddings by calculating their averaged mAP / Top-1 in Table X. For label embeddings, we obtain an averaged mAP / Top-1 of 0.15 / 0.07 in Setting 1, and 0.16 / 0.07 in Setting 2 across WLE, GLE and WLE+GLE. For the seven sets of sentence embeddings, an averaged mAP / Top-1 of 0.16 / 0.08 is achieved in both data settings. Meanwhile, the hybrid embeddings reach an averaged mAP / Top-1 of 0.19 / 0.09 in Setting 1, and 0.20 / 0.10 in Setting 2. It shows that sentence embeddings have better averaged accuracies than label embeddings while hybrid embeddings have the best performance. A possible interpretation for this would be that sentence descriptions convey richer semantic information than textual labels while their combination includes all their semantic information, which results in an improvement on classification performance.

Comparison of data settings. Table X shows that similar averaged results (i.e., mAP and Top-1) are observed in both data settings for different semantic embeddings. It would be probably due to the sound classes in these AudioSet class folds being comprehensive enough.

McNemar’s test. We apply McNemar’s test [40] to analyze statistical significance of the differences between the results of semantic embeddings. For the sake of simplicity, we compare WLE+GLE+BSE and GLE+BSE, which achieve the best Top-1 and the second best Top-1 in Setting 1. With the contingency Table XI, we obtain a McNemar’s test statistic of 52.05 and a p-value of $5.41e-13$, which shows a significant difference between the results of WLE+GLE+BSE and GLE+BSE.

Summary. The experimental results show that label/sentence embeddings of sound classes, such as WLE/GLE and WSE/GSE/BSE, are useful for zero-shot learning in audio classification. Classification performance can be improved by concatenating individual label/sentence embeddings generated with different language models, such as WLE+GLE and WSE+GSE. With hybrid embeddings (e.g., WLE+BSE, GLE+BSE), which are obtained by concatenating individual label embeddings and sentence embeddings, the results are

improved further.

C. Discussion

So far, zero-shot learning for audio classification has received relatively little attention, for example, in comparison to computer vision research. When formulated as a multi-label classification problem with some feature inputs, these two fields are very similar. On the other hand, there are some factors that make the fields different. In computer vision, zero-shot learning has been studied with even larger datasets (e.g., ImageNet with over 21,000 classes [41]), in comparison to existing works (e.g., 1,126 classes used in [24] and 521 classes in this work) in the audio field. With the fine-grained classes in ImageNet, Xian et al. [20] conducted an in-depth study of zero-shot learning with respect to the semantic similarity, hierarchy and popularity between image classes. Meanwhile, a wider set of semantic information has been explored for zero-shot learning in computer vision. For instance, human-defined visual attributes have been widely used as the semantic information of visual classes [15], [16], [18], [20]. Class hierarchies have also been studied for zero-shot learning in [17], [18]. Another factor is about the intrinsic properties of classes in these two fields, which are also present in supervised classification. For example, in audio, multiple sound sources are often present simultaneously, forming a complex mixture of sounds. Visual objects may be partly occluded, but not usually distorted by other sources. Furthermore, the presence of visual objects in static image can be considered to be binary (present or not), whereas the activities of sound classes are almost always time-varying.

VII. CONCLUSION

In this paper, we present a zero-shot audio classification method via semantic embeddings that are learned from either textual labels or sentence descriptions of sound classes. The experimental results on ESC-50 show that classification performance is significantly improved by involving sound classes that are semantically close to the test classes in training. Furthermore, we explore zero-shot learning on a large-scale audio dataset, which contains 521 sound classes extracted from AudioSet. We demonstrate that both label and sentence embeddings are useful for zero-shot audio classification. Compared with individual label/sentence embeddings, concatenating embeddings generated with language models improves the results.

