AN ATTENTION MECHANISM FOR MUSICAL INSTRUMENT RECOGNITION

Siddharth Gururani[1] **Mohit Sharma**[2] **Alexander Lerch**[1]

[1] Center for Music Technology, Georgia Institute of Technology, USA
[2] School of Interactive Computing, Georgia Institute of Technology, USA
{siddgururani, mohit.sharma, alexander.lerch}@gatech.edu

ABSTRACT

While the automatic recognition of musical instruments has seen significant progress, the task is still considered hard for music featuring multiple instruments as opposed to single instrument recordings. Datasets for polyphonic instrument recognition can be categorized into roughly two categories. Some, such as MedleyDB, have strong per-frame instrument activity annotations but are usually small in size. Other, larger datasets such as OpenMIC only have weak labels, i.e., instrument presence or absence is annotated only for long snippets of a song. We explore an attention mechanism for handling weakly labeled data for multi-label instrument recognition. Attention has been found to perform well for other tasks with weakly labeled data. We compare the proposed attention model to multiple models which include a baseline binary relevance random forest, recurrent neural network, and fully connected neural networks. Our results show that incorporating attention leads to an overall improvement in classification accuracy metrics across all 20 instruments in the OpenMIC dataset. We find that attention enables models to focus on (or 'attend to') specific time segments in the audio relevant to each instrument label leading to interpretable results.

1. INTRODUCTION

Musical instruments, both acoustic and electronic, are necessary tools to create music. Most musical pieces comprise of a combination of multiple musical instruments resulting in a mixture with unique timbre characteristics. Humans are fairly adept at recognizing musical instruments in the music they hear. Recognizing instruments automatically, however, is still an active area of research in the field of Music Information Retrieval (MIR). Instrument recognition in isolated note or single instrument recordings has achieved a fair amount of success [14, 26]. Recognizing instruments in music with multiple simultaneously playing instruments, however, is still a hard problem. The task is difficult because of (i) the superposition (in both time and frequency) of multiple sources/instruments, (ii) the large variation of timbre within one instrument, and (iii) the lack of annotated data for supervised learning algorithms.

Identifying music in audio recordings is helpful for general retrieval systems by allowing users to search for music with specific instrumentation [32]. Instrument recognition can also be helpful for other MIR tasks. For example, instrument tags may be vital for music recommendation systems to model users' affinity towards certain instruments, genre recognition systems could also improve with genre-dependent instrument information. Building models conditioned on a reliable detection of instrumentation could also lead to improvements for tasks such as automatic music transcription, source separation, and playing technique detection.

As mentioned above, one of the challenges in MIR in general, and in instrument recognition in particular, is the lack of large-scale annotated or labeled data for supervised machine learning algorithms [17, 36]. Datasets for instrument recognition in polyphonic music can broadly be divided into strongly and weakly labeled. A weakly labeled dataset (WLD) contains clips that may be several seconds long and have labels for one or more instruments for their entirety without annotating the exact onset and offset times of the instruments. A strongly labeled dataset (SLD), however, contains audio with fine-grained labels of instrument activity. WLDs are easier to annotate compared to SLDs and therefore scale better. Even though SLDs enable strong supervision of learning algorithms, the smaller size may lead to poor performance of deep learning methods. WLDs, however, have the disadvantage that an instrument may be marked positive even if the instrument is active for a very short duration of the entire clip. This makes it challenging to train models with WLDs.

Models for recognition in weakly labeled data may benefit from inferring the specific location in time of the instrument to be recognized. We formulate the polyphonic instrument recognition task as a multi-instance multi-label (MIML) problem, where each weakly labeled example is a collection of short-time instances, each with a contribution towards the labels assigned to the example. Toward that end, we apply an attention mechanism to aggregate the predictions for each short-time instance and compare this approach to other models which include binary-relevance

random forests, fully connected networks, and recurrent neural networks. We hypothesize that the ability of the attention model to weigh relevant and suppress irrelevant predictions for each instrument leads to better classification accuracy. We visualize the attention weights and find that the model is able to mostly localize the instruments, thereby enhancing the interpretability of the classifier.

The next section reviews literature in instrument recognition and audio tagging or classification. Sect. 3 discusses various datasets for instrument recognition and the challenges associated. Next, Sect. 4 formulates the problem and describes the model. Sect. 5 specifies the various experiments and the evaluation metrics to measure performance. We report the results of the experiments and discuss them in Sect. 6. Finally, in Sect. 7 we conclude the paper suggesting future directions for research.

2. RELATED WORK

2.1 Musical Instrument Recognition

Instrument recognition in audio containing a single instrument can refer to both recognition from isolated notes or recognition from solo recordings of pieces. We refer to [15, 26] for a review of literature in single instrument and monophonic instrument recognition.

