IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2022, accepted April 17, 2022, date of publication April 21, 2022, date of current version May 2, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3169279

Predicting Classification Accuracy of Unlabeled Datasets Using Multiple Deep Neural Networks

SHINGCHERN D. YOU[1], (Senior Member, IEEE), HSIAO-CHUNG LIU[2], AND CHIEN-HUNG LIU[1]

[1]Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei 10608, Taiwan
[2]Super Micro Computer Inc., Bade, Taoyuan 33463, Taiwan

Corresponding author: Shingchern D. You (scyou@ntut.edu.tw)

This work was supported in part by the Ministry of Science and Technology (MOST), Taiwan, under Grant MOST 109-2221-E-027-083.

ABSTRACT In machine learning problems, we usually assume that the validation accuracy is a good estimation of prediction accuracy for datasets without ground truth. In reality, this assumption may not hold. Therefore, we propose an approach to estimate the prediction accuracy of a target model on unlabeled datasets. The proposed approach uses multiple target homogeneous models to assign each unlabeled sample a confidence value, based on the number of models agreeing on the predicted label. With the confidence values, the prediction accuracy of the target model on the datasets can be estimated. In the experiments, the target model is a convolutional neural network (CNN) model, and the homogeneous models only differ in initial weights. The experiments are conducted with datasets from a wide variety of music genres. The estimation performance of the proposed approach is compared with the reversed testing qualities (RTQ) and the ensemble average qualities (EAQ) approaches. The RTQ approach was proposed to estimate the prediction accuracy of trained models, and the EAQ approach was originally designed for estimating the predictive uncertainty of individual samples. We apply all three compared models to estimate prediction accuracy of datasets by using a linear model. The parameters of the linear model are either computed by using multiple labeled datasets or one labeled dataset. The experimental results show that when compared with the RTQ approach, the proposed approach has much lower estimation errors for some datasets. When compared with the EAQ approach, the proposed approach is more robust for datasets with large distribution shifts. Finally, we show an additional benefit of the proposed approach. In case that the estimated accuracy is unsatisfactory, we may re-train the target model with a new training set, which contains the original training samples plus new training samples with manual labeling from the unlabeled dataset. The experimental results confirm that it is more effective to select (and label) new samples from those with low confidence values than those randomly selected. Overall, the proposed approach is a promising approach for estimating prediction accuracy on unlabeled datasets.

INDEX TERMS Prediction accuracy estimation, unlabeled dataset, machine learning, convolutional neural network, vocal detection.

I. INTRODUCTION

One of the core problems in supervised machine learning is to predict the classification accuracy of a model in real-world applications. Typically, we assume that the labeled training samples and the test samples to be classified later are from the same source, and thus have the same (or at least similar) distribution. Under this assumption, we may partition the labeled dataset into, say, 10-folds and use the accuracy of the cross validation to predict the test accuracy. In many applications, this assumption is reasonable. However, the assumption may not be true if both training and test datasets are from different sources. Unfortunately, unless using special techniques, the model may not be able to know that the test samples are dissimilar to the training samples. Moreover, even if the training dataset and the test dataset are from different sources, the prediction accuracy may be still high. In short, it is really difficult to know if a model has high prediction accuracy on unlabeled datasets.

To illustrate our argument in the previous paragraph, we present our previous studies [1], [2] here. During the studies, we created a dataset from excerpts of soundtracks in the free music archive (FMA) website [3], [4] and used it as an external dataset to test a model trained with the

The associate editor coordinating the review of this manuscript and approving it for publication was Amin Zehtabian.

Jamendo dataset [5] for detecting vocal segments. Note that both Jamendo and FMA datasets contain western popular music. The model trained with Jamendo training set has 94.1% accuracy on Jamendo test dataset, but only 82.7% on FMA test dataset. On the other hand, a model trained with the FMA training set has 92.5% accuracy on Jamendo test and 89.8% on FMA test set [2]. Based on the above observation, we are unable to assert if these two datasets are similar or not. We can only reasonably presume that the FMA dataset could cover more music genres than the Jamendo dataset does. Unfortunately, unless we closely examine the contents of both datasets; otherwise, we are not able to know whether the training and the test datasets cover similar contents. Therefore, it would be very useful if we are able to estimate the prediction accuracy of a trained model.

Although estimating the prediction accuracy of a model is practically very useful, this problem seems to receive less attention. Bhaskaruni *et al.* pointed that they could not find any paper directly related to this topic in 2018 [6]. In their paper, Bhaskaruni *et al.* considered many types of quality metrics. To simplify the discussion, we consider only accuracy in this paper.

In contrast to estimating the prediction accuracy of a model, many existing papers focused on evaluating the predictive uncertainty (or equivalently, confidence) of individual samples based on various approaches, such as deep ensemble networks [7]–[10] or Bayesian networks [11]–[13]. Certainly, these two problems are strongly related to each other. However, they are not the same problem. For example, knowing the prediction accuracy of a model on a new dataset helps to determine whether or not a new model should be trained. If the prediction accuracy is high enough (based on prior knowledge), we may directly use the trained model. If, unfortunately, the prediction accuracy seems low, we can train a new model with the original training set plus some labeled samples from the new test dataset. In this case, knowing the predictive uncertainty of individual samples does not help us make such a decision. Therefore, a procedure is required to convert from the sample uncertainty to the estimated model accuracy. We will show in the experiments that directly averaging the confidence values of all samples does not yield the best estimation.

The contributions of this paper include the following:

- Present a method to estimate the prediction accuracy of a model on unlabeled datasets and compare its performance with the approach proposed by Bhaskaruni *et al.* and the approach based on averaging confidence values of all samples.
- Show an efficient way to re-train the model for a new dataset by giving priority to labeling samples with lower confidence values and including them in the training set for a second-run training.
- Release several labeled datasets for researchers to repeat our experiments and to conduct new experiments easier.

This paper is arranged as follows. Section II describes the related work. Section III discusses the proposed approach and models used in the experiments. Section IV presents the experiments and results, and finally section V is the conclusion.

II. RELATED WORK

Based on the concept of the reverse testing framework [14], Bhaskaruni *et al.* proposed a method to estimate the prediction accuracy of unlabeled datasets [6]. Their approach is outlined here. Suppose that A is a labeled dataset and B is an unlabeled dataset. Let Model One be trained with the training dataset A. Once training is complete, Model One is used to predict the labels of samples in dataset B. The predicted labels are called as pseudo labels. We then use pseudo labels in dataset B to train a new model, called Model Two, and use Model Two to predict the labels of dataset A. Because we have the ground truth of dataset A, we can then compute the accuracy (and other metrics) of Model Two. By assuming that both models have comparable prediction accuracy, we can use the accuracy of Model Two as an estimate of the prediction accuracy of Model One (when predicting unlabeled dataset B). The prediction accuracy of Model Two is called reversed testing qualities (RTQ). The RTQ approach will be the comparison counterpart of our approach in the experiments.

B. Lakshminarayanan *et al.* proposed a deep ensemble approach to estimate the confidence values of test samples based on multiple identical neural networks [7]. To smooth the predictive distributions, they added adversarial samples to the training set. The confidence value is computed by averaging the predicted probabilities of all models. In fact, the proposed approach is also based on multiple identical models. The difference is that our approach is mainly used to estimate the prediction accuracy of a model, not the confidence value of an individual sample. Nevertheless, as this approach is similar to ours, we also include this approach in the experiments as a comparison counterpart.

