Multi-Task Learning for Interpretable Weakly Labelled Sound Event Detection
Soham Deshmukh, Bhiksha Raj, Rita Singh

Abstract—Weakly Labelled learning has garnered lot of attention in recent years due to its potential to scale Sound Event Detection (SED). The paper proposes a Multi-Task Learning (MTL) framework for learning from Weakly Labelled Audio data which encompasses the traditional Multiple Instance Learning (MIL) setup. The MTL framework uses two-step attention mechanism and reconstructs Time Frequency (T-F) representation of audio as the auxiliary task. By breaking the attention into two steps, the network retains better time level information without compromising classification performance. The auxiliary task uses an auto-encoder structure to encourage the network for retaining source specific information. This indirectly de-noises internal T-F representation and improves classification performance under noisy recordings. For evaluation of proposed methodology, we remix the DCASE 2019 task 1 acoustic scene data with DCASE 2018 Task 2 sounds event data under 0, 10 and 20 dB SNR. The proposed network outperforms existing benchmark models over all SNRs, specifically 22.3 %, 12.8 %, 5.9 % improvement over benchmark model on 0, 10 and 20 dB SNR respectively. The results and ablation study performed demonstrates the usefulness of auto-encoder for auxiliary task and verifies that the output of decoder portion provides a cleaned Time Frequency (T-F) representation of audio/sources which can be further used for source separation. The code is publicly released.

Index Terms—sound event detection, multi task learning, deep neural networks, attention, auto encoder

I. INTRODUCTION

The goal of Sound Event Detection (SED) is to determine the presence, nature and temporal location of sound events in audio signals. This is usually accomplished using machine learning and signal processing algorithms, and is of great added benefit to applications like wearable devices [1], mobile robots [2] and public security [3].

Many SED algorithms rely on strongly labelled data [4] [5] [6] for training in order to perform accurate event detection. Here the term strongly labelled refers to audio in which events are annotated with their corresponding onset and offset time. Usually, only the audio segments within the onset-offset time boundaries are used as training data, while segments outside the annotated onset and offset time boundaries are considered to be non-target events.

However, producing strongly labelled data for SED is quite expensive in terms of the expertise, time and human resources required for the purpose. Annotation outcomes are often restricted to minutes for every few hours [6] of actual data and time spent listening to it.

A solution to limited strongly label data is to utilise the large amount of data being generated on multimedia sites. Utilising publicly available multimedia data comes with its own set of unique problems, one of them being the nature of annotation. The annotation is generally in the form of weak information like titles, tags, description etc. The term 'weak' information or label refers to information which simply indicates presence or absense of particular events in the video or audio and does not provide any information with respect to number of audio events, duration of events and time localisation of events in the audio or video. Weakly-supervised learning for sound events [7] mitigates this problem to a great extent, by allowing for the use of weakly labelled data to train SED algorithms designed for this. Weakly Labelled Data (WLD) contains event labels at an audio clip level without including any annotations pertaining to the onset and offset times for the events in the clip. This type of data is relatively easy to produce, and there are currently several datasets such as [8], [9] etc. which continue to be used for weakly-supervised training for SED algorithms.

Thus, weakly labelled data has garnered significant attention in recent times, and is considered to be promising in scaling up the scope of SED in real-life applications. Generally, SED based on Weakly labelled data (WLSED) is formulated as a Multiple-Instance Learning (MIL) problem [10]. Multiple-instance learning is a form of supervised learning in which the labels are available for a 'bag' of instances and not for individual instances.

In MIL setup for weakly labelled audio, an audio clip is considered as 'bag' of instances, where the individual instances are the segments of the audio clip [7]. This formulation provides two fold benefit. First, it helps in learning the clip level mapping to audio event using weakly labelled data and second it provides a time-level localisation of the audio events in the clip.

In neural MIL formulation, the first half of network produces temporal predictions which are then aggregated by the second half of network usually a pooling operator to produce audio clip level predictions. The temporal predictions are generally generated by encoding audio using a Convolution Neural Networks (CNN) [11] based architecture. This is followed by a global max pooling (GMP), global average pooling (GAP) operator to get audio clip level predictions. Recent papers have developed different pooling mechanism like Adaptive pooling [12], Global Weighted Rank Pooling (GWRP) [13]. Attention pooling [14] where the aim is to make the pooling...
Section IV contains the details about the developed dataset, feature extraction and network details. Section V contains the Weakly labelled SED results under different SNR, Ablation test results and End to End interpretable visualization. Section VI contains future work and directions to build on this work, followed by Conclusion in Section VII.