Current research has focused on instrument recognition in polyphonic and multi-instrument recordings. While traditional approaches extract features followed by classification algorithms were previously prevalent [9, 21], deep neural networks have dominated recent work in this field. Han et al. [13] applied Convolutional Neural Networks (CNNs) to the task of predominant instrument recognition on the IRMAS dataset [5] and outperformed various feature-based techniques. Li et al. [25] proposed to learn features from raw audio using CNNs for instrument recognition using the MedleyDB dataset [4]. Gururani et al. [12] compared various neural network architectures for instrument activity detection using two multi-track datasets containing fine-grained instrument activity annotations: MedleyDB and Mixing Secrets [11] . They found significant improvement of CNNs and Convolutional Recurrent Neural Networks (CRNNs) over fully connected networks and proposed a method for visualizing model confusion in a multi-label setting. Hung et al. [19] utilized the fine-grained instrument activity as well as pitch annotations in the MusicNet dataset [33] and showed the benefits of pitch-conditioning on instrument recognition performance. In follow-up research, Hung et al. [18] proposed a multi-task learning approach for instrument recognition involving the prediction of pitch in addition to instrumentation. They released a synthetic, large-scale, and strongly-labeled dataset generated from MIDI files for evaluation and found that multi-task learning outperforms their previous approach of using pitch features as additional inputs.

2.2 Audio event detection, tagging and classification

The task of audio or sound event classification shares many commonalities with instrument recognition. Both tasks aim to identify a time-variant sound source in a mixture of multiple sound sources. A few key differences are that research in sound event classification typically focuses on uncorrelated sounds such as motor noise, car horns, baby cries, or dog barks, while musical audio is highly correlated. Additionally, music has a rich harmonic and temporal structure usually absent in audio captured from real world acoustic scenes.

For a historic review of work in sound event and audio classification, we refer readers to the survey article by Stowell et al. [31]. We focus on more recent literature involving deep neural network architectures —which are now the standard approach— as well as on methods that focus on addressing weak labels.

Hershey et al. [16] adapted deep CNN architectures from computer vision and found that they are effective for large-scale audio classification. Cakir et al. [6] researched the benefits of CRNNs for sound event detection over models comprising of only CNNs. They found that the ability of RNNs to capture long-term temporal context helps improve performance against models only comprising CNNs. Adavanne et al. [1] proposed to use spatial features extracted from multi-channel audio as inputs for CRNN architectures. They found that presenting these features as separate layers to the model outperforms concatenation of these features at the input stage.

Learning from weakly labeled data has also been a focus in audio classification. Most works utilize the Multiple-Instance Learning (MIL) framework for the task, where each example is a labeled bag containing multiple instances whose labels are unknown. Kumar and Raj [24] utilized support vector machines and neural networks for solving the MIL problem. They train bag-level classifiers capable of predicting instances and are hence also useful for localization of sound events. Similarly, Kong et al. [22] proposed decision-level attention to solve the MIL problem for Audio Set [10] classification. Attention is applied to instance predictions to enable weighted aggregation for bag-level prediction. Kong et al. [23] extended this and propose feature-level attention where instead of applying attention to the instance predictions, it is applied to the hidden layers of a neural network to construct a fixed-size embedding for the bag. Finally a fully connected network predicts the labels for the bag using the embedding vector. McFee et al. [27] compared various methods for aggregating or pooling instance-level predictions. They developed an adaptive pooling operation capable of interpolating between common pooling operations such as mean-, max- or min-pooling.

3. DATA CHALLENGE

In Sec. 2.1, we introduced research on instrument recognition in polyphonic, multi-timbral music. One theme that emerges is that with almost every new publication, a new dataset is released by the authors in an effort to address issues with previous ones. While releasing new datasets is highly encouraged and vital for research in MIR in general, an uncoordinated effort leads to lack of uniformity in the

datasets used. In this section we briefly describe the common datasets for instrument recognition and identify the challenges associated with them.

The IRMAS dataset [5] is a frequently used dataset for predominant instrument recognition. It consists of a separate training and testing set, each containing annotations for 11 predominant instruments. The dataset consists of short excerpts —3 s for training and variable length for testing— of weakly labeled data. One fundamental problem of the IRMAS annotations is that the training set lacks multi-label annotation; this can be problematic for a general use case as instrument co-occurrence is ignored.