As there are many different approaches to evaluate predictive uncertainty of test samples, Y. Ovadia, *et al.* conducted experiments to evaluate their relative performance [8]. They concluded that ''Deep ensembles (described in the previous paragraph) seem to perform the best across most metrics and be more robust to dataset shift.''

Vocal detection technique is to detect the presence of vocal signals (singing voice) in a segment of audio work. This technique is a fundamental step for many advanced applications and has been studied for many years. Typically, a vocal detection approach contains a feature extraction step and a feature classification step. The chosen features are usually time-frequency representations, such as MFCC (Mel-scale Frequency Cepstral Coefficients) [15] or spectrogram [1], [2]. Based on our previous experiments, we concluded that spectrogram is a better type of features in this problem. As to the classifier, previously the HMM (hidden Markov model) was widely used [16]. Recently, convolutional neural networks (CNN) have been proven to outperform conventional classifiers [17]. Previously, we showed that the

"spectrogram plus CNN" approach actually outperformed the "end-to-end" approach [2]. With this observation, we choose the "spectrogram plus CNN" approach in the experiments.

III. PROPOSED APPORACH

This section describes the proposed approach. However, before describing our approach, we first describe the used model for accuracy estimation, which is based on the CNN model.

A. SCNN MODEL

We previously studied many models for the vocal classification problem, and finally concluded that a model of "spectrogram plus 18-layer CNN" is better than other configurations [2]. In the rest of the paper, we use this model (denoted as SCNN-18) to conduct experiments. That is, we want to estimate the prediction accuracy of this model on an unlabeled dataset. To do so, we also use multiple such models, to be discussed in subsection III.B.

The used 18-layer CNN model is shown in Fig. 1 [2]. In the figure, each box contains some numbers to indicate the size of the layered feature maps. For example, the second box has "21 × 512 × 64" meaning that the input to this layer has 64 feature maps each with a size of 21 × 512. The input to the CNN model is a 2-second audio clip, consisting of 32,000 PCM samples. The audio clip is multiplied with a series of 2048-coefficient Hamming windows with a hop length of 512. The windowed samples are converted to spectral coefficients by FFT (fast Fourier transformation). The modulus of each spectral coefficient becomes one feature value of the spectrogram, and the spectrogram is the input to the first layer (i.e., the first box with numbers 63 × 1024 × 1) in Fig. 1. The detailed values of the hyperparameters of this CNN are given in Table 1. The activation functions of all layers are ReLU (rectified linear unit), except the output layer which is softmax. A solid box also includes a maxpooling layer, whereas a dashed box does not.

During training, the dropout rate [18] is set to 0.5 and the batch normalization [19] is used. The simulation programs are developed by using the Keras library [20] and the Tensorflow [21] framework. In addition, we use ADADELTA [22] as the optimizer. The training epoch for each trial is 200. Unless otherwise specified, the prediction accuracy of a particular model is computed by an average of 10 trials, such as the values given in Table 2 of subsection IV.A.

B. PROPOSED APPROACH

The proposed approach is based on the concept of majority voting ensemble. To perform majority voting, multiple homogeneous models are trained first. The final prediction output is obtained based on which class receives more votes [2]. In this paper, CNN models with identical structure, but trained with different random initial weights, are referred to as homogeneous models.

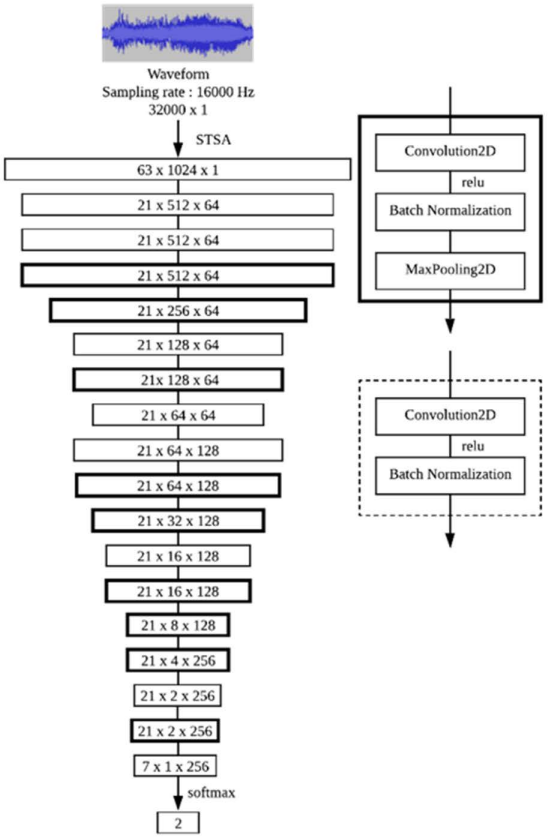


FIGURE 1. The SCNN-18 structure, from [2].

TABLE 1. Hyper-parameters of the SCNN-18 structure. The term "Same" in the padding field means that padding is used and the size of the feature map does not change after convolution.

	No. of Filters	Kernel Size	Padding	Max Pooling	Stride
Layer 1	64	(3,2)	X	X	(3,2)
Layer 2	64	(1,2)	Same	X	(1,1)
Layer 3	64	(1,2)	Same	X	(1,1)
Layer 4	64	(1,2)	Same	(1,2)	(1,1)
Layer 5	64	(1,2)	Same	(1,2)	(1,1)
Layer 6	64	(1,2)	Same	X	(1,1)
Layer 7	64	(1,2)	Same	(1,2)	(1,1)
Layer 8	128	(1,2)	Same	X	(1,1)
Layer 9	128	(1,2)	Same	X	(1,1)
Layer 10	128	(1,2)	Same	(1,2)	(1,1)
Layer 11	128	(1,2)	Same	(1,2)	(1,1)
Layer 12	128	(1,2)	Same	X	(1,1)
Layer 13	128	(1,2)	Same	(1,2)	(1,1)
Layer 14	256	(1,2)	Same	(1,2)	(1,1)
Layer 15	256	(1,2)	Same	(1,2)	(1,1)
Layer 16	256	(1,2)	Same	X	(1,1)
Layer 17	256	(3,2)	Same	(3,2)	(1,1)
Layer 18	256	(1,1)	Same	X	(1,1)

For conventional applications, finding the predicted class based on voting is the end point. However, for an unlabeled sample, if we consider whether the votes are close to tie or overwhelming to one class, it may reveal some useful information. Specifically, among the homogeneous models, some have higher prediction accuracy (for an unknown dataset) and some have lower accuracy. For a particular sample if

almost all models predict the same label, say vocal, then this sample is very likely a vocal sample. On the other hand, if the votes are close to 50-50, then it is hard to determine whether this sample is vocal or not. Following this simple observation, we may consider the votes a sample received as a confidence value of this sample. With the confidence values, we propose an algorithm to estimate the prediction accuracy of an unlabeled dataset.

The proposed approach is outlined as follows.