There is plenty of room to explore for future work. For instance, the pre-trained language models can be fine-tuned with sound-specific text corpora for generating high-quality semantic embeddings of sound classes. To improve label and sentence embeddings, word embeddings can be aggregated using advanced methodologies from natural language processing literature instead of averaging them. We also notice that different semantic embeddings result in learning variable volumes of parameters in the bilinear compatibility function, which would have an effect on classification performance. Future studies on semantic embeddings should take these into account. On the other hand, instead of averaging VGGish-generated embeddings of non-overlapping segments, acoustic

embeddings can be improved by generating them from the entire audio clips with advanced models (e.g., attention models). Meanwhile, training acoustic/language embedding models jointly with the bilinear compatibility learning framework would be beneficial to zero-shot learning. Furthermore, rather than a simple linear acoustic-semantic projection, nonlinear ones should be investigated in the future.

REFERENCES

- [1] M. Palatucci, D. Pomerleau, G. Hinton, and T. M. Mitchell, “Zero-Shot Learning with Semantic Output Codes,” in *Proc. 22nd Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 1410–1418.
- [2] T. Heittola. (2020) Datasets. [Online]. Available: <https://homepages.tuni.fi/toni.heittola/datasets.html>
- [3] K. J. Piczak, “ESC: Dataset for Environmental Sound Classification,” in *Proc. 23rd ACM Int. Conf. Multimedia (MM)*, 2015, pp. 1015–1018.
- [4] J. Salamon, C. Jacoby, and J. P. Bello, “A Dataset and Taxonomy for Urban Sound Research,” in *Proc. 22nd ACM Int. Conf. Multimedia (MM)*, 2014, pp. 1041–1044.
- [5] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio Set: An ontology and human-labeled dataset for audio events,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2017, pp. 776–780.
- [6] E. Fonseca, J. Pons, X. Favory, F. Font, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter, and X. Serra, “Freesound Datasets: A Platform for the Creation of Open Audio Datasets,” in *Proc. 18th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2017, pp. 486–493.
- [7] J. Salamon and J. P. Bello, “Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification,” *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, 2017.
- [8] N. R. Koluguri, M. Kumar, S. H. Kim, C. Lord, and S. Narayanan, “Meta-Learning for Robust Child-Adult Classification from Speech,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2020, pp. 8094–8098.
- [9] B. Shi, M. Sun, K. C. Puvvada, C. C. Kao, S. Matsoukas, and C. Wang, “Few-Shot Acoustic Event Detection via Meta Learning,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2020, pp. 76–80.
- [10] S. Y. Chou, K. H. Cheng, J. S. R. Jang, and Y. H. Yang, “Learning to Match Transient Sound Events Using Attentional Similarity for Few-shot Sound Recognition,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2019, pp. 26–30.
- [11] B. McFee, E. J. Humphrey, and J. P. Bello, “A software framework for musical data augmentation,” in *Proc. 16th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2015, pp. 248–254.
- [12] C. Finn, P. Abbeel, and S. Levine, “Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks,” in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, vol. 70, 2017, pp. 1126–1135.
- [13] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, “Generalizing from a Few Examples: A Survey on Few-Shot Learning,” *ACM Comput. Surv.*, vol. 53, no. 3, Jun. 2020.
- [14] Y. Fu, T. Xiang, Y. G. Jiang, X. Xue, L. Sigal, and S. Gong, “Recent Advances in Zero-Shot Recognition: Toward Data-Efficient Understanding of Visual Content,” *IEEE Signal Process. Mag.*, pp. 112–125, 2018.
- [15] C. H. Lampert, H. Nickisch, and S. Harmeling, “Attribute-Based Classification for Zero-Shot Visual Object Categorization,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 453–465, 2014.
- [16] B. Romera-Paredes and P. H. S. Torr, “An Embarrassingly Simple Approach to Zero-Shot Learning,” in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, vol. 37, 2015, pp. 2152–2161.
