

Pakhi Rajput

INCREMENTAL LEARNING FOR AUDIO CLASSIFICATION WITH PRE-TRAINED MODELS

Bachelor's Thesis
Faculty of Engineering and Natural Sciences
Manjunath Mulimani
April 2nd 2025

ABSTRACT

Pakhi Rajput: Incremental Learning for Audio Classification with Pre-Trained Models
Bachelor of Science Thesis
Tampere University
Computing and Electrical Engineering
April 2nd 2025

Deep learning models have long been used for audio classification tasks, but when it comes to training for incremental tasks, they frequently forget what they have already learned. Catastrophic forgetting is a phenomenon that makes incremental learning tasks more difficult. In order to minimize the substantial loss of previously learned information, this bachelor's thesis investigates the use of pre-trained models, such as PANNs, for incremental learning in audio classification. The effectiveness of pre-trained models for audio classification in incremental learning is first reviewed in the thesis. In order to assess the models, it then creates a series of classification tasks using the ESC-50 dataset. The trade-offs between knowledge retention and task adaptability are examined using the results. Performance degradation on previously learned tasks persists even though this method reduces catastrophic forgetting when compared to traditional methods. The results point to possible directions for further study and aid in the development of more effective incremental learning techniques for audio classification applications.

Keywords: deep learning, audio classification, catastrophic forgetting, incremental learning, PANNs, ESC-50

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

PREFACE

I would like to thank my thesis supervisor, Manjunath Mulimani, for providing me with this topic and guiding me through my research and thesis writing process. His insightful suggestions, constant encouragement and deep understanding of the subject helped in shaping the direction and quality of my work.

Tampere, Finland, 2nd April, 2025

Pakhi Rajput

CONTENTS

1. INTRODUCTION	1
2. LITERATURE REVIEW.....	2
2.1 Incremental Learning	2
2.2 Audio Classification.....	3
2.3 Convolutional Neural Networks (CNNs)	4
2.4 Pre-trained Models for Audio Classification.....	5
2.5 Related Work	6
3. METHODOLOGY.....	7
3.1 Dataset Preparation	7
3.2 Pre-trained Model Selection.....	7
3.2.1 PANNs and CNN14 Model Selection	7
3.2.2 PANNs CNN14 Model Architecture.....	8
3.2.3 Incremental Learning PANNs CNN14 Model Architecture.....	8
3.3 Incremental Learning Setup	9
3.3.1 Implementation Details.....	9
3.3.2 Loss Functions.....	10
3.4 Evaluation Metrics.....	12
4. RESULT AND DISCUSSION	14
4.1 Baseline Performance.....	14
4.2 Incremental Learning Performance	15
4.3 Catastrophic Forgetting Analysis.....	16
5. CONCLUSION	18
REFERENCES.....	19

LIST OF FIGURES

<i>Figure 1</i> An example CNN model architecture	4
<i>Figure 2</i> Loss and Accuracy Curves for Baseline Performance.....	14
<i>Figure 3</i> Accuracy Per Task for Incremental Learning.....	15
<i>Figure 4</i> Plot for Catastrophic Forgetting Value Per Task	17

LIST OF SYMBOLS AND ABBREVIATIONS

PANNs	Pre-trained Artificial Neural Networks
mAP	Mean Average Precision
AUC-ROC	Area Under the Receiver Operating Characteristic Curve
EWC	Elastic Weight Consolidation
LwF	Learning without Forgetting
CNNs	Convolutional Neural Networks
RNNs	Recurrent Neural Networks
FIM	Fisher Information Matrix
ER	Experience Replay
ReLU	Rectified Linear Unit
KD	Knowledge Distillation

1. INTRODUCTION

Deep learning models have been used extensively for audio classification tasks in recent years due to its ability to learn complex patterns compared to traditional approaches. However, they still struggle with incremental learning tasks where the model needs to learn new tasks sequentially. One of the major issues faced in this process is catastrophic forgetting, where the retention of old data deteriorates as it learns new data over time.

This thesis will be focusing on the use of pre-trained audio classification models, here PANNs, for incremental learning task. Using pre-trained models, which have been trained on extensive audio datasets, require little fine-tuning to adapt to new tasks and help in reducing time and computing power. While these pre-trained models have been effective in static classification, their performance with incremental learning tasks is still in question.