The MedleyDB [4] and Mixing Secrets [11] datasets are both multi-track datasets. Due to the availability of instrument-specific stems, strong annotations of instrument activity are available. Thus, these two multi-track datasets provide all the necessary detailed annotations for instrument activity detection and have been used in [12, 25]. These datasets have two disadvantages when training models. First, with a few hundred distinct songs models trained with the data are hardly generalizable. Second, the datasets are not well balanced in terms of either musical genre or instrumentation. However, this may not be a problem if the datasets were larger and the distribution represented the real-world.

Most of these problems were addressed with the release of the OpenMIC dataset [17]. This dataset contains 20,000 10 s clips of audio from different songs across various genres. Each clip is annotated with the presence or absence of one or more of 20 instrument labels. OpenMIC presents a larger sample size as well as a uniform distribution across instruments. It is, however, weakly labeled, i.e., each 10 s clip has instrument presence or absence tags without specific onset and offset times. Due to the nature of weak labels, models cannot be trained using fine-grained instrument activity annotation as done, e.g., in [12, 19]. Additionally, not all clips are labeled with all 20 instruments, i.e., there are missing labels. This complicates the training procedure if models are to predict the presence/absence for all 20 instruments for an input audio clip. Despite their drawbacks, creation of WLDs scales better since weak labels are cheaper to obtain; models capable of exploiting WLDs may thus be vital for the future development of instrument recognition.

4. METHOD

Before describing the model details, we provide a formalization of our approach to the instrument recognition problem in weakly labeled data.

4.1 Pre-Processing

As mentioned in Sect. 3, the OpenMIC dataset consists of 10 s audio clips, each labeled with the presence or absence of one or more of 20 instrument labels. For each audio file in the dataset, the dataset creators also release features extracted from a pre-trained CNN, known as "VGGish" [16]. The VGGish model, based on the VGG architectures for object recognition [30], is trained for audio classification. The model produces a 128-dimensional feature vector for 0.96 s windows of audio with no overlap. The features are ZCA-whitened and quantized to 8-bits. For a 10 s audio file, we obtain a 10×128-dimensional matrix. We also normalize the 8-bit integers to a quantized range of $[0, 1]$.

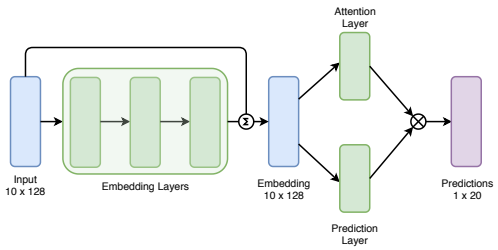

Figure 1: Model Architecture

4.2 Formulation

4.2.1 Multi-Instance Multi-Label Problem

In the most general setting, instrument recognition can be framed as Multi-Instance Multi-Label (MIML) classification [38, 42, 43]. Under this setting, we are given a training dataset $\{(\mathbf{X}_1, \mathbf{Y}_1), \dots (\mathbf{X}_m, \mathbf{Y}_m)\}$ where $\mathbf{X}_i$ is a bag containing r instances $X_i = \{\boldsymbol{x}_{i,1}, \dots \boldsymbol{x}_{i,r}\}$ and $\mathbf{Y}_i = [\boldsymbol{y}_{i,1}, \dots, \boldsymbol{y}_{i,L}] \in \{0, 1\}^L$ is a label vector with L labels with $\boldsymbol{y}_{i,j} = 1$ if any of the instances in $\mathbf{X}_i$ contains label j. In the remainder of this section, we will drop the indices used to reference a specific data point and simply represent a sample from the dataset as $(\mathbf{X}, \mathbf{Y})$. In our case, a bag $\mathbf{X}$ refers to the 10×128-dimensional feature matrix representing one audio clip and each bag contains 10 instances. Our problem is also a Missing Label problem since for a sample $(\mathbf{X}, \mathbf{Y})$, not all $\boldsymbol{y}_j$ are known or annotated (compare Sect. 3).

In our experiments, we assume that all labels can be independently predicted for each instance. Under this assumption, the MIML problem decomposes into L (20 for OpenMIC dataset) instantiations of Multi-Instance Learning (MIL) [8,41] problems, one for each label in the dataset.

Note that exploiting label-correlation in multi-label classification has shown to significantly improve the classification performance [28, 28, 34, 40]. However, exploring ways to incorporate label-correlation for instrument recognition in the OpenMIC dataset has the additional challenge of missing and sparse labels [3]. Also, as is prevalent in most MIL approaches [8], we assume independence among different instances in a bag. Neighboring instances in a bag representing a polyphonic music snippet will, however, likely have high correlation. Relaxing the aforementioned assumptions about independence among labels, and instances in a bag is left for future work since in our current work, we focus on the impact of attention for aggregating instance-level predictions.