- [17] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba, “Semantic Label Sharing for Learning with Many Categories,” in *Proc. 11th Eur. Conf. Comput. Vision (ECCV)*, 2010, pp. 762–775.
- [18] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, “Label-Embedding for Image Classification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1425–1438, 2016.
- [19] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele, “Latent Embeddings for Zero-Shot Classification,” in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, 2016, pp. 69–77.
- [20] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, “Zero-Shot Learning — A Comprehensive Evaluation of the Good, the Bad and the Ugly,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 2251–2265, 2018.
- [21] Z. Zhang and V. Saligrama, “Zero-Shot Learning via Semantic Similarity Embedding,” in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, 2015, pp. 4166–4174.
- [22] S. Changpinyo, W. L. Chao, B. Gong, and F. Sha, “Synthesized Classifiers for Zero-Shot Learning,” in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, 2016, pp. 5327–5336.
- [23] M. T. Islam and S. Nirjon, “SoundSemantics: Exploiting Semantic Knowledge in Text for Embedded Acoustic Event Classification,” in *Proc. 18th Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, 2019, pp. 217–228.
- [24] J. Choi, J. Lee, J. Park, and J. Nam, “Zero-shot Learning for Audio-based Music Classification and Tagging,” in *Proc. 20th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2019, pp. 67–74.
- [25] H. Xie and T. Virtanen, “Zero-Shot Audio Classification Based on Class Label Embeddings,” in *Proc. IEEE Workshop Appl. Signal Process. Audio and Acoustic. (WASPAA)*, 2019, pp. 264–267.
- [26] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese Neural Networks for One-shot Image Recognition,” in *Proc. 32nd Annu. Int. Conf. Mach. Learn. (ICML)*, 2015.
- [27] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, “CNN architectures for large-scale audio classification,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2017, pp. 131–135.
- [28] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in *Proc. 1st Int. Conf. Learn. Representations (ICLR)*, 2013.
- [29] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global Vectors for Word Representation,” in *Proc. 19th Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
- [30] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol. (NAACL-HLT)*, 2019, pp. 4171–4186.
- [31] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and Their Compositionality,” in *Proc. 26th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2013, pp. 3111–3119.
- [32] T. Mikolov, K. Chen, G. Corrado, and J. Dean. (2013) Word2Vec. [Online]. Available: <https://code.google.com/archive/p/word2vec>
- [33] J. Pennington, R. Socher, and C. Manning. (2014) GloVe: Global Vectors for Word Representation. [Online]. Available: <https://nlp.stanford.edu/projects/glove>
- [34] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. (2019) BERT. [Online]. Available: <https://github.com/google-research/bert>
- [35] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *Proc. 3rd Int. Conf. Learn. Representations (ICLR)*, 2015.
- [36] N. Usunier, D. Buffoni, and P. Gallinari, “Ranking with Ordered Weighted Pairwise Classification,” in *Proc. 26th Annu. Int. Conf. Mach. Learn. (ICML)*, 2009, pp. 1057–1064.
- [37] J. Weston, S. Bengio, and N. Usunier, “WSABIE: Scaling Up to Large Vocabulary Image Annotation,” in *Proc. 22nd Int. Joint Conf. Artif. Intell. (IJCAI)*, 2011, pp. 2764–2770.
- [38] D. P. Kingma and J. L. Ba, “Adam: A Method for Stochastic Optimization,” in *Proc. 3rd Int. Conf. Learn. Representations (ICLR)*, 2015.
- [39] C. Buckley and E. M. Voorhees, “Retrieval Evaluation with Incomplete Information,” in *Proc. 27th ACM Annu. Int. Conf. Res. Develop. Inf. Retrieval (SIGIR)*, 2004, pp. 25–32.
- [40] T. Kavzoglu and I. Colkesen, “An assessment of the effectiveness of a rotation forest ensemble for land-use and land-cover mapping,” *Int. J. Remote Sens.*, vol. 34, no. 12, pp. 4224–4241, 2013.
- [41] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, 2009, pp. 248–255.