1. Use the training dataset to train $M = 2N + 1$ homogeneous models, where N is a sufficiently large positive number (such as 10, to be discussed later).
2. Let $b_i \leftarrow 0$ for $1 \leq i \leq N + 1$.
3. Use the trained models to predict the label of one sample in the unlabeled dataset and record the prediction results. Suppose that x models predict the sample as "vocal" and the rest as "nonvocal."
4. If $x \leq N$, then $b_{x+1} \leftarrow b_{x+1} + 1$; otherwise $b_{2N+2-x} \leftarrow b_{2N+2-x} + 1$. In the following, we say that this sample is put in bin b_{x+1} (or equivalently, b_{2N+2-x}). The bin index, $(x+1)$ or $(2N + 2 - x)$, of the sample is its confidence value.
5. Repeat steps 3 to 4 until all samples in the dataset are predicted.
6. Compute $R(k)$ over the cumulated bin number k as
$$R(k) = (\sum_{i=1}^{k} b_i)/T \tag{1}$$
where T is the number of samples in the test dataset, and k is in the range of 1 to $N + 1$.
7. Use $R(k)$ to estimate the prediction accuracy of the test dataset with a linear model (given below).

In the following, we use a numerical example to explain the concept of the proposed approach. Suppose that $N = 10$; therefore, totally 21 models are used in the voting. If the predicted results for a sample are 21:0 or 0:21, then b_1 is incremented by one. If the voting results are 20:1 or 1:20, then b_2 is incremented by one, and so on. When all samples are predicted, we know that b_1 contains the number of samples with voting of 21:0 or 0:21, and b_2 with 20:1 or 1:20. Next, $R(k)$ is the sum of bin contents from 1 to k over the total number of samples, with an upper bound of 1. If $R(4) = 0.9$, it means that 90% of samples have predicted labels agreed by at least 18 models. If $R(4)$ is high, the dataset is likely to have high accuracy. Consequently, we can use $R(k)$ as a confidence quality for prediction accuracy. Therefore, we call $R(k)$ as the **multi-model confidence qualities** (MCQ). Although we use SCNN-18 models in the experiments, the same concept could be applied to estimate the prediction accuracy of other types of models. As how to generate multiple homogeneous models of that type, it could be accomplished by varying the value of a less important hyperparameter.

In actual applications, we need to find an equation to covert from $R(k)$ to the estimated accuracy $\hat{A}$. This can be accomplished by using a simple linear equation, i.e.,
$$\hat{A} = a_0 R(k) + b_0 \tag{2}$$
where a_0 is the slope of the line and b_0 is the y-intercept. If we have multiple labeled datasets, we can obtain the values of a_0 and b_0 by using the linear regression equation [23] based on known $(R(k), \hat{A})$ pairs of the labeled datasets. On the other hand, if we do not have multiple datasets, we can always partition the labeled dataset into a training set and a validation set. Then, use the training set to train the models and use the $\hat{A}$ of the validation set to determine a_0 by assuming $b_0 = 0$.

IV. EXPERIMENTS AND RESULTS

This section covers experiments and results. Before describing the experiments, we first explain the experimental datasets and experimental environment. We then describe the experiments to determine the suitable values for N and k in the proposed approach and the experiments to compare the MCQ with two other approaches in estimating the prediction accuracy of various datasets. Finally, we show an additional benefit of the proposed approach when retraining the model is necessary.

A. EXPERIMENTAL DATASETS

In order to conduct the experiments, we collect many soundtracks from various sources. Next, the audio segments of 2s are excerpted from the soundtracks. Each audio segment is a sample in a dataset. The labels (vocal/nonvocal) of the audio segments from soundtracks without annotations are determined by human listeners. The datasets are available to public [24]. The listing of the datasets is given in Table 2 along with the actual accuracy of the datasets predicted by using the SCNN-18 models trained by Jamendo Train dataset or FMA-C-1 Train dataset, respectively. In the table, some datasets are further divided into training and test sets. With this arrangement, we can train a model and use the model to predict samples from the same source (such as KTV).

The following briefly describes the listed datasets. Jamendo Train and Jamendo Test datasets are excerpted from the Jamendo dataset [15]. The labels of the excerpted segments are based on the associated annotations. The FMA-C-1 and FMA-C-2 datasets are excerpted from the FMA website [4]. A distinct feature of these two datasets is that only one segment is excerpted from one soundtrack [1]. All other datasets may contain multiple segments from the same soundtrack. The difference between FMA-C-1 and FMA-C-2 is that C-2 has a similar number of samples for each (broad) music genre. The Test-Hard dataset is an artificial dataset, containing collections of samples wrongly predicted by a simple 4-layer CNN classifier [25]. The dataset A-Cappella, as its name suggests, contains A-Cappella works collected from the Internet. Thus, this dataset has only vocal samples. The Instrumental dataset has segments containing instrumental performance only (such as piano, flute, horn, etc.), and thus it contains no vocal samples. The music genres of the Instrumental dataset are mostly easy listening (or background music). The KTV samples are excerpted from Karaoke videos. The videos in KTV discs have two audio channels, one with accompaniment only and the other one

TABLE 2. Datasets used in the experiments with true prediction accuracy.

Dataset	Vocal Segments	Nonvocal Segments	Jamendo Train Accuracy	FMA-C-1 Train Accuracy
Jamendo Train	6,981	6,376	-	95.45%
Jamendo Test	1,487	1,499	93.75%	92.73%
FMA-C-1 Train	5,007	7,247	83.88%	-
FMA-C-1 Test	1,669	2,416	82.39%	89.63%
FMA-C-2 Train	5,277	8,475	87.33%	92.23%
FMA-C-2 Test	1,759	2,824	85.73%	90.70%
Test Hard	4,746	3,545	66.14%	72.44%
A-Cappella	7,922	0	95.02%	94.33%
Instrumental	0	7,516	79.72%	87.19%
KTV Train	6,370	164	95.42%	96.53%
KTV Test	1,332	35	94.02%	95.61%
MIR-1K	2,817	2,817	87.02%	89.17%
Chinese-CD Train	3,060	2,163	90.70%	93.09%
Chinese-CD Test	1,280	943	91.05%	92.84%
Taiwanese-CD Train	760	489	90.15%	93.92%
Taiwanese -CD Test	314	213	88.79%	93.46%
Taiwanese-stream Train	2,753	1,396	83.78%	90.00%
Taiwanese-stream Test	1,188	586	82.62%	88.69%
Classical Train	2,007	3,726	83.27%	90.13%
Classical Test	847	1,597	81.47%	89.34%
RWC	5,007	2,978	92.04%	92.98%

with a mix of vocal and accompaniment. The vocal and nonvocal samples are from these two channels, respectively. The MIR-1K dataset is originally used for MIR contest [26]. Therefore, it has a complete annotation for many different purposes. In our case, we rely on the annotation to label samples. The Chinese-CD and Taiwanese-CD datasets are excerptions from various Chinese or Taiwanese CD titles. The audio soundtracks in the CDs are mostly popular music, with a few folk or traditional songs. The Taiwanese-stream dataset contains segments of Taiwanese songs collected over the Internet. The Classical dataset contains only classical music. In this dataset, the nonvocal samples are mostly from orchestra works, whereas vocal samples are from solo or chorus in opera performance. Finally, the RWC dataset [27]–[29] was purchased from C Music Corporation [30]. We excerpt segments only from the ''Popular Music Database & Royalty-Free Music Database'' and use the given annotation for labeling. Unlike other datasets mentioned here where only one or some segments are excerpted from a soundtrack, each of the soundtracks in this dataset is partitioned into a maximum number of nonoverlapping segments of 2s duration. Thus, a 120 seconds soundtrack is partitioned into 60 segments.