4.2.2 Multi-Instance Learning

In the MIL setting, a bag label is produced through a score function $S(\mathbf{X})$. Under the assumption of independence

among instances, $S(\mathbf{X})$ admits a parametrization of the form

$$S(\mathbf{X}) = \mu\Big(f(\boldsymbol{x}) \Big) \tag{1}$$

where $f(.)$ is a score function for an instance $\boldsymbol{x}$, and $\mu(.)$ is a permutation-invariant aggregation operation for instance scores $f(\boldsymbol{x})$ [37]. This parameterization induces a natural approach to classify a bag of instances: (i) to produce scores for each instance in the bag using an instance-level scoring function $f(\boldsymbol{x})$, and (ii) to aggregate the scores across different instances in the bag using the aggregation function $\mu(.)$. In our approach, we use a classification function to produce instance-level scores $f(\boldsymbol{x})$, which are essentially the probabilities of a label being present for each instance. The max and avg functions are two commonly used permutation-invariant operations to aggregate instance-level scores to bag-level scores. McFee et al. found that *learning* an aggregation operation, however, significantly improved performance over fixed predefined operations like max and avg. We choose to represent our aggregation operation $\mu(.)$ as a weighted sum of instance-level scores, i.e.,

$$S(\mathbf{X}) = \sum_{\boldsymbol{x} \in \mathbf{X}} w_{\boldsymbol{x}} \; f(\boldsymbol{x}) \tag{2}$$

where $w_{\boldsymbol{x}}$ is a learnable weight for instance $\boldsymbol{x}$. Our choice of $f(.)$ and $\mu(.)$ has the two advantages that (i) the resulting $S(.)$ is the probability of a label being present in the bag and can be directly used to make a prediction and (ii) the learned weights for each instance add interpretability to the MIL models by encoding beliefs placed by the MIL model on the score of each instance.

4.2.3 Attention Mechanism

The learnable aggregation operation is equivalent to attention. Given a bag $\mathbf{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r\}$ of r instances, the instance level scoring function $f(.)$ produces a bag $\{f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_r)\}$ of instance scores. The bag-level score $S(\mathbf{X})$ is then computed using Eq. (2).

We further impose the restriction that instance weights w_x should sum to 1, i.e., $\sum_{\boldsymbol{x} \in \mathbf{X}} w_{\boldsymbol{x}} = 1$. This ensures that the aggregation operation is invariant to the size of the bag, thus allowing the model to work with sound clips of arbitrary length. Furthermore, this normalization leads to a probabilistic interpretation of the instance weights which can then be used to infer the relative contribution of each instance towards $S(\mathbf{X})$. For an instance $\boldsymbol{x} \in \mathbf{X}$, the weight $w_{\boldsymbol{x}}$ is thus parametrized as

$$w_{\boldsymbol{x}} = \frac{\sigma(\boldsymbol{v}^\top h(\boldsymbol{x}))}{\sum_{\boldsymbol{x}' \in \mathbf{X}} \sigma(\boldsymbol{v}^\top h(\boldsymbol{x}'))} \tag{3}$$

where $h(\boldsymbol{x})$ is a learned embedding of the instance $\boldsymbol{x}$, $\boldsymbol{v}$ are the learned parameters of the attention layer, and $\sigma(.)$ is the *sigmoid* non-linearity.

This corresponds to the attention mechanism traditionally used in sequence modeling [2, 35]. For example, Raffel and Ellis [29] produced attention weights in a manner similar to Eq. (3) with the only difference being the use of *softmax* operation to perform normalization of weights across the instances.

4.3 Model Architecture

Computing bag-level scores $S(.)$ involves computing instance-level scores $f(.)$ and aggregating the scores across instances using a learned set-operator $\mu(.)$ which performs weighted averaging with the weights computed with Eq. (3). For our experiments, we represent the scores, both instance level $f(.)$ and bag-level $S(.)$, as the probability estimate of the instance or bag being a positive sample for a given label. We first pass each instance $\boldsymbol{x}$ through an embedding network of three fully connected layers to project each instance to a suitable embedding space. Next, instance-level scores $f(.)$ are computed from the output of embedding network with another fully connected layer. Similarly, attention weights are computed by normalizing the outputs of a fully connected layer, the weights of which correspond to parameters $\boldsymbol{v}$ in Eq. (3). Note that the output dimension of these two parallel fully connected layers is equal to the number of labels, i.e., 20. Figure 1 illustrates the model architecture. In the embedding layer, the number of hidden units is 128. We also found that adding a skip connection from the input to the final embedding stabilized the training across different random seeds. We use batch normalization, ReLU activations, and a dropout of 0.6 after each embedding layer. The model has 55336 learnable parameters.

