Loss Function Approaches for Multi-label Music Tagging

1st Dillon Knox  
Department of Electrical Engineering (of USC)  
dillonkn@usc.edu

2nd Timothy Greer  
Department of Computer Science (of USC)  
timothdg@usc.edu

3rd Benjamin Ma  
Department of Computer Science (of USC)

4th Emily Kuo  
Department of Electrical Engineering (of USC)

5th Krishna Somandepalli  
Department of Electrical Engineering (of USC)

6th Shrikanth Narayanan  
Department of Electrical Engineering (of USC)  
shri@sipi.usc.edu

Abstract—Given the ever-increasing volume of music created and released every day, it has never been more important to study automatic music tagging. In this paper, we present an ensemble-based convolutional neural network (CNN) model trained using various loss functions for tagging musical genres from audio. We investigate the effect of different loss functions and resampling strategies on prediction performance, finding that using focal loss improves overall performance on the the MTG-Jamendo dataset: an imbalanced, multi-label dataset with over 18,000 songs in the public domain, containing 57 labels. Additionally, we report results from varying the receptive field on our base classifier—a CNN-based architecture trained using Mel spectrograms—which also results in a model performance boost and state-of-the-art performance on the Jamendo dataset. We conclude that the choice of the loss function is paramount for improving on existing methods in music tagging, particularly in the presence of class imbalance.

Index Terms—music tagging, loss functions, multi-label deep learning, convolutional neural networks

I. INTRODUCTION AND RELATED WORK

Content-based automatic music tagging is a challenging task: a robust system must accurately predict multi-label tags associated with a piece of music (such as mood, theme, or genre), regardless of the frequency of such labels. Great strides in the field have been enabled recently by the release of large, high-quality music datasets, like the MTG-Jamendo dataset [1]. In MediaEval’s Emotions and Themes in Music challenge, participants are tasked with building models that maximize multi-label tag prediction performance on the autotagging-moodtheme subset of this dataset [2], further driving research on multi-label music information retrieval and music tagging.

Models based on convolutional neural networks (CNNs) are an effective choice for a wide variety of audio-based tasks, including speech recognition [3], acoustic scene classification [4], and music-related tasks, like the 2019 MediaEval Emotions and Themes in Music challenge [5, 6]. Inspired by the success of CNNs in previous music tagging applications, we utilize a VGGish-based short-chunk CNN with residual connections in this paper, as implemented in [7], extending this work by experimenting with different loss functions designed to address multi-label class imbalance.

Class imbalance within a dataset is a difficult hurdle for many applications of machine learning, as it can have a significantly detrimental effect on overall performance [8, 9]. This problem extends to the multi-label setting, where the imbalance for each class will be high in most cases with a large number of classes, as there are generally only a handful of classes per sample, leading to preponderance of negative labels [10]. We attempt to resolve this issue by experimenting with mixup [6], class-aware sampling [11], and various loss functions that address class imbalance.

We test focal loss, originally developed for computer vision models, as a way to train the model to emphasize improving predictions on samples that have lower confidence [12]. We also examine class-balanced loss, which increases loss penalties for under-represented classes [13]. Finally, we evaluate the recently-proposed distribution-balanced loss, which attempts to overcome the confounding issues of label co-occurrence and negative-class dominance by rebalancing weights with respect to class co-occurrence and incorporating negative-tolerant regularization [14]. Using these different loss functions, we improve on the performance achieved by the highest-performing previous model [5] on the MTG-Jamendo dataset, demonstrating state-of-the-art performance on this task.

II. DATA PREPARATION

We use the autotagging-moodtheme subset from MTG-Jamendo in this study. This subset includes 18,486 audio tracks with mood and theme annotations, such as “happy” and “film.” In total, there are 57 tags, and tracks often have more than one tag. The dataset is not balanced: the top 14 labels are.

1In general, the multi-label vector for each sample will be sparse.
present in 68% of the samples and the imbalance factor (the count of the most common tag divided by the count of the least common tag) is 15.7. See Fig. 1. Additionally, the cross-correlation matrix of the labels reveals that there are only a few pairs of labels that have high co-occurrence.  