B. EXPERIMENTAL ENVIRONMENT

We use three computers with NVIDIA graphic cards to carry out the experiments. The specifications of the computers are listed in Table 3. The simulation programs are written in Python with Tensorflow [21] and Keras [20] tools. The detailed versions of the tools are listed in Table 4.

TABLE 3. Computers used in the experiments.

Processor	Intel Core i7-6850K	Intel Core i9-7900X	Intel Core i7-6850K
Memory	32 GB	40 GB	32 GB
Graphic Card	GeForce GTX 1080ti×3	GeForce GTX 1080ti×3	GeForce RTX 2070×3
OS	Ubuntu 20.04	Ubuntu 20.04	Ubuntu 18.04

TABLE 4. Software versions used in the experiments.

Software	Version
Python	3.6.9
Tensorflow	2.3.0
CUDA	11.2

C. DETERMINING CLASSIFIER NUMBER M AND CUMULATED BIN NUMBER k

Before conducting the experiments, we want to have visual observations of the proposed approach to see if $R(k)$ is close to the true accuracy for some datasets. To this end, we arbitrarily set $M = 21$ (i.e., $N = 10$). The 21 SCNN models are trained with the Jamendo Train dataset. Then, the values of $R(k)$, $1 \leq k \leq 4$, for the following datasets are calculated: FMA-C-1 Train, FMA-C-2 Train, Test-Hard, A-Cappella, Instrumental, KTV Train, MIR-1K, Chinese-CD Train, Taiwanese-CD Train, Taiwanese-stream Train, and Classical Train. Next, one 2-D point $(R_s(k), A_s)$ is constructed with the true accuracy A_s and the MCQ $R_s(k)$ for a dataset s. By placing all points obtained from all datasets on plots, we obtain Fig. 2, where the horizontal axis is the accuracy of datasets and the vertical axis is $R_s(k)$. The lines and the score values in the figure are computed by using the linear regression tool in the Scikit-learn library [31]. Fig. 2 shows that if k is 2, 3 or 4, the linear regression line fits the data points pretty well. Also note that it is reasonable to have higher regression scores if k increases. When $k = 11$, the regression line will be a horizontal line and the score is 1.00. However, in this case, we are unable to do any prediction because $R_s(k) = 1.0$ for any dataset s. Therefore, the regression score is not a useful indicator for evaluating the estimation performance.

In addition to the visual inspection in Fig. 2, we also use the computed regression lines to observe the errors of the estimated model accuracy versus different k values. To this end, we use the points $(R_s(k), A_s)$in Fig. 2 to perform a leave-one-out cross validation (see also subsection IV.D.I). As there are 11 points on the plot, we then use 10 points to construct a linear regression model, and use this linear model to estimate accuracy $\hat{A}_p$ on dataset p (i.e., the left 11^{th}-point) based on $R_p(k)$. The results are shown in Fig. 3. We observe from Fig. 3 that the best k value for Jamendo-trained models is 3, whereas this value is 5 for models trained with the FMA-C-1 dataset. After that, the estimation errors slightly increase.

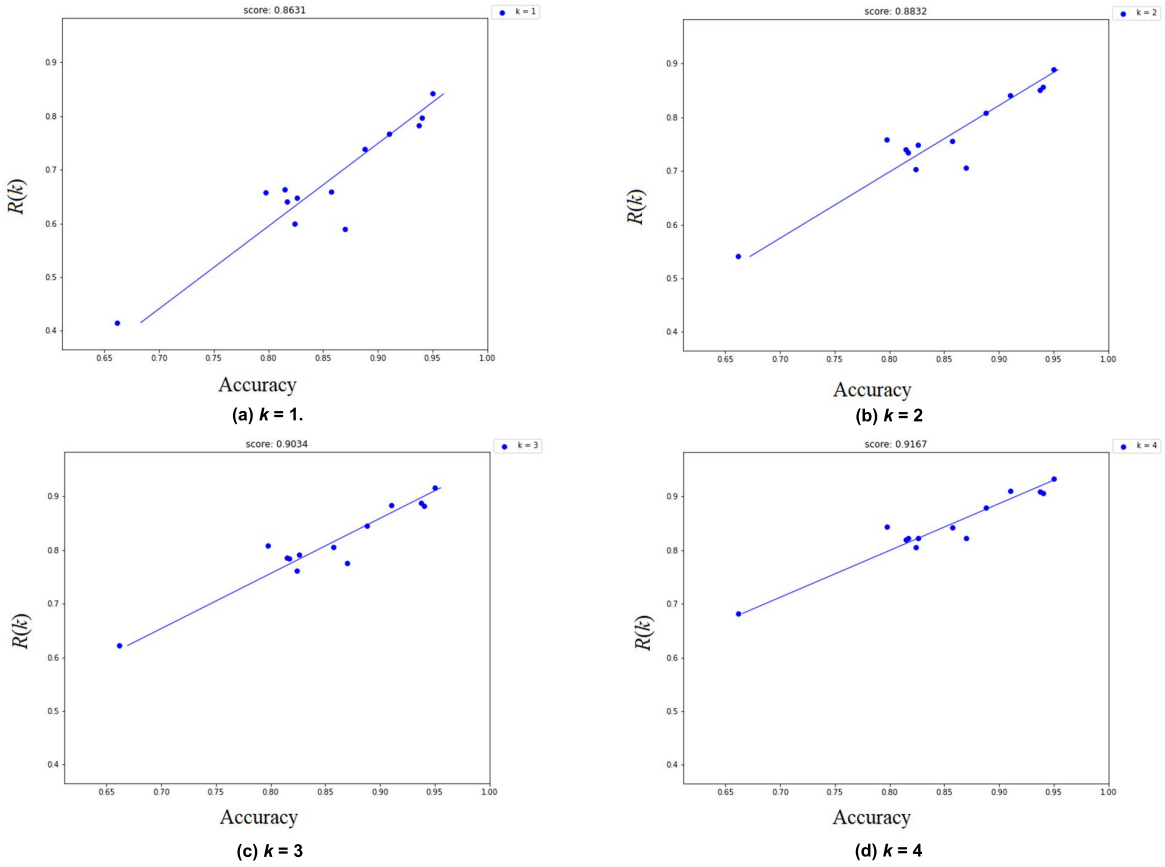


FIGURE 2. The value of R(k) versus true accuracy for some datasets, where (a) to (d) are for k = 1 to 4, respectively.

To further investigate if the proposed approach is sensitive to the number of models, M, we again use the leave-one-out cross validation mentioned in Fig. 3 for various M and $1 \leq k \leq 4$ for models trained with Jamendo Train, FMA-C-1 Train, FMA-C-2 Train, and Chinese CD Train, respectively. The results are shown in Fig. 4, where we observe that there is no specific ''trend'' or stable regions on the curves for all training sets. Therefore, no universal optimal value of M could be obtained. Consequently, it is acceptable to use any reasonable value of M. In addition, M does not significantly affect the prediction errors as the errors are less than 2% for most cases. Actually, Y. Ovadia, *et al.* also observed a similar situation of insensitivity of performance over M when conducting experiments with deep ensemble (the origin of the EAQ approach): ''We found that relatively small ensemble size (e.g. $M = 5$) may be sufficient'' [8]. With this observation, we use **M= 21 and k= 4** in the experiments. Please note that the chosen M and k are not the optimal ones in any of the training datasets (i.e., Jamendo Train, FMA-C-1 Train, FMA-C-2 Train, or Chinese-CD Train).