II. PROBLEM SETUP AND RELATED WORK

Weakly labelled learning in context of SED is generally formulated as Multiple Instance Learning (MIL) [10] problem where a single binary class label is assigned to a collection (bag) of similar training examples (instances). WLD is used when relative labels are present or where obtaining strong labels is too expensive. The overall SED setup used for WLD is shown in Fig. 1.

A. MIL formulation for SED

In case of SED, let’s say the audio is consisting of multiple frames totalling up to T frames. Let the raw audio be \( \{x_i\}_{i=1}^T \). The feature extracted from raw audio \( \{x_i\}_{i=1}^T \) can be represented by \( \{\hat{x}_i\}_{i=1}^T \). The extracted features constitute a bag \( B = \{\hat{x}_i\}_{i=1}^T \). MIL assumption states that the weak labels of bag \( B \) are \( y = \max_i \{y_i\}_{i=1}^T \), where \( y_i \) is the strong label corresponding to feature \( \hat{x}_i \). Thus Weakly labelled data’s training pair consists of \( \{B, y\} \). The networks used to approximate MIL framework for SED can be divided into two sequential networks. The first network models a function \( g_1 \) which generates a T-F segmentation mask. Ideally, the output generated by this network is supposed to be T-F segmentation masks for each class. The paper will refer to the first network as ‘Segmentation network’. This is followed by a second function \( g_2 \) which learns a mapping between T-F segmentation mask to probabilities of corresponding sound tags. The output of second network are audio clip level class probabilities. As the second networks role is to perform classification of T-F representation into appropriate classes, the paper will refer to this second network as ‘Classification Network’.

B. Segmentation Network

For learning segmentation map, the initial work proposed CNN based architecture [15] for the task. SVM max margin formulation was also used to solve WLD MIL [15] [16]. To better approximate the MIL framework, [17] [18] used a time-distributed CNN with a global max pool to pick out location...
of relevant temporal events. The input data consists of log-scale Mel spectrograms. The combination of Log-scale Mel spectrograms and CNNs have been previously successfully used for sound classification [19] [20] [21]. Convolutional Recurrent Neural Network (CRNN) have been explored in [22] for weakly labelled audio classification. First the network made of CNN layers operate on the input data consists of log Mel spectrogram to extract relevant source level features. This is followed by a Bidirectional Recurrent Neural Network (Bi-RNN) to capture the temporal context information between frames in T-F representation. Recent work proposed Atrous convolutions in SED framework [14], which concludes the size of receptive field is more important than Acoustics scene classification than using local pooling to fix size of feature maps. This paper address the challenges of generating better class wise T-F representation, by demonising the internal learned T-F representations by an auto-encoder based auxiliary task.

C. Classification Network

For classification map, pooling layer such as global max pooling (GMP), global average pooling (GAP) [23], global weighted rank pooling (GWRP) [13], global attention pooling [24], [14] or other poolings [25] [26] [12], fully connected layers are applied to predict the presence probabilities of audio classes. The global max pooling is known to under predict the sound events as the operator only takes into account the most prominent feature ignoring others. To combat this [13] used GWRP, instead of max pooling. GWRP can be seen as an extension of both GAP and GMP. The GWRP operator converts to GMP when \( r = 0 \) and GAP when \( r = 1 \), where \( r \) is a hyperparameter that varies according to frequency of occurrence of sound events. However, global pooling can only predict the time domain segmentation, but not the T-F segmentation. To make pooling model more flexible and aware to each contributing location in T-F output, attention mechanism was proposed [24] [14]. Attention mechanism is more flexible and weighs each contributing location in T-F output to form the final predictions. However, training using attention over entire \((C \times T \times F)\) is unstable with traditional Binary Cross Entropy loss. Both the challenges are addressed by two-step attention mechanism proposed in the paper.