4.4 Loss Function and Training Procedure

Our model performs a multi-label classification over 20 labels given an input. However, as we point out earlier, the OpenMIC dataset does not contain all labels for each instance. This leads to missing ground truth labels for training with loss functions such as binary cross-entropy (BCE). To account for this, we utilize the partial binary cross-entropy (BCE_p) loss function introduced for handling missing labels [7]:

$$\begin{aligned} \mathrm{BCE}_p(\boldsymbol{y}, \boldsymbol{q}) &= \frac{g(p_y)}{L} \sum_{l \in L^o} \boldsymbol{y}_l \log \boldsymbol{q} + (1 - \boldsymbol{y}_l) \log(1 - \boldsymbol{q}) \\ g(p_y) &= \alpha p_y^\gamma + \beta \end{aligned} \tag{4}$$

Here $g(p_y)$ is a normalization function, p_y is the proportion of observed labels for the current data point, L is the total number of labels, L^o is the list of observed labels for the input data, $\boldsymbol{y}_l \in \{0, 1\}$ is the ground truth (absent or present) for label l, and $\boldsymbol{q}$ is the model's probability output for the label l being present in the input data $\mathbf{X}$. The hyperparameters in Eq. (4) are α, β, and γ. Note that in the absence of $g(p_y)$, data points with few observed labels will have a lower contribution in loss computation than those with several observed labels. This is undesirable behavior and the inclusion of a normalization factor, dependent on the proportion of observed labels, is important. Therefore, we set α, β, and γ to 1, 0, and -1, respectively. This normalizes the loss for a data point by the number of observed labels and is equivalent to only computing the loss for observed labels.

Finally, the Adam optimization algorithm [20] is used for training with a batch size of 128 and learning rate of $5e^{-4}$ for 250 epochs. We checkpoint the model at the epoch with the best validation loss.

5. EVALUATION

In this section we describe the experimental setup including the dataset, the baseline methods, and evaluation metrics.

5.1 Dataset

We use the OpenMIC dataset for the experiments in this paper. In addition to the audio and label annotations, the data repository contains pre-computed features extracted from the publicly available VGG-ish model for audio classification. We utilize those features in our experiments to strictly focus on handling the weak labels and avoid further complexity by having to learn features from the raw data or spectrogram representations. Pilot experiments for feature learning showed that CNN architectures based on state-of-the-art instrument recognition models were unable to outperform the baseline model of 20 instrument-wise random forest classifiers trained using the pre-computed features. For reproducibility and comparability, we utilize the training and testing split released with the dataset. Additionally, we randomly sample and separate 15% data from the training split to create a validation set.

5.2 Experiments

We compare the attention model (ATT) with the following models:

1. RF_BR: This model is the baseline random forest model in [17]. A binary-relevance transformation is applied to convert the multi-label classification task into 20 independent binary classification tasks [39].
2. FC: A 3-layer fully connected network trained to predict the presence or absence of all instruments for a given data instance. Here, the input features of dimension 10×128 are flattened into a single feature vector for classification. Dropout is used for regularization and the Leaky ReLU (0.01 slope) is used. The model has 986772 parameters.
3. FC_T: This model serves as an ablation study to observe the benefits of the attention mechanism. FC_T uses the same embedding layer as ATT. However, the aggregation of predictions in time is simply performed with average-pooling. The model has 52116 parameters.
4. RNN: A 3-layer bi-directional gated recurrent unit model with 64 hidden units per direction. The model processes the input features and produces a single embedding which is then fed to a classifier for all 20 instruments. The model has 226068 parameters.

Source code for the Pytorch implementation of the neural network models is publicly available. [1] For each model, we train 10 randomly initialized instances with different random seeds and compute the classification metrics for each.

[1] https://github.com/SiddGururani/AttentionMIC

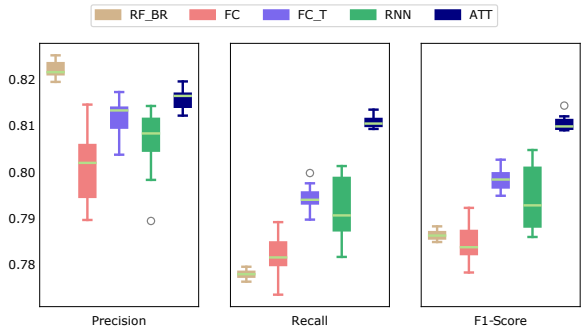

Figure 2: Precision, recall, and F1-score for different models

This gives us a distribution of each model's performance. One benefit of ATT over the FC and RNN models is its small size. Both the ATT and FC_T utilize weight-sharing for embedding instances from the bags. This leads to significantly fewer learnable parameters compared to FC and RNN while performing better than both of these models.