The research will begin with splitting the ESC-50 dataset into sequential tasks for incremental learning. It will then examine strategies like weight regularization, memory replay, and knowledge distillation to reduce the effects of catastrophic forgetting. The study will assess the model's effectiveness by using metrics like accuracy, F1 Score, Mean Average Precision (mAP), AUC, and catastrophic forgetting. The ability of such pre-trained models in learning new audio categories without noticeably forgetting previously learned ones is assessed.

Chapter 2 will present some background about audio classification and incremental learning, while exploring some previous studies done in the field. Chapter 3 will expand on the experimental setup and the methodology used for optimizing results. Chapter 4 will present the experimental results and analysis of the metrics. Chapter 5 will conclude the discussion by summarizing the results and providing insights to possible future studies.

2. LITERATURE REVIEW

This chapter will elaborate on the key concepts of this research: incremental learning, audio classification, pre-trained models and related work done in the field. Some of the methods used to avoid catastrophic forgetting which are discussed in the incremental learning section include Elastic Weight Consolidation (EWC), Learning without Forgetting (Knowledge Distillation), and Memory Replay. The Audio Classification section discusses deep learning models like CNNs, RNNs, and Transformers. In terms of pre-trained models, audio classification models like PANNs, VGGish, and YAMNet will be discussed. The selection of the pre-trained model for this study and the dataset used will also be explained briefly.

2.1 Incremental Learning

Incremental learning is an approach where a model learns new tasks sequentially while retaining information from previously learned tasks. Traditional deep learning models like CNNs are trained in a single batch. Incremental learning aims to adapt continuously as new tasks keep adding over the time [1].

Neural Networks learn by adjusting their weights based on recently learned data. This creates the challenge of catastrophic forgetting when it comes to incremental learning since the old knowledge is often overwritten [2]. Several methods have been proposed to counter this problem of catastrophic forgetting.

One of the methods is called Elastic Weight Consolidation (EWC), which estimates the importance of each model parameter in previous task using the Fisher Information Matrix (FIM). After identifying such parameters, it introduces a penalty to limit the change in these important weights [2]. This penalty is added as a regularization term to the loss function.

Another method called Learning Without Forgetting (LwF) helps in reducing the effects of catastrophic forgetting. It makes use of knowledge distillation, which means that the model stores the original predictions (soft targets) for the previous task before training on a new task. Training on new task introduces a knowledge distillation loss which ensures that the model's output on new task remains similar to previous outputs [3]. This loss is calculated by training both, the teacher model and the new model, on samples

from previous task and using the outputs to calculate a difference. This allows the model to retain past knowledge while learning new information.

A Reply-based method commonly used for continual learning is Experience Replay (ER) which stores past experiences (or training sample) in a memory buffer to replay periodically during training. The model samples from both new data and memory buffer during training on new task to retain previous knowledge [4].

This research makes use of these methods to mitigate the problem of catastrophic forgetting. Elastic Weight Consolidation helps in preserving layers that are sensitive to task-specific patterns and prevents overwriting them. Knowledge Distillation retains previous knowledge of learned tasks without storing all their data by forcing the student to mimic the outputs of the teacher model. Experience Replay helps in maintaining class balance as new classes are introduced over time by periodically introducing previous samples while training on new tasks.

2.2 Audio Classification

Audio Classification is the process of analysing and classifying audio signals into different categories. Virtual assistants, environmental sound classification applications, and music genre identification are some of the areas that make use of audio classification. Traditional audio classification models such as Support Vector Machines (SVM), K-Nearest Neighbour (KNN), and Hidden Markov models (HMMs) made use of feature extraction (Mel-Frequency Cepstral Coefficients (MFCCs), Spectrograms, and Wavelet Transforms) to classify audio. Deep learning models provide better accuracies because of their ability to understand complex patterns. Unlike traditional methods, deep learning models can automatically extract high-dimensional features from large audio datasets, using spectrograms or waveforms, bypassing the need for manual feature extraction [5].

Some of the deep learning models used for audio classification include Convolutional Neural Networks, Recurrent Neural Networks, and Transformers. These are discussed shortly below.