D. Multi-Task Learning

Multi-Task Learning has been associated with many other names, some of the common ones are joint learning, learning to learn, and learning with auxiliary tasks. Multi-task learning is a type of inductive transfer. For MTL, this inductive bias generally takes the form of an auxiliary task which forces the network to learn representation to jointly solve more than one task. This generally leads to solutions that generalise better [27] and is shown to work across many application of machine learning natural language processing, speech recognition and computer vision [28] [29] [30].

In audio domain, MLT has been recently applied to jointly learn features for multiple speech classification tasks: speaker identification, emotion classification, and automatic speech recognition [31] [32]. The above cited work learns representation to solve all the downstream task equally well. Our work employs a different formulation, where rather than multiple tasks, we employ an auxiliary task which doesn’t require additional data and the dummy task’s performance is not of interest. The key is selecting appropriate tasks as the auxiliary task for Multi-Task Learning, if incorrectly selected the auxiliary task can hurt the performance of primary task. To the best of our knowledge, this is the first work which formulates Multi-Task Learning for Weakly labelled SED data and determines appropriate auxiliary task for the same.

III. PROPOSED METHODOLOGY

This section contains the details of Multi-Task Learning formulation for SED, its corresponding segmentation mapping network \( g_1 \), classification mapping network \( g_2 \), and the auxiliary T-F reconstruction auto-encoder task. This paper’s Multi-Task Learning formulation is depicted in Fig 2.

A. Multi-Task Learning SED formulation

In this section we will formulate the Multi-Task-Learning formulation starting from MIL formulation for weakly labelled SED. Let the raw audio be represented as \( X = \{x_i\}_{i=1}^T \) where \( x_i \in \mathbb{R} \) and \( i \) indicates the specific frame in \( \mathbb{R} \) of total frames \( T \). Features are extracted from raw audio and brought to a T-F form: \( \hat{X} = \{\hat{x}_i\}_{i=1}^T \) where \( \hat{x}_i \in \mathbb{R}^m \) where \( m \) is dimension of feature used to represent a frame of audio. Every bag of frames \( B = \{\hat{x}_i\}_{i=1}^T \) has a weak label \( y \in \mathbb{R}^K \). The weakly labelled training data in terms of bags is:

\[
B_j = (\{\hat{x}_i\}_{i=1}^T, y)_{j=0}^N
\]

where \( N \) are the number of examples.

The primary task in our setup of Multi-task learning framework is SED. The first part of network for SED task is to learn a segmentation mapping \( g_1(\cdot) \) where:

\[
g_1 : \hat{X} \mapsto Z
\]

The segmentation network maps the feature set \( \{\hat{x}_i\}_{i=1}^T \) to \( Z = \{z_i\}_{i=1}^T \) where \( z_i \in \mathbb{R}^{K \times F} \) and \( K \) is the number of audio events. The second part of SED task is network which classifies \( \{z_i\}_{i=1}^T \) to \( P = \{p_i\}_{i=1}^K \) where \( P \in \mathbb{R}^K \). The network learns a mapping:

\[
g_2 : Z \mapsto P
\]

where \( g_2 \) maps each classes T-F segmentation to their presence probabilities of \( k^{th} \) event known as \( p_k \).

The auxiliary task in MTL setup is reconstruction of T-F representation. For reconstruction, an autoencoder structure will be used. The first part of autoencoder is encoder network which compacts data into bottleneck features. The aim of bottleneck features is such that they allow reconstruction of data with minimal error. The encoder learns a function \( g_3(\cdot) \) where \( g_3 : \hat{X} \mapsto Z \).
We make an assumption that audio’s T-F representation \( \{\hat{x}_i\}_{i=1}^T \) can be completely explained as a linear or non-linear combination of each audio event’s independent T-F representation. This implies

\[
\{\hat{x}_i\}_{i=1}^T = \sum_{k=0}^{K} W_k \{z_i\}_{i=1}^T
\]  

(4)

where \( i \) ranges from 1 to \( T \) and \( \hat{x}_i \in R^{m} \), \( z_i \in R^{K \times F} \). This assumption allows the network to have a shared encoder for both SED task and auxiliary reconstruction. From now on we will represent \( g_1(.) = g_3(.) = g(.) \) as the shared segmentation mapping function, such that \( g : X \mapsto Z \). The shared encoder performs compaction in the number of filter and keeps the T-F dimensions untouched.