5.3 Metrics

While the total number of clips per instrument label in the OpenMIC dataset is balanced, the number of positive and negative examples is not well balanced for each instrument label. Therefore, we separately compute the precision, recall and F1-score for the positive and negative class. Thereafter, we compute the macro-average of these metrics to report the final instrument-wise metrics, meaning that positive and negative examples are weighted equally. We call these the instrument-wise precision, recall, and F1-score. Additionally, to measure the overall performance of a classifier, we macro-average the instrument-wise precision, recall and F1-score. We use a fixed threshold of 0.5 to convert the outputs into binary predictions for computing the classification metrics.

6. RESULTS AND DISCUSSION

Figure 2 shows the overall performance of ATT compared to the baseline models with box plots for the macro-averaged precision, recall, and F1-score. Additionally, we compare the instrument-wise F1-score for each model in Figure 3. Note that we only show the mean instrument-wise F1-score across 10 seeds in Figure 3 for improved visibility.

We observe that while the attention mechanism does not lead to an improvement in precision compared to the other models, the recall is improved significantly and consequently the F1-score is also improved. We also observe that ATT performs better than RF_BR in almost every instrument label, especially for the labels with high positive-negative class imbalance, such as clarinet, flute, and organ. This ties to the observation made about improved recall, as ATT is able to overcome this imbalance possibly due to the ability to localize the relevant instances for the minority class. In the case of an imbalanced instrument label, the recall for the minority class greatly suffers for RF_BR.

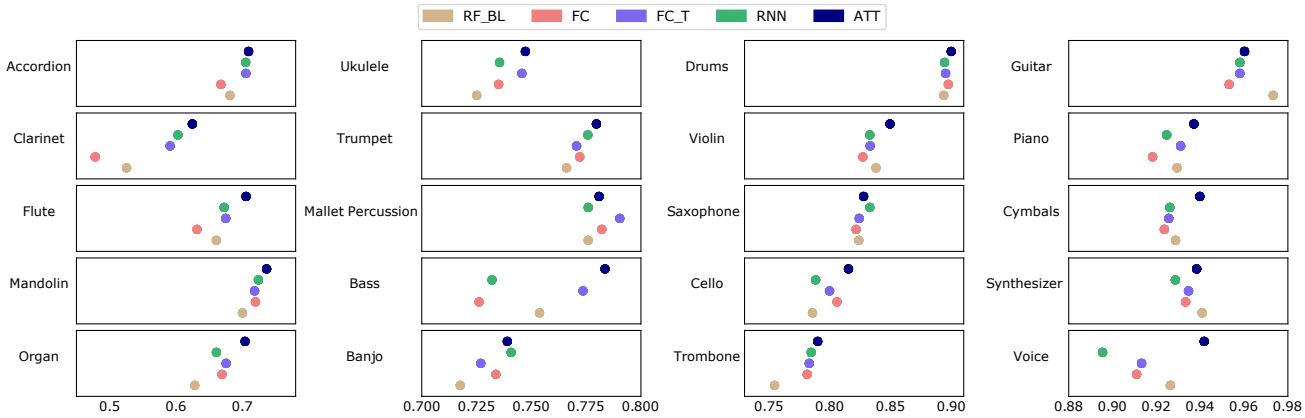

Figure 3: Instrument-wise F1-scores

While this problem is easily mitigated in standard multi-class problems by using balanced sampling, it is difficult to address with multi-label data. Comparing to FC_T, we can attribute the better performance of ATT to better aggregation of instance-level predictions. FC_T is essentially the same model as ATT using mean pooling instead of attention, and ATT outperforms it for most instrument classes, especially the generally more difficult to classify instruments. The RNN model also beats the RF_BR baseline. In polyphonic music, the instances in a bag are structured and highly correlated and hence using a recurrent network to model the temporal structure in the instance sequence leads to a powerful embedding of the bag, incorporating useful information from each instance.

We visualize the attention weights for two example clips in Figure 4. The left clip is from the test set and starts with the vocals fading out until 2 seconds. From 5 second onwards, the vocals grow in loudness until the end of the clip. The violin plays throughout but is the pre-dominant instrument only for a few seconds between 3 and 6 seconds, as visualized in the corresponding attention weights as well. The right clip is from the training set and contains vocals starting from 6 second onwards. The attention weights for vocals directly coincides with that. It is interesting to note that the annotation for vocals was missing for this clip.