2The “Christmas” and “Holiday” labels had the highest cross-correlation with $\rho = .43$

Fig. 1. The Jamendo tag distribution. Every other tag is listed for legibility. The most frequent tag is “happy” and the least frequent tag is “sexy.” There are a number of tail tags, or tags that are not well-represented in the dataset.

Given the class imbalance of this dataset and lack of cross-correlation between the most frequent labels, we chose to expand the training set by using instances from the Music4All dataset [15] that exactly match the challenge label set, resulting in an additional 5,666 musical samples. The expanded dataset is shown in Fig. 2. Additionally, we pretrain the low-level convolutional layers of our models using over 2,000 hours of audio from the Million Song Dataset (MSD) [16], in the manner presented by Won et al. [7].

For feature extraction, we first resample all audio to 16 kHz, and then extract 128-bin Mel spectrograms using a 512-sample FFT, with a window size of 32 ms and 50% window overlap, similar to [7].

III. APPROACH

A. Model Architecture

We use a short-chunk CNN (first proposed by [17]) with residual connections for our model, based on the architecture presented by Won et al. [7], with the following modifications. We increased the receptive field of the CNN, given that our label set is generally composed of high-level music descriptors, such as emotions and themes. We hypothesize that increasing the receptive field will help capture information necessary for genre tagging, as it synthesized inputs in such a way to make it more amenable for long-term modeling. The original model has seven convolutional layers; we modify the stride of the final two layers to increase the receptive field along the temporal dimension from 3.69 seconds [7] to 4.6 seconds, which results in a better overall performance. Koutini et al. [5] have previously investigated the affect of receptive field size on overall performance, but found varying receptive field along the temporal dimension to have a lower impact on performance than varying along the frequency dimension.

For training, we used a combined optimization method, where an Adam optimizer [18] with a learning rate of 5e-4 was used for the initial 80 epochs, and then a stochastic gradient descent (SGD) method was used for the remainder of training. The batch size was set to 32, and each model was trained for 200 epochs, with checkpointing for best validation loss after each epoch.

B. Loss Functions

We explore three loss functions aimed at increasing average class-wise performance on an imbalanced dataset. Where applicable, we modify the loss functions for multi-label classification. Note that our implementations assume multi-hot encoding for the label vectors.

1) Focal Loss: We implement a multi-label version of focal loss [12]. The focal loss for a sample $y$ is given as

$$FL(y) = -\sum_{c=1}^{C} y_c \alpha (1 - \hat{y}_c)^\gamma \log(\hat{y}_c) + (1 - y_c)(1 - \alpha)(\hat{y}_c)^\gamma \log(1 - \hat{y}_c)$$

where $\hat{y}_c$ is the softmax output for $y$ at label $c$ and $y_c$ is one if $c$ is a label for $y$ and zero otherwise. In our experiments, we set parameter $\alpha = 0.25$ and $\gamma = 2$, as recommended by [12]. Here, $\gamma$ suppresses the contribution to loss from the relatively well-classified examples, focusing instead on harder-to-classify examples. In the case of our dataset, where no single label is present in a majority of instances, the negative classes may be easy for the model to learn. Thus, we instead want to focus on the harder cases where a given label is present. $\alpha$
provides an additional weight term for positive and negative classes.

2) Class-Balanced Loss: We also implement a class-balanced version of focal loss [13]. Here, the focal loss weight $\alpha$ is replaced by a ratio based on the number of samples containing a given label. Concretely:

$$CBL(y) = -\sum_{c=1}^{C} \frac{1 - \beta}{1 - \beta n_c} y_c(1 - \hat{y}_c)^\gamma \log(\hat{y}_c) + (1 - y_c)(\hat{y}_c)^\gamma \log(1 - \hat{y}_c)$$

where $n_c$ is the number of samples in the training set in which label $c$ appears and $\beta$ is a hyperparameter. We set $\beta$ to 0.995 for our experiments.