The second part of auxiliary task is a decoder network which learns a mapping \( g_4 \) such that \( g_4 : Z \mapsto \hat{X} \) where \( \hat{X} \) is the reconstructed T-F representation. Specifically:

\[
\{\hat{x}_i\}_{i=1}^T = g_4(\{z_i\}_{i=1}^T)
\]  

(5)

Here the mapping function \( g_4(.) \) is ideally such that:

\[
g_4^{-1}(g(.)) = g^{-1}(g_4(.)) = I
\]  

(6)

To force the above inverse relation in auxiliary task, the network Loss for that task be \( L_2 \) to minimise the difference between true T-F representation \( \{\hat{x}_i\}_{i=1}^T \) and predicted T-F representation \( \{\hat{x}_i\}_{i=1}^T \) of audio.

To solve for the SED task, the network should learn \( g(.) \) shared mapping such that the mask \( \{z_i\}_{i=1}^T \) should accurately segment each audio event and classification mapping \( g_3(.) \) should map it to correct audio event. Let the loss to enforce these conditions for primary task be \( L_1 \).

The functions \( g(\cdot), g_2(\cdot), g_4(\cdot) \) will be expressed using Neural Networks. In terms of Neural Network terminology, this is equivalent to learning a weights \( W = [w, w_2, w_4] \) where \( w, w_2, w_4 \) are weights corresponding to each function \( g(\cdot), g_2(\cdot), g_4(\cdot) \) respectively. The optimisation problem can be framed in terms of these weights \( W \) over all data points as:

\[
\min_W L_1(P, y|w, w_4) + \alpha L_2(\{\hat{x}_i\}_{i=1}^T, \{\hat{x}_i\}_{i=1}^T|w, w_2)
\]  

(7)

The two-component loss function will be refered as \( L(W) \). Parameter alpha (\( \alpha \)) accounts for scale difference between losses \( L_1 \) and \( L_2 \). It also helps in adjusting weightage given to auxiliary task relative to primary task to guide learning of weights.

B. Shared Segmentation Network

The Segmentation mapping here processes the audio clip and converts it into a processed T-F (Time Frequency) representation for each class. The audio clip \( X_i \) is first converted into a T-F representation \( \hat{X} = \{\hat{x}_i\}_{i=1}^T \). Here \( \hat{X} = \{\hat{x}_i\}_{i=1}^T \) is spectrogram or log Mel spectrogram. CNN based network is used here for modelling mapping \( g(.) \). The CNN based network has multiple CNN blocks. Each CNN block consists of three subparts. In first part, features \( \hat{X} = \{\hat{x}_i\}_{i=1}^T \) is processed using 2 repeating units of 2-dimensional linear convolution, Batch Normalisation [33] and ReLU [34] non-linearity. This is followed by 2-dimensional Average pooling layer with appropriate stride and padding to main original T-F dimensions of audio. Batch Normalisation helps to stabilise training by reducing the layer\(\hat{\Delta}Z\)s internal covariate shift and recent work suggests it makes the optimisation landscape significantly smoother which results in the stable behaviour of
gradients \[35\]. This 2 unit CNN block is repeated to form the segmentation mapping section of the network.

The segmentation network also acts as the encoder of the auto-encoder framework and has to jointly encode features relevant for audio classification and audio reconstruction. Having a common encoder force the network to learn a shared representation, helps the network to exploit commonalities and differences across SED and T-F reconstruction and enables the network to generalise better on our original task. The Multi-Task Learning (MLT) setup using an auxiliary task to introduce an inductive bias, which will cause the network to prefer solutions that generalise better \[36\]. This will result in improved learning and predictive power for SED. The MLT setup used here is a hard parameter sharing instead of soft parameter sharing which greatly reduces the risk of overfitting \[37\].

C. Classification Network

The modelling choice for classification mapping results in different intermediate T-F representation of audio. The traditional choices for modelling classification mapping are Global Max Pooling \[23\], Global Average Pooling \[38\] and Global Weighted Rank Pooling \[39\] which results in compromise between time level precision and classification performance. We use a two-step attention process to covert segmentation mapping \{\( z_i \)\}_{i=1}^{T} \) into audio level predictions \( T \). By adding \( \sigma \) in attention output it squashes the output between 0 and 1 and makes sure the output of attention doesn’t go above 1 which results in stable training. The performance comparison between traditional attention pooling and 2-step attention is done in results section. Loss used here is Binary Cross Entropy to compute error between predicted probability of audio events \( P \) and true weak labels \( y \).