7. CONCLUSION

Weakly labeled datasets for instrument recognition in polyphonic music are easier to develop or annotate than strongly labeled datasets. This calls for a paradigm shift in the approaches towards supervised learning approaches better suited for weakly labeled data. We formulate the instrument recognition task as a MIML problem and introduce an attention-based model, evaluated on the OpenMIC dataset for 20 instruments, and compared against several other baseline models including: (i) binary-relevance random forest, (ii) fully connected networks, and (iii) recurrent neural networks, We find that the attention mechanism improves the overall performance as well as the instrument-wise performance of the model while keeping the model light-weight. The example visualizations show that the model indeed is

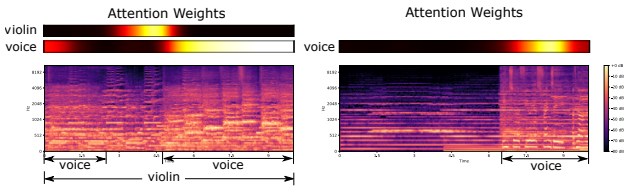

Figure 4: Attention Weight Visualization: The horizontal bars above the mel-spectrogram represent the attention weights across the instances of the clip for the respective instruments.

able to attend to relevant sections on a clip.

Some of the assumptions made in the formulation of the MIML problem are strong and may be worth relaxing due to the nature of musical data. We plan to further explore the task of instance-level embeddings using recurrent networks or using self-attention mechanisms as used in Transformer networks [35]. Additionally, we plan to address the problem of missing labels or label sparsity in the OpenMIC dataset using the curriculum learning-based methods proposed in [7]. Our concern is that the dataset is not large enough with enough labels for strictly supervised learning approaches to significantly improve the results much further than what we achieve with the attention mechanism, and we therefore plan to tackle the problem from other angles, such as handling missing labels or data augmentation.

8. ACKNOWLEDGEMENTS

This research is partially funded by Gracenote, Inc. We thank them for their generous support and meaningful discussions. We also thank Nvidia Corporation for their donation of a Titan V awarded as part of the GPU grant program.

9. REFERENCES

[1] Sharath Adavanne, Pasi Pertilä, and Tuomas Virtanen. Sound event detection using spatial features and convolutional recurrent neural network. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 771–775, New Orleans, LA, USA, 2017.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. of the International Conference on Learning Representations, (ICLR)*, San Diego, CA, USA, 2015.

[3] Wei Bi and James Kwok. Multilabel classification with label correlations and missing labels. In *Proc. of the AAAI Conference on Artificial Intelligence*, pages 1680–1686, Québec City, Québec, Canada, 2014.

[4] Rachel M Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Pablo Bello. Medleydb: A multitrack dataset for annotation-intensive MIR research. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 155–160, Taipei, Taiwan, 2014.

[5] Juan J Bosch, Jordi Janer, Ferdinand Fuhrmann, and Perfecto Herrera. A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 559–564, Porto, Portugal, 2012.

[6] Emre Cakır, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 25(6):1291–1303, 2017.

[7] Thibaut Durand, Nazanin Mehrasa, and Greg Mori. Learning a deep convnet for multi-label classification with partial labels. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 647–657, Long Beach, CA, USA, 2019.

[8] James Foulds and Eibe Frank. A review of multi-instance learning assumptions. *The Knowledge Engineering Review*, 25(1):1–25, 2010.

[9] Ferdinand Fuhrmann. *Automatic musical instrument recognition from polyphonic music audio signals*. PhD thesis, Universitat Pompeu Fabra, 2012.

[10] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780, New Orleans, LA, USA, 2017.

[11] Siddharth Gururani and Alexander Lerch. Mixing secrets: A multi-track dataset for instrument detection in polyphonic music. In *Late Breaking Demo (Extended Abstract), Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, Suzhou, China, 2017.

[12] Siddharth Gururani, Cameron Summers, and Alexander Lerch. Instrument activity detection in polyphonic music using deep neural networks. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 569–576, Paris, France, 2018.

[13] Yoonchang Han, Jaehun Kim, and Kyogu Lee. Deep convolutional neural networks for predominant instrument recognition in polyphonic music. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 25(1):208–221, 2017.

[14] Yoonchang Han, Subin Lee, Juhan Nam, and Kyogu Lee. Sparse feature learning for instrument identification: Effects of sampling and pooling methods. *The Journal of the Acoustical Society of America*, 139(5):2290–2298, 2016.

[15] Perfecto Herrera-Boyer, Geoffroy Peeters, and Shlomo Dubnov. Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1):3–21, 2003.

[16] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. CNN architectures for large-scale audio classification. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, USA, 2017.

[17] Eric Humphrey, Simon Durand, and Brian McFee. Openmic-2018: An open dataset for multiple instrument recognition. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 438–444, Paris, France, 2018.

[18] Yun-Ning Hung, Yi-An Chen, and Yi-Hsuan Yang. Multitask learning for frame-level instrument recognition. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 381–385, Brighton, UK, 2019.