D. COMPARISON COUNTERPARTS AND PROCEDURE

To have a comparative study of the performance of the proposed approach, we choose the RTQ [6] approach and the ensemble approach modified from B. Lakshminarayanan *et al.* [7] as the comparison targets. Note that the RTQ does not need any hyperparameter. For Lakshminarayanan's ensemble approach, as it only provided an uncertainty value for each test sample, we simply average the computed uncertainty values in a dataset, and call it EAQ (ensemble average qualities). Specifically, assume that there are M models and T samples in the experiments. Let $p_{i,j}$ be the vocal probability for model i and sample j. According to [7], the uncertainty predictive value for sample j to be a vocal sample is estimated as

$$\bar{p}_j = \frac{1}{M} \sum_{i=1}^{M} p_{i,j}. \tag{3}$$

In terms of implementation, $p_{i,j}$ is the softmax output of the vocal class in one model. We then use the average of $\max(\bar{p}_j, 1-\bar{p}_j)$ as the EAQ, i.e.,

$$EAQ = \frac{1}{T} \sum_{j=1}^{T} max(\bar{p}_j, 1 - \bar{p}_j). \tag{4}$$

The chosen models are the same trained models as used in MCQ approach. As the RTQ approach does not benefit from using adversarial samples, we did not use them in the experiments for fair comparison.

The comparison consists of four parts. The first one is again a leave-one-out cross validation. The second one uses Jamendo Test or FMA-C-1 Test to determine a_0 by assuming

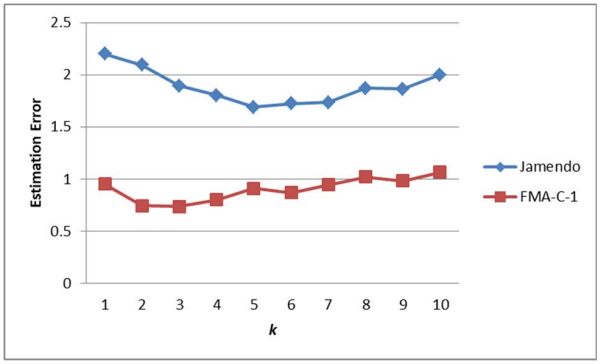


FIGURE 3. The estimation errors of the models for various k values.

$b_0 = 0$ in Eq. (2). The third one uses both leave-one-out and one dataset approaches to estimate the accuracy of the RWC dataset. We conduct this experiment is because the RWC dataset is not used in subsection IV.C to determine k and M in the proposed approach. Finally, we compute the correlation coefficients of MCQ, EAQ, and RTQ versus true accuracy to justify the results of the above three experiments.

The chosen datasets in the first two comparisons are as follows: Jamendo Test, FMA-C-1 Test, FMA-C-2 Test, KTV Test, Chinese-CD Test, Taiwanese-stream Test, Taiwanese-CD Test, Classical Test, MIR-1K, Instrumental, A-Cappella, and Test-Hard. Note that if a dataset, such as Chinese-CD, has a training set and a test set, we always use the test set to conduct experiments here, as the training sets have already been used to select the k and M values. We use the RWC dataset in the third experiment because it is a new dataset to all compared approaches.

1) LEAVE-ONE-OUT CV COMPARISON

This experiment again uses the leave-one-out cross validation to compare the errors of prediction accuracy estimated by three approaches. For the RTQ and EAQ approaches, we follow the same procedure as we did for the MCQ approach in subsection IV.C.

Fig. 5 (a) and (b) show the errors between the true accuracy (given in Table 2) and the estimated accuracy of three approaches with the Jamendo Train and the FMA-C-1 Train datasets to train SCNN-18 models, respectively. The results show that the errors of the RTQ approach fluctuate significantly, some very small and some very large, for models trained with both training sets. Both the proposed and the EAQ approaches have comparable estimation errors for some datasets. However, the proposed approach has notable lower errors on Test-hard and MIR-1k datasets for models trained with the Jamendo Train dataset. For models trained with FMA-C-1 Train, the EAQ yields an unacceptably large estimation error on the Test-hard dataset.

2) USING ONE DATASET TO PREDICT ACCURACY OF OTHER DATASETS

In our experiment in subsection IV.D.I, we assume that multiple labeled datasets are available. In practical applications, we may have only one labeled dataset. Therefore, we also investigate the estimation errors of all three approaches with only one labeled dataset. We conjecture that the experiment in subsection IV.D.I is to determine the performance upper bound of the proposed approach, whereas the experiment here is to determine the performance lower bound.

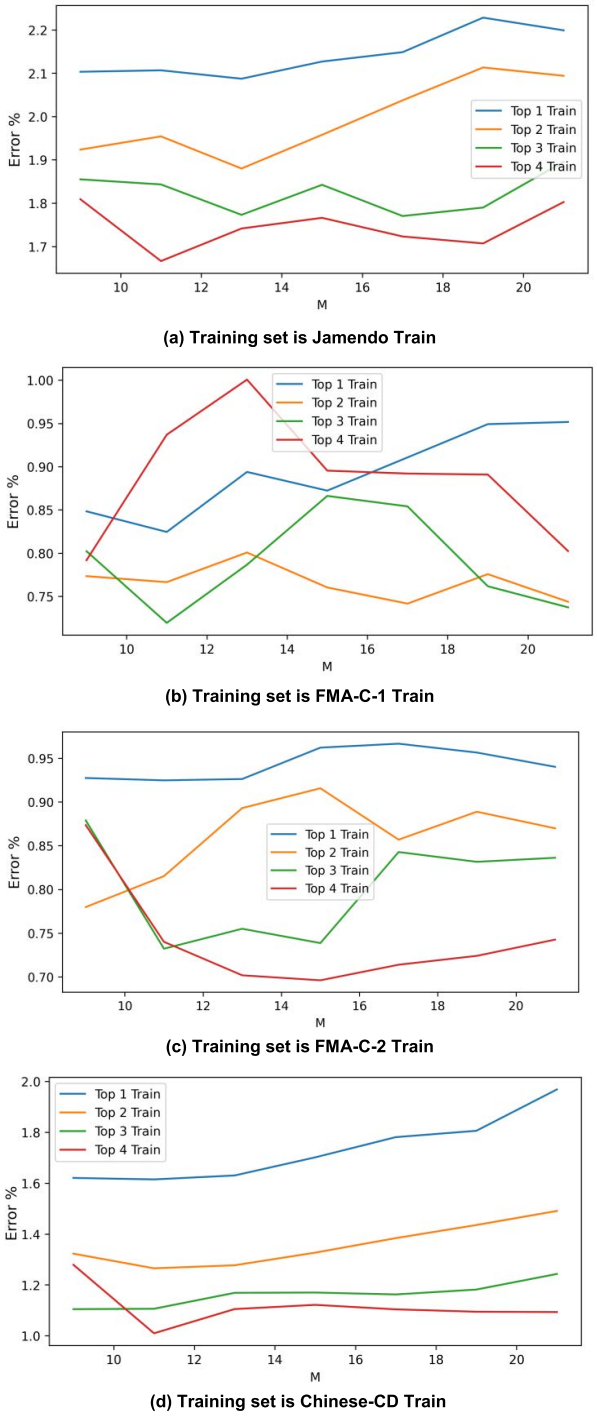


FIGURE 4. The results of leave-one-out cross validation for various numbers of M (up to 21) and k, where the legend Top 1 means k = 1, and so on.