D. Decoder Network

The decoder of the auxiliary-task takes \( Z = \{ z_i \}_{i=1}^{T} \) as input and reconstructs it to \( \{ \hat{x}_i \}_{i=1}^{T} \). CNN based network is used for down-sampling number of filters of CNN from \( K \) (Number of classes). The CNN based network has multiple CNN blocks. The architecture closely follows the common encoder structure except the number of filters are decreased from one block to another. Each CNN block consists of three subparts. In first part, features \( \tilde{X} = \{ \tilde{x}_i \}_{i=1}^{T} \) is processed using 2 repeating units of 2-dimensional linear convolution, Batch Normalisation \[33\] and ReLU \[34\] nonlinearity. This is followed by 2-dimensional Average pooling layer with appropriate stride and padding to main original T-F dimensions of audio. Loss used here is Mean Squared Error (MSE) to compute error between T-F representation \( \{ \hat{x}_i \}_{i=1}^{T} \) and reconstructed T-F representation \( \{ \hat{x}_i \}_{i=1}^{T} \).

IV. Experiments

A. Dataset

The dataset is made by mixing DCASE 2019 Task 1 of Acoustic scene classification \[40\] and DCASE 2018 Task 2.
The DCASE 2018 task 2 provides annotated audio clips associated to one of the 41 events like ‘Flute’, ‘Gunshot’ and ‘Bus’ from Google Audioset Ontology [8]. DCASE 2019 contains 10 sec clips from 10 different scenes like âAirportâ, âMetro stationâ and âUrban parkâ among others. From DCASE 2018 task 2, two second clips are extracted and mixed with 10 second background noise from DCASE 2019 task 1 for SNR 0 dB, 10 dB and 20 dB. Each audio clip contains three non-overlapping audio events. For each SNR, the 8000 mixed audio clips are divided into 4 cross-validation folds. The mixed audio clips are single channel with a sampling rate of 32 kHz and the 2 second clips are extracted and mixed with 10 second background noise from DCASE 2019 task 1 for SNR 0 dB, 10 dB and 20 dB.

### TABLE III

| Layers                                    | Output size (C × T × F) |
|-------------------------------------------|-------------------------|
| Internal T-F representation               | 256 × 311 × 64          |
| Conv2D: [K = 3, C = 128], BN, ReLU × 2   | 128 × 311 × 64          |
| 2D Average pooling [K = 3, S = 1, P = 1]  | 128 × 311 × 64          |
| Conv2D: [K = 3, C = 64], BN, ReLU × 2    | 64 × 311 × 64           |
| 2D Average pooling [K = 3, S = 1, P = 1]  | 64 × 311 × 64           |
| Conv2D: [K = 3, C = 32], BN, ReLU × 2    | 32 × 311 × 64           |
| 2D Average pooling [K = 3, S = 1, P = 1]  | 32 × 311 × 64           |
| Conv2D: [K = 3, C = 1], BN, ReLU × 2     | 311 × 64                |

B. Evaluation Metric

The metric used to evaluate the proposed framework against benchmark networks is Area Under the Receiver Operating Characteristic Curve (ROC AUC) [42], micro Precision and macro Precision. The metrics are chosen such that they are threshold independent and characterises the performance of network in both balanced and imbalace class settings. The metrics are defined as follows:

1) **ROC AUC:** A receiver operating characteristic curve plots true positive rate (TPR) vs false positive rate (FPR). The area under the ROC curve is computed which summaries the ROC AUC curve. Using the AUC does not require manual selection of a threshold. Bigger AUC indicates better performance. A random guess has an AUC of 0.5

2) **Precision:** The precision is intuitively the ability of the classifier not to label as positive a sample that is negative.

\[ P = \frac{TP}{TP + FP} \]

C. Feature Extraction

The raw data is converted to T-F representation by applying a FFT with a window size of 2048 and overlap of 1024 between consecutive windows. This is followed by applying mel filter banks with 64 bands and converting it to log scale to obtain Log mel spectrogram. This is used as input to the shared segmentation mapping network and is found to work well with neural networks [13], [44].