3) Distribution-Balanced Loss: Lastly, we use distribution-balanced loss, which was first presented by Wu et al. [14]:

$$DBL(y) = \frac{1}{C} \sum_{c=1}^{C} r_c y_c \log(1 + e^{-(\hat{y}_c - \nu_c)}) + \frac{1}{\lambda} (1 - y_c) \log(1 + e^{\lambda(\hat{y}_c - \nu_c)})$$

Here, $\lambda$ is a scale factor for the negative logits, controlling for the preponderance of negative labels, while $r_c$ is a class-specific rebalancing weight that tries to close the gap between the expected number of samples and actual number of samples for a given class after resampling. We set $\lambda$ to 2.0 for our experiments, in line with the experiments by Wu et al. [14], and the default recommended by Lin et al. [12]. Wu et al. used a binary cross-entropy (BCE) variant of this equation, as shown above; we implement a focal-loss-based function for our study.

IV. RESULTS

A. Our Models

We report the results from models trained using the following approaches: the short-chunk CNN model described above using using one the above four different loss functions (BCE, focal loss, class-balanced focal loss, and distribution-balanced focal loss), and an ensemble model comprised of the individual models trained using each of the four different loss functions. Additionally, we trained an identical ensemble model using only data from the MTG-Jamendo dataset, to demonstrate performance training on a smaller dataset (see Table IV).3

B. Results

We display the PR-AUC and ROC-AUC test-set performance of each loss function approach against provided baselines in Table I.

We find that the model trained using focal loss produces the best performance both in terms of PR-AUC and ROC-AUC. All variants of focal loss outperform binary cross-entropy, and our final ensemble of averaging the predictions from models trained using the four different loss functions achieves the highest performance.

To further investigate the effects of using different loss functions on class-wise performance, we split the label set into head, middle, and tail classes, based on frequency in the training set. Head classes contain over 550 samples, middle classes contain between 200 and 550 samples, and tail classes contain at most 200 samples. The results in terms of PR-AUC are displayed in Table II.

Indeed, we find that the focal loss-based methods perform better than BCE loss on the less-frequent classes (both middle and tail subsets). In our experiments, this comes with a slight penalty in performance on the head classes, but leads to overall better performance, as well as a better-performing ensemble model.

Lastly, we try class-aware sampling [11] for all models, but observe a performance degradation in each. The results are displayed in Table III. Interestingly, distribution-balanced loss was formulated with resampling as an initial step, making the drop-off in performance here rather surprising. Further tuning of the loss function hyperparameters could be warranted. We additionally experiment with mixup [19], which has been shown to lead to performance increases on this task [5]. We find that mixup does indeed increase performance using binary cross-entropy, but shows lower performance for focal loss and its variants.

We submitted our best models to the 2020 MediaEval Emotions and Themes in Music Challenge [2], where our ensemble model (using pretraining and Music4All data) and our best single model outperformed all prior and contemporary submissions to the challenge. We show our models’ performances in Table IV.

3Our code can be found at https://github.com/usc-sail/media-eval-2020

<table>
<thead>
<tr>
<th>Approach</th>
<th>Head</th>
<th>Middle</th>
<th>Tail</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCE Loss</td>
<td>0.179</td>
<td>0.163</td>
<td>0.101</td>
</tr>
<tr>
<td>Focal Loss</td>
<td>0.174</td>
<td>0.171</td>
<td>0.113</td>
</tr>
<tr>
<td>CB Focal Loss</td>
<td>0.173</td>
<td>0.170</td>
<td>0.104</td>
</tr>
<tr>
<td>DB Focal Loss</td>
<td>0.173</td>
<td>0.170</td>
<td>0.105</td>
</tr>
<tr>
<td>Ensemble</td>
<td>0.182</td>
<td>0.179</td>
<td>0.109</td>
</tr>
</tbody>
</table>

TABLE I

TEST-SET PERFORMANCE OF OUR MODEL TRAINED USING VARIOUS LOSS FUNCTIONS. BCE STANDS FOR “BINARY CROSS-ENTROPY”; CB FOR “CLASS-BALANCED”; DB FOR “DISTRIBUTION-BALANCED.”