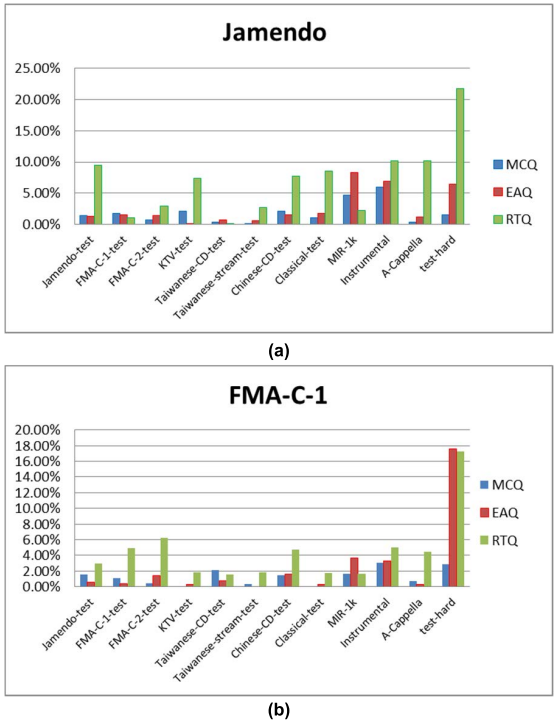


FIGURE 5. Estimation errors of leave-one-out cross validation for all three approaches. (a) Training set is Jamendo Train. (b) Training set is FMA-C-1 Train.

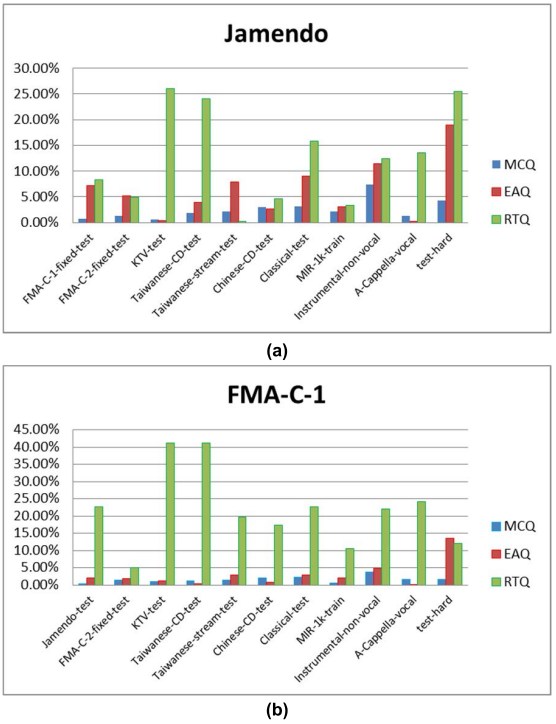


FIGURE 6. Estimation errors with one labeled dataset for all three approaches. (a) Training set is Jamendo. (b) Training set is FMA-C-1.

In this experiment, we use the Jamendo Train dataset to train models, and the Jamendo Test dataset to determine a_0 by assuming $b_0 = 0$ in Eq. (2) for the MCQ and EAQ approaches. The same rule also applies to the FMA-C-1 Train dataset. In addition, the experimental datasets are the same as those used in subsection IV.D.I except the one used for a_0 calculation. For the RTQ approach, the RTQ value is directly used to estimate the prediction accuracy, which is the same as the original paper [6]. Equivalently, it means $a_0 = 1$ and $b_0 = 0$ in Eq. (2).

The estimation errors are shown in Fig. 6 (a) and (b). It is observed that the proposed approach now has larger estimation errors when compared with Fig. 5. However, the estimation errors are mostly within the range of $2 \sim 4\%$. As to the RTQ approach, it has pretty large estimation errors. For Jamendo training set, three datasets have estimation errors greater than 20%, and similarly for FMA-C-1 training set, two datasets have estimation errors over 40%. When comparing the MCQ and EAQ approaches, the EAQ approach has a larger average error on models trained with Jamendo. For models trained with FMA-C-1, although the EAQ is slightly better than the proposed approach, its estimation error on Test-hard dataset is not to be neglected.

Table 5 shows the average estimation errors of the approaches under comparison. Overall speaking, the proposed approach has lower average estimation errors for unlabeled datasets.

TABLE 5. Average errors of various methods.

Approach	Jamendo Training Set	FMA-C-1 Training Set
MCQ CV	1.9%	1.3%
EAQ CV	2.6%	2.5%
RTQ CV	7.0%	4.5%
MCQ One set	2.5%	1.6%
EAQ One set	6.3%	3.0%
RTQ One set	12.6%	21.7%

3) ESTIMATING PREDICATION ACCURACY FOR RWC DATASET

Since we used a wide variety of music genres in our previous experiments, the experimental results might be pessimistic in some applications where the training and test datasets likely have similar styles or genres. In addition, in subsection IV.C we have used some datasets to determine k and M for the proposed approach. Considering these cases, we use the RWC dataset to repeat the above experiments. Recall that the RWC dataset is never used in subsection IV.C and it also contains western popular music, similar to Jamendo or FMA-C-1 training datasets.

The experimental steps are the same as previous experiments, and thus omitted here. The experimental results are shown in Fig. 7(a) and (b). It can be observed that all three approaches now have lower estimation errors although the RTQ approach still performs poorly. As to the MCQ and the

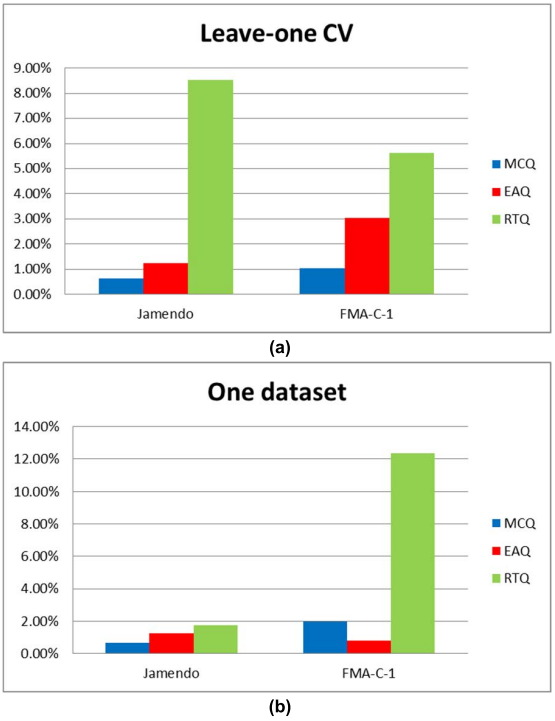


FIGURE 7. Estimation errors of the RWC dataset by using all three approaches. (a) Use leave-one-out cross validation. (b) Use one test dataset.

EAQ approaches, the MCQ has lower estimation errors for models trained with Jamendo. If the models are trained with the FMA-C-1 dataset, the EAQ performs slightly better if only one dataset is used for estimation. However, in this case, the MCQ still has an acceptable estimation error (≈2%) for practical applications. Therefore, this experiment confirms that the proposed approach can be applied to real applications if both labeled and unlabeled datasets have similar music genres.