D. Network

This subsection shared segmentation, classification and decoder network’s detailed description along the training details. The segmentation network details are shown in Table [I]. It takes log mel spectrogram as input and employs CNN block structure similar to VGG [45]. The smallest unit of the network consists of CNN layer of kernel size 3, Batch Normalisation and ReLU sequentially be called CNN sub-block. This sub-block is repeated twice and is followed by 2D Average Pooling of stride 1 and padding 1 to form a CNN block. Empirically, 2D Max pooling and 2D Average Pooling showed similar performance and therefore 2D Average Pooling was arbitrarily chosen. The CNN block is repeated 4 times with increasing channel size (32, 64, 128, 256). This input is passed to class convolution block as shown in Fig. [2] to reduce it down to number to classes. Note that the Decoder network takes the input of class convolution as it’s input and not the output of class convolution. This is determined empirically, as the performance of both is comparable and considering the input of class convolution saves an additional convolution block which would be necessary to upscale the class T-F representation in the decoder network.

The classification network (Table [II]) consists of 2 step attention pooling to summarize and reduce the class T-F representation to get audio clip level predictions. The 2-step attention pooling is described in Section [III-C]. The Decoder network (Table [III]) emulates the encoder network with decrease in the number of channels to reduce it to log mel spectrogram of input. Using the terminology of sub-block and block, the decoder consists of 3 CNN-block with decreasing channel size (256, 128, 64, 32). This is followed by a reversed class convolution block consisting of one CNN block with only one channel to reduce the T-F representation to log mel spectrogram.
| Networks   | SNR 20 dB micro-P | SNR 20 dB macro-P | SNR 20 dB AUC | SNR 10 dB micro-P | SNR 10 dB macro-P | SNR 10 dB AUC | SNR 0 dB micro-P | SNR 0 dB macro-P | SNR 0 dB AUC |
|-----------|-------------------|-------------------|---------------|-------------------|-------------------|---------------|----------------|----------------|-------------|
| GAP       | 0.5067            | 0.6127            | 0.9338        | 0.4291            | 0.5390            | 0.5390        | 0.3295         | 0.4093         | 0.8694      |
| GMP       | 0.5390            | 0.5186            | 0.8497        | 0.5263            | 0.5023            | 0.8422        | 0.4640         | 0.4441         | 0.8189      |
| GWRP [13] | 0.7018            | 0.7522            | 0.9362        | 0.6538            | 0.7129            | 0.9265        | 0.5285         | 0.6084         | 0.8985      |
| Atrous AP [14] | 0.7391 | 0.7586          | 0.9279        | 0.6740            | 0.7404            | 0.9211        | 0.5714         | 0.6341         | 0.9014      |
| 2APAE     | 0.7829            | 0.7645            | 0.9390        | 0.7603            | 0.7486            | 0.9343        | 0.6986         | 0.6892         | 0.9177      |

The entire network is trained end-to-end with a batch size of 24 and learning rate of 1e-3 using Adam optimiser [46] on 4 GPU cards with 12 GB RAM each.

V. RESULTS

A. Sound Event Detection Results

The proposed architecture is compared with traditional methods and current benchmark architecture for weakly labelled sound event detection specifically GMP, GAP, GWRP [13]. Attention pooling with Atrous convolution [14]. The results are shown in Table IV and are cross-validation scores across 4 folds. The important metric here is Micro Precision (micro-P), as it calculates metrics globally by counting the total true positives, false negatives and false positives. This is a better indicator of network performance as it takes into account class imbalance rather than simple unweighted averaging that macro Precision does. 2APAE (2 step Attention Auto Encoder) has micro-Precision score of 0.7829, 0.7603 and 0.6986 on SNR 20, 10 and 0 dB respectively. The results show that 2APAE network achieves the best score across all SNR’s across all metrics. In terms of micro-Precision, 2APAE outperforms existing benchmark of Atrous AP (Atrous Attention Pooling [14]) on SNR 20, 10 and 0 dB by 5.9%, 12.8% and 22.3% respectively. The second benefit of breaking the attention into two steps apart from improving performance, is to provide stable training.