<table>
<thead>
<tr>
<th>Approach</th>
<th>PR-AUC</th>
<th>ROC-AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCE Loss</td>
<td>0.150</td>
<td>0.766</td>
</tr>
<tr>
<td>Focal Loss</td>
<td>0.156</td>
<td>0.778</td>
</tr>
<tr>
<td>CB Focal Loss</td>
<td>0.153</td>
<td>0.773</td>
</tr>
<tr>
<td>DB Focal Loss</td>
<td>0.153</td>
<td>0.768</td>
</tr>
<tr>
<td>Ensemble</td>
<td>0.161</td>
<td>0.781</td>
</tr>
</tbody>
</table>

TABLE II

CLASS-WISE SUBSET PERFORMANCE OF VARIOUS LOSS FUNCTIONS IN TERMS OF PR-AUC.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Head</th>
<th>Middle</th>
<th>Tail</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCE Loss</td>
<td>0.179</td>
<td>0.163</td>
<td>0.101</td>
</tr>
<tr>
<td>Focal Loss</td>
<td>0.174</td>
<td>0.171</td>
<td>0.113</td>
</tr>
<tr>
<td>CB Focal Loss</td>
<td>0.173</td>
<td>0.170</td>
<td>0.104</td>
</tr>
<tr>
<td>DB Focal Loss</td>
<td>0.173</td>
<td>0.170</td>
<td>0.105</td>
</tr>
<tr>
<td>Ensemble</td>
<td>0.182</td>
<td>0.179</td>
<td>0.109</td>
</tr>
</tbody>
</table>
TABLE III
TEST-SET PERFORMANCE OF OUR MODEL TRAINED USING VARIOUS LOSS FUNCTIONS WITH CLASS-AWARE RESAMPLING.

<table>
<thead>
<tr>
<th>Approach</th>
<th>PR-AUC</th>
<th>ROC-AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCE Loss</td>
<td>0.148</td>
<td>0.759</td>
</tr>
<tr>
<td>Focal Loss</td>
<td>0.146</td>
<td>0.763</td>
</tr>
<tr>
<td>CB Focal Loss</td>
<td>0.141</td>
<td>0.760</td>
</tr>
<tr>
<td>DB Focal Loss</td>
<td>0.147</td>
<td>0.759</td>
</tr>
<tr>
<td>Ensemble</td>
<td>0.152</td>
<td>0.768</td>
</tr>
</tbody>
</table>

TABLE IV
OUR MODELS COMPARED TO MEDIAEVAL’S EMOTIONS AND THEMES IN MUSIC TASK. OUR MODELS (IN BOLD) SHOW STATE-OF-THE-ART PERFORMANCE ON PREDICTING MULTI-LABEL MUSIC GENRE TAGS ON THE MTG-JAMENDO DATASET.

<table>
<thead>
<tr>
<th>Submission</th>
<th>PR-AUC</th>
<th>ROC-AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ensemble_all_data</td>
<td>0.161</td>
<td>0.781</td>
</tr>
<tr>
<td>best_single_model</td>
<td>0.156</td>
<td>0.778</td>
</tr>
<tr>
<td>CP-JKU_avg_ensemble [5]</td>
<td>0.155</td>
<td>0.773</td>
</tr>
<tr>
<td>ensemble_jamendo_only</td>
<td>0.142</td>
<td>0.763</td>
</tr>
<tr>
<td>HCMUS_ensemble_wavenet</td>
<td>0.141</td>
<td>0.766</td>
</tr>
<tr>
<td>AugsBurger_fusion_with_attention_CNN</td>
<td>0.131</td>
<td>0.753</td>
</tr>
</tbody>
</table>

V. CONCLUSION

We present a state-of-the-art ensemble-based CNN model for predicting emotions and themes labels for music. We find that focal loss helps predict labels that do not occur frequently in the dataset, and that emphasizing correct predictions of these labels results in better model performance. We posit that the choice of a loss function is an essential consideration when developing a prediction model for multi-label classification, particularly in music processing. Future work will examine whether our approach generalizes to other tasks in automatic music perception, such as music emotion recognition.

VI. ACKNOWLEDGMENTS

This study is done at the Center for Computational Media Intelligence at University of Southern California’s Signal Analysis and Interpretation Laboratory and supported by research awards from Google and the U.S. Chamber of Commerce Foundation.

REFERENCES