4) CORRELATION COEFFICIENTS OF COMPARED APPROACHES

To further justify the performance differences among three approaches, we compute the correlation coefficients of the qualities versus true accuracy for the datasets used in subsection IV.D.I. The computed results are given in Table 6. The results show that both the MCQ and EAQ have high correlation coefficients (more than 0.9), whereas the RTQ is not (-0.26 or -0.5). For models trained with Jamendo, the proposed approach has a slightly higher correlation coefficient. This number, in a sense, justifies that the proposed approach has noticeably lower estimation errors in Jamendo-trained models.

E. RE-TRAINING THE MODELS

In this subsection, we show that an additional benefit of using the proposed approach. Suppose that a new dataset is estimated to have low prediction accuracy. If we need to improve the prediction accuracy, what can we do?

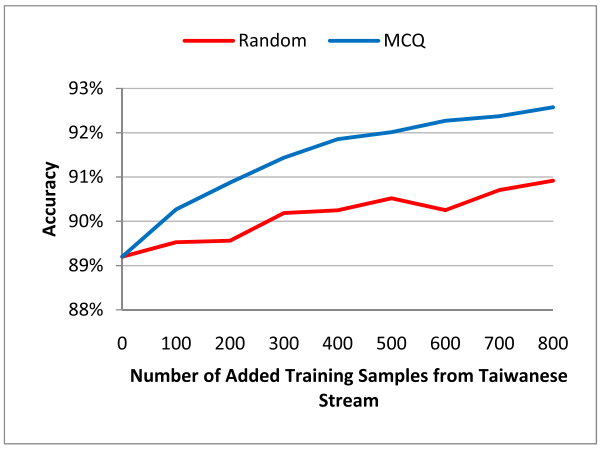
TABLE 6. Correlation coefficients of all three methods.

Approach	Jamendo Training Set	FMA-C-1 Training Set
MCQ	0.96	0.98
EAQ	0.95	0.98
RTQ	-0.26	-0.50

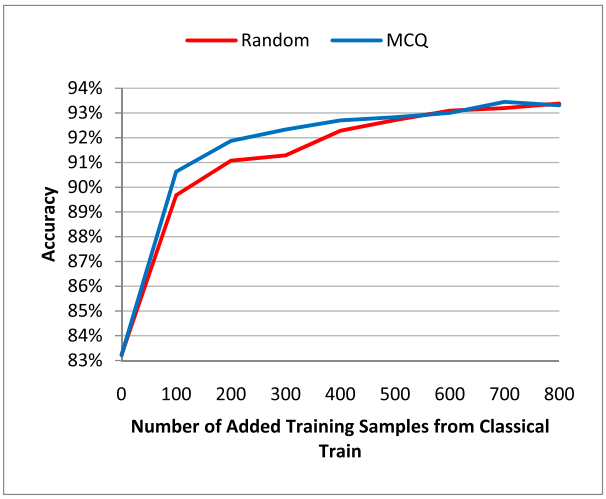
A straightforward solution is to label a small number of samples from the unlabeled dataset. Then, the labeled new samples and the original training samples together are used to train new models. The question is how to select the samples to maximize the prediction accuracy, i.e., new models after re-training have highest possible accuracy for a given number of newly labeled samples. If we have no *a priori* knowledge of the unlabeled samples, we have to randomly pick samples to label. However, in our case, we know that the bin number associated with a sample can serve as the confidence value of the sample. Therefore, with the proposed approach, we can give priority to labeling samples with large bin numbers (i.e., low confidence values). In our case $N = 10$, so we can pick samples in bin_{11} first. If necessary, then pick samples in bin_{10}, bin_9, and so on.

To evaluate if the presented selection method is more effective than a random selection, we conduct the following experiment. The original SCNN-18 models are trained with the Chinese-CD Train dataset. The models are used to compute the confidence values of samples in the Taiwanese-stream Train and Classical Train datasets. The number of samples to be labeled in each dataset is from 100, 200, until 800. The newly labeled samples are added to the Chinese-CD Train dataset to train new SCNN-18 models. The new models are then used to predict the labels of samples in the Taiwanese-stream Test and Classical Test datasets. The average accuracy of ten trials is shown in Fig. 8, where we observe that the original SCNN-18 model has relatively high accuracy for Taiwanese-stream Test dataset (in Fig 8(a)), up to about 89.2%. By adding 200 labeled samples from Taiwanese-stream Train, we are able to improve the accuracy by 1.6% with the proposed approach. On the other hand, the random selection approach improves the accuracy by only about 0.4%. When adding 800 additional training samples, the proposed approach has 92.6% accuracy whereas the random approach has only 90.9%.

For the Classical Test dataset, the original model has an accuracy value of 83.2%, shown in Fig. 8(b). Again, by adding 200 labeled samples from the Classical Train dataset, the accuracy is boosted to 91.9%, whereas the accuracy is 91.1% with the random selection. Based on the results in Fig. 8, we conclude that the proposed selection approach is more effective, especially if the number of selected samples is small, such as 200. Actually, it is more cost-effective to add 200 new labeled samples in this dataset because the accuracy is improved by more than 8%. Adding another 600 new samples only further improves the accuracy by about 1.5%.



(a) Accuracy for predicting Taiwanese-stream Test set.



(b) Accuracy for predicting Classical Test set.

FIGURE 8. The accuracy of new models after adding various numbers of new samples in training.

From this experiment we conclude that we may just need to label a few hundred new samples for the new model to reach satisfactory (say, 90+%) accuracy.

F. DISCUSSIONS AND FUTURE DIRECTIONS

We showed the relative estimation errors of MCQ, EAQ, and RTQ approaches in subsections IV.D.I to IV.D.III. From the experimental results we observe that all three approaches could achieve low estimation errors if the distributions of the training dataset and the unlabeled dataset are close to each other, such as the case of estimating the prediction accuracy of the RWC dataset based solely on the Jamendo training set. On the other hand, if the distributions of the training dataset and the unlabeled dataset are different, then the RTQ approach is not a good approach for accuracy estimation. In this case, both MCQ and EAQ approaches are better choices.

As both MCQ and EAQ approaches are based on the use of multiple homogeneous models, the performance discrepancy between these two approaches is only due to the calculation of the estimation qualities. When closely observing Fig. 5, 6, and 7, and comparing the accuracy given in Table 2, we know that the EAQ has a small estimation error if the trained model has high actual accuracy, or small distribution shift, such as the KTV Test dataset. On the other hand, if the actual accuracy of a dataset is low, the EAQ approach has a noticeably higher estimation error, such as the Test-hard dataset. In comparison, the proposed MCQ approach has much lower estimation errors in the Test-hard dataset. For real applications, we are unable to foresee whether the accuracy predicted by a model is high or low. In this regard, the EAQ approach is not robust enough for datasets with any degree of distribution shifts. On the other hand, the proposed MCQ approach is more robust to distribution shifts, and is a better approach for real applications.

Though the proposed approach is promising, the proposed accuracy estimation approach has some limitations, given below.