The attention pooling used with atrous convolution resulted in overflow issues during training, where the final predictions probabilities when over 1 for some audio events or classes. This doesn’t fit well with Binary Cross entropy as loss function which expects the input probabilities between 0 to 1. Empirically, adding a squashing function after attention hampers learning. By breaking the attention into two steps, it allows for the intermediate use of sigmoid which helps in ensuring the outputs don’t explode above 1.

B. Ablation Study

To determine the contribution of 2-step Attention and decoder on the SED performance, ablation study is performed. As described in Section III-A, the total loss is:

\[ L = L_1(P, y|w, w_1) + \alpha L_2(\{\hat{x}_i\}_{i=1}^T, \{\hat{x}_i\}_{i=1}^T|w, w_2) \]  

The value of \( \alpha \) determines the contribution of auxiliary task to the weakly labelled SED. By varying the value of alpha, we can determine the performance improvement by 2-step attention (\( \alpha = 0.0 \)) and contribution of auxiliary task (\( \alpha > 0 \)). The alpha values are kept low as the \( L_2 \) loss term (Mean Squared Error) is magnitude greater than \( L_1 \) loss. The lower \( \alpha \) value helps in adjusting for this large scale difference between the two losses.

When \( \alpha = 0.0 \), the network has no contribution from the auxiliary task and can be used to evaluate the performance of 2-step attention. In terms of micro-Precision, the 2-step attention mechanism outperforms existing benchmark of Atrous AP (Atrous Attention Pooling [14] from Table V on SNR 20, 10 and 0 dB by 5.2%, 10.2% and 21.4% respectively. By adding the auxiliary task contribution with a relative weightage of \( \alpha = 0.001 \), compared to proposed 2-step attention, an improvement of 0.7%, 2.3% and 0.7% is observed. The numbers indicate that 2-step attention has heavy contribution on the improvement of performance, with extra performance gains from auxiliary task.

When \( \alpha \) is increased to 0.01, the performance compared to \( \alpha = 0.001 \) is decreased. This indicates that the auxiliary task’s loss contribution starts to overpower the primary SED task’s loss contribution rather than improving generalisation. The shared encoder then learns features more relevant to auxiliary task rather than primary SED task.

C. Interpretable visualisation

Using 2-step attention makes it easy to visualise the contribution of each Mel bin and each time frame in the T-F representation to each audio event and final predictions.
Fig. 3. Visualisation of 2APAE Attention and Decoder results. The figure has total 7 subplots each visualising a step of the 2APAE process. Subplot 1 depicts the scaled log mel spectrogram input to network. Subplot 2 is the decoder output. Subplot 3 and 4 are attention weights for two prominent classes in the prediction. Subplot 5 is the output of 1\textsuperscript{st} step attention. Subplot 6 and 7 is the attention weight and output of 2\textsuperscript{nd} step attention respectively. The y-axis in subplot 1 - 4 is Mel-bins and Number of Sound Events in subplot 5-6. The x-axis is time in subplot 1-6 and Number of Sound Events in subplot 7.
Before learning attention weights generally a class-dependent threshold is applied \[13\] 47 to remove False positive and fake activations. But this post-processed varies from network to network and doesn’t paint an accurate picture of contribution of each component in attention as it becomes highly dependent on threshold chosen and adds author’s bias for choosing of threshold. Hence in this paper we plot the raw attention weights without applying any thresholds.

Fig. \[\text{3}\] gives the end to end visualisation of the how the input is processed along what part attention focuses on when SNR is 20 dB. The first sub-figure is the scaled log mel spectrogram that is input to the network. From the first input log-mel subplot, we can determine that there are three audio events happening at three distinct time. The third and fourth sub-figure are 1st step attention’s weight for two top probable sound events ‘Violin’ and ‘Cello’ where the y-axis is mel bins and x-axis is time. As visible, the attention is giving higher probability to time-aligned portion of classes with particular mel bins given higher (brighter color) weightage. The fifth subplot is the output of 1st step attention over mel bins where y-axis is particular audio events or classes and x-axis is time. As visible the attention output is forming a mask similar where the audio events are present giving importance to particular classes. The sixth subplot visualises the 2nd step attention weights where y-axis is classes and x-axis is time. There are four distinct lines on the sixth subplot, corresponding to 4 audio events in time, where 1 event is false activation. The audio events are not perfectly time-aligned. This is due to the use of 2D-Average pooling in the shared encoder, which reduces the T-F representation per two convolution layers. The addition of 2D average pooling improves weakly labelled SED performance, but results in losing time-level precision. The seventh subplot is the output of 2nd step attention over time where y-axis is the prediction probability and x-axis is the audio event or class label.