[19] Yun-Ning Hung and Yi-Hsuan Yang. Frame-level instrument recognition by timbre and pitch. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 135–142, Paris, France, 2018.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of the International Conference on Learning Representations, (ICLR)*, San Diego, CA, USA, 2015.

[21] Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G Okuno. Instrument identification in polyphonic music: Feature weighting to minimize influence of sound overlaps. *EURASIP Journal on Applied Signal Processing*, 2007(1):155–155, 2007.

[22] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark Plumbley. Audio set classification with attention model: A probabilistic perspective. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 316–320, Calgary, Canada, 2018.

[23] Qiuqiang Kong, Changsong Yu, Turab Iqbal, Yong Xu, Wenwu Wang, and Mark D. Plumbley. Weakly labelled audioset classification with attention neural networks. *CoRR*, abs/1903.00765, 2019.

[24] Anurag Kumar and Bhiksha Raj. Audio event detection using weakly labeled data. In *Proc. of the 24th ACM International Conference on Multimedia (ACMMM)*, pages 1038–1047, Amsterdam, The Netherlands, 2016.

[25] Peter Li, Jiyuan Qian, and Tian Wang. Automatic instrument recognition in polyphonic music using convolutional neural networks. *CoRR*, abs/1511.05520, 2015.

[26] Vincent Lostanlen, Joakim Andén, and Mathieu Lagrange. Extended Playing Techniques: The Next Milestone in Musical Instrument Recognition. In *Proc. of the International Conference on Digital Libraries for Musicology (DLfM)*, pages 1–10, Paris, France, 2018.

[27] Brian McFee, Justin Salamon, and Juan Pablo Bello. Adaptive pooling operators for weakly labeled sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 26(11):2180–2193, 2018.

[28] Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, Tao Mei, and Hong-Jiang Zhang. Correlative multi-label video annotation. In *Proc. of the ACM International Conference on Multimedia (ACMMM)*, pages 17–26, Augsburg, Germany, 2007.

[29] Colin Raffel and Daniel P. W. Ellis. Feed-forward networks with attention can solve some long-term memory problems. *CoRR*, abs/1512.08756, 2015.

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of the International Conference on Learning Representations, (ICLR)*, San Diego, CA, USA, 2015.

[31] Dan Stowell, Dimitrios Giannoulis, Emmanouil Benetos, Mathieu Lagrange, and Mark D. Plumbley. Detection and classification of acoustic scenes and events. *IEEE Transactions on Multimedia*, 17(10):1733–1746, 2015.

[32] Takumi Takahashi, Satoru Fukayama, and Masataka Goto. Instrudive: A Music Visualization System Based on Automatically Recognized Instrumentation. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 561–568, Paris, France, 2018.

[33] John Thickstun, Zaïd Harchaoui, and Sham Kakade. Learning features of music from scratch. In *Proc. of the International Conference on Learning Representations, (ICLR)*, Toulon, France, 2017.

[34] Konstantinos Trohidis, Grigorios Tsoumakas, George Kalliris, and Ioannis P Vlahavas. Multi-label classification of music into emotions. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 325–330, Philadelphia, PA, USA, 2008.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008. Curran Associates, Inc., Long Beach, CA, USA, 2017.

[36] Chih-Wei Wu and Alexander Lerch. From labeled to unlabeled data – on the data challenge in automatic drum transcription. In *Proc. of the International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018.

[37] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan R Salakhutdinov, and Alexander J Smola. Deep sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3391–3401. Curran Associates, Inc., Long Beach, CA, USA, 2017.

[38] Zheng-Jun Zha, Xian-Sheng Hua, Tao Mei, Jingdong Wang, Guo-Jun Qi, and Zengfu Wang. Joint multi-label multi-instance learning for image classification. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, Anchorage, AK, USA, 2008.

[39] Min-Ling Zhang, Yu-Kun Li, Xu-Ying Liu, and Xin Geng. Binary relevance for multi-label learning: an overview. *Frontiers of Computer Science*, 12(2):191–202, 2018.

[40] Min-Ling Zhang and Kun Zhang. Multi-label learning by exploiting label dependency. In *Proc. of the ACM International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD)*, pages 999–1008, Washington, DC, USA, 2010.

[41] Zhi-Hua Zhou and Min-Ling Zhang. Neural networks for multi-instance learning. In *Proc. of the International Conference on Intelligent Information Technology*, pages 455–459, Beijing, China, 2002.

[42] Zhi-Hua Zhou and Min-Ling Zhang. Multi-instance multi-label learning with application to scene classification. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1609–1616. Curran Associates, Inc., Vancouver, BC, Canada, 2007.

[43] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291 – 2320, 2012.