- The proposed approach was developed for binary classification problem. Modifications are needed to extend the proposed approach to multi-class problems.
- The labeled dataset must have samples on all classes. For example, we are unable to use the proposed approach with a vocal-only training set, such as A-Cappella.
- As the proposed approach uses multiple identical deep networks, training these deep networks requires a lot of computational resources. Therefore, it is not suitable for applications that are time sensitive and have limited computing resources. For such applications, alternative approaches using only one model could be more appropriate.
- We used 16,343 samples in the Jamendo dataset and 16,339 samples in the FMA-C-1 dataset to conduct the experiments. When the proposed approach is applied with a small training set, performance degradation might occur.

Previously we mentioned that the experiment with the leave-one-out cross validation could be used to estimate the performance upper bound of the proposed approach. The experimental results given in Table 5 confirm this conjecture. A meaningful future direction is how to approach the performance attained with the leave-one-out cross validation approach, but with one set, not multiple sets, of labeled data. For this problem, we prepare to study various types of data augmentation methods [32] to see if any of the methods could artificially produce multiple labeled datasets so that the leave-one-out cross validation approach could be applied.

Recently, the self-supervised learning approach [33] has received lots of attention. It would be interesting to investigate if the proposed approach could also be applied to models trained with the self-supervised learning approach.

V. CONCLUSION

In this paper, we present the use of multiple models to estimate the prediction accuracy of an unlabeled dataset.

The experimental results show that, when compared with the RTQ method, the proposed approach has lower (or much lower) estimation errors. When compared with the EAQ approach, the proposed approach is more robust for out-of-distribution datasets. In addition, we propose to use the bin number as the confidence value for unlabeled samples. The confidence value helps to determine which samples to label first in case a re-training is necessary. In the future, we plan to investigate how to use the data augmentation technique to improve the estimation accuracy and how to apply the proposed approach to models trained with the self-supervised learning approach.

REFERENCES

[1] S. D. You, C.-H. Liu, and W.-K. Chen, "Comparative study of singing voice detection based on deep neural networks and ensemble learning," *Hum.-Centric Comput. Inf. Sci.*, vol. 8, no. 1, p. 34, Nov. 2018.

[2] S. D. You, C.-H. Liu, and J.-W. Lin, "Improvement of vocal detection accuracy using convolutional neural networks," *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 2, pp. 729–748, 2021.

[3] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A dataset for music analysis," in *Proc. Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, 2017, pp. 316–323.

[4] *Free Music Archive*. Accessed: Nov. 17, 2021. [Online]. Available: https://freemusicarchive.org/

[5] M. Ramona, G. Richard, and B. David, "Vocal detection in music with support vector machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2008, pp. 1885–1888.

[6] D. Bhaskaruni, F. P. Moss, and C. Lan, "Estimating prediction qualities without ground truth: A revisit of the reverse testing framework," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 49–54.

[7] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6405–6416.

[8] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 13991–14002.

[9] S. Fort, H. Hu, and B. Lakshminarayanan, "Deep ensembles: A loss landscape perspective," 2019, *arXiv:1912.02757*.

[10] J. Liu, J. Paisley, M.-A. Kioumourtzoglou, and B. Coull, "Accurate uncertainty estimation and decomposition in ensemble learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 8952–8963.

[11] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient Langevin dynamics," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 681–688.

[12] R. Moradi, S. Cofre-Martel, E. Lopez Droguett, M. Modarres, and K. M. Groth, "Integration of deep learning and Bayesian networks for condition and operation risk monitoring of complex engineering systems," *Rel. Eng. Syst. Saf.*, vol. 222, Jun. 2022, Art. no. 108433, doi: 10.1016/j.ress.2022.108433.

[13] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Inf. Fusion*, vol. 76, pp. 243–297, Dec. 2021.

[14] W. Fan and I. Davidson, "Reverse testing: An efficient framework to select amongst classifiers under sample selection bias," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2006, pp. 147–156.

[15] M. Rocamora and P. Herrera, "Comparing audio descriptors for singing voice detection in music audio files," in *Proc. 11th Brazilian Symp. Comput. Music*, San Pablo, Brazil, 2007, pp. 187–196.

[16] A. L. Berenzweig and D. P. W. Ellis, "Locating singing voice segments within music signals," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, New Paltz, NY, USA, Oct. 2001, pp. 119–122.

[17] S. Leglaive, R. Hennequin, and R. Badeau, "Singing voice detection with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brisbane, QLD, Australia, Apr. 2015, pp. 121–125.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jun. 2014.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.

[20] *Keras: Simple. Flexible. Powerful*. Accessed: Nov. 17, 2021. [Online]. Available: https://keras.io/

[21] *Tensorflow: An end-to-end Open Source Machine Learning Platform*. Accessed: Nov. 17, 2021. [Online]. Available: https://www.tensorflow.org/

[22] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*.

[23] *Simple Linear Regression*. Accessed: Mar. 17, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Simple_linear_regression

[24] *NTUT-LabASPL*. Accessed: Apr. 8, 2022. [Online]. Available: https://github.com/NTUT-LabASPL

[25] H.-M. Huang, W.-K. Chen, C.-H. Liu, and S. D. You, "Singing voice detection based on convolutional neural networks," in *Proc. 7th Int. Symp. Next Gener. Electron. (ISNE)*, Taipei, Taiwan, May 2018, pp. 1–4.

[26] *MIR-1K Dataset*. Accessed: Nov. 17, 2021. [Online]. Available: https://sites.google.com/site/unvoicedsoundseparation/mir-1k

[27] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Popular, classical and jazz music databases," in *Proc. 3rd Int. Conf. Music Inf. Retr. (ISMIR)*, 2002, pp. 287–288.

[28] M. Goto, "Development of the RWC music database," in *Proc. 18th Int. Congr. Acoust. (ICA)*, 2004, pp. 1–4.

[29] M. Mauch, H. Fujihara, K. Yoshii, and M. Goto, "Timbre and melody features for the recognition of vocal activity and instrumental solos in polyphonic music," in *Proc. 12th Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, 2011.

[30] *RWC Music Database*. Accessed: Mar. 22, 2021. [Online]. Available: https://staff.aist.go.jp/m.goto/RWC-MDB/

[31] *Linear Regression in Sklearn Model*. Accessed: Mar. 17, 2022. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

[32] J. Schlüter and T. Grill, "Exploring data augmentation for improved singing voice detection with neural networks," in *Proc. 16th Int. Soc. Music Inf. Retr. Conf.*, Malaga, Spain, 2015, pp. 121–126.

[33] A. N. Carr, Q. Berthet, M. Blondel, O. Teboul, and N. Zeghidour, "Self-supervised learning of audio representations from permutations with differentiable ranking," *IEEE Signal Process. Lett.*, vol. 28, pp. 708–712, 2021.

SHINGCHERN D. YOU (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of California at Davis, Davis, CA, USA, in 1993. He is currently a Professor with the Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan. His research interests include machine learning and applications of signal processing to audio and communication systems.

HSIAO-CHUNG LIU received the M.S. degree in computer science and information engineering from the National Taipei University of Technology, Taiwan, in 2021. He is currently a Software Engineer at Super Micro Computer Inc. His research interests include vocal detection, deep learning applications, and software engineering.

CHIEN-HUNG LIU received the M.S. degree in electrical engineering from the University of Southern California, in 1994, and the Ph.D. degree in computer science and engineering from the University of Texas at Arlington, in 2002. He is currently an Associate Professor with the Computer Science and Information Engineering Department, National Taipei University of Technology, Taiwan. His research interests include software testing, vocal detection, deep learning applications, and software engineering.