The subplot two is the output of the decoder used as auxiliary task. The goal of the auxiliary task and mainly the decoder is to reconstruct the log mel spectrogram from the shared encoder’s output. From the figure, it can be seen that the decoder is not also able to reconstruct the audio events clearly, it is also denoising the log mel spectrogram in 20 dB SNR. For the decoder to perform well, the internal T-F representation has to be denoised as well. From sub-figure two things can be inferred:

- In Multi-Task Learning framework, adding the reconstruction auxiliary task to primary weakly labelled SED leads to denoising of the internal T-F and the encoder representation of log mel spectrogram. This is backed by ablation study’s result in Section \[\text{V-B}\] which shows adding reconstruction auxiliary task improves performance. That improved performance can be attributed to this denoising of internal T-F representations.
- The denoised log mel representation, will help in improving the performance of audio source separation systems. This setup either pretrained or jointly trained with source separation, can be used to produce cleaned audio log mel representation for better source separation.

**VI. FUTURE WORK AND DIRECTIONS**

The paper shows, Weakly labelled SED can be viewed from the lens of Multi-Task Learning where Multiple Instance Learning SED is the primary task coupled with an constructive auxiliary task. This results in interesting directions for extending and building upon this work:

- Exploring different auxiliary task to help Weakly labelled SED detection. There can be single or multiple auxiliary task which might help creating better shared segmentation mapping for audio events.
- Jointly training source sound separation along with weakly labelled SED in MTL framework
- Developing better loss function and dynamic \(\alpha\) determination to properly utilise multiple audio task in MTL framework and improve performance

**VII. CONCLUSION**

The paper proposes a Multi-Task learning formulation for Weakly Labelled Sound Event Detection. In MTL framework, it proposes a two-step attention mechanism and reconstructs Time Frequency (T-F) representation of audio as the auxiliary task. By breaking the attention into two steps, the network retains better time level information without compromising classification performance under all SNR as shown in the result Section \[\text{V-A}\] and visualisation Section \[\text{V-C}\]. The proposed network outperforms existing benchmark models over all SNRs, specifically 22.3 %, 12.8 %, 5.9 % improvement over benchmark model 0, 10 and 20 dB SNR respectively. By using the T-F reconstruction as auxiliary task, it encourages the shared segmentation mapping network or encoder to retain source specific information and indirectly de-noises internal T-F representation. This results in improved performance in noisy environments as shown by the ablation study results in Section \[\text{V-B}\]. For evaluation of proposed methodology, we remix the DCASE 2019 task 1 acoustic scene data with DCASE 2018 Task 2 sounds event data under 0, 10 and 20 db SNR.

**REFERENCES**

[1] Y. Xu, W. J. Li, and K. K. Lee, *Intelligent Wearable Interfaces*. USA: Wiley-Interscience, 2008.
[2] S. Chu, M. Mataric, C. Kuo, and S. Narayanan, “Where am i? scene recognition for mobile robots using audio features,” in 2006 IEEE International Conference on Multimedia and Expo. Los Alamitos, CA, USA: IEEE Computer Society, jul 2006, pp. 885–888. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICME.2006.2622661
[3] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti, “Scream and gunshot detection and localization for audio-surveillance systems,” in 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, 2007, pp. 21–26.
[4] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumley, “Detection and classification of acoustic scenes and events,” IEEE Transactions on Multimedia, vol. 17, no. 10, pp. 1733–1746, 2015.
[5] E. ĀGakāisr, G. Parascandolo, T. Heititol, H. Huttunen, and T. Virtanen, “Convolutional recurrent neural networks for polyphonic sound event detection,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 6, pp. 1291–1303, 2017.
[6] A. Mesaros, T. Heittola, and T. Virtanen, “Tut database for acoustic scene classification and sound event detection,” in 2016 24th European Signal Processing Conference (EUSIPCO), 2016, pp. 1128–1132.