Recurrent Neural Networks (RNNs) are very commonly used in speech recognition and Natural Language Processing (NLP) tasks. They are designed to handle sequential data by connecting the current state of the network output to a combination of the network input and to previous states of the network [5]. Convolutional Neural Networks (CNNs) make use of three different types of layers: convolution layer, pooling layer, and fully

connected layer. In terms of audio classification task, the convolution layer learns frequency-time patterns in the spectrogram, pooling layer reduces dimensionality, and fully connected layer converts learned features into class predictions. Transformers are deep learning models which are commonly used for NLP tasks. They transform an input sequence into an output sequence by learning context and tracking relationships between sequence components.

This study focuses more on CNNs since they can capture spatial and hierarchal patterns in spectrograms which are commonly used to represent audio signals. The basic CNN architecture is briefly explained in the next section.

2.3 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) have become increasingly popular deep learning models, specifically in classification and detection tasks. They are ideal for audio classification tasks because of their ability to capture local and hierarchical patterns in spectrograms, which are often used to represent audio signals in 2D format.

The basic architecture of a CNN consists of an input layer, convolution layers, pooling layers, fully connected layer and an output layer. An example CNN model is shown in Figure 1 below.

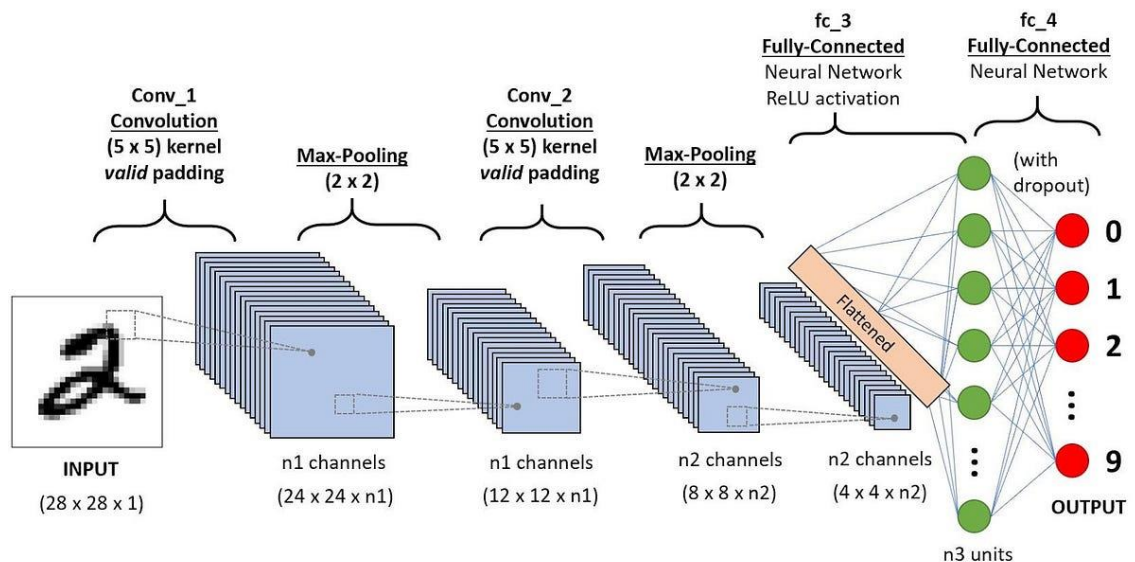


Figure 1 An example CNN model architecture [14]

The model shown in Figure 1 takes an image as an input and applies convolution and max pooling layers on it before flattening it into a fully connected layer and finally the output layer. All these layers are explained below.

The input layer takes the input data for the whole model usually in the form of a 2D matrix. The convolution layers then apply a filter on the input and create convoluted features. The filter is a matrix of weights which runs over the input and the dot products of the filter weights and input values are calculated to produce feature maps. These filter weights are determined during training by using back propagation and gradient descent. The pooling layer is used to reduce dimensionality because of the high number of parameters, and broaden the receptive field, which is a region of the input matrix a neuron is responsive to. Most common types of pooling are max pooling and average pooling [11].

Once all the features are learned, the data in the feature maps is flattened to a 1-dimension vector and passed to a fully connected dense layer for decision making. The fully connected layers map the flattened feature maps to the output layer by using an activation function [11].

Activation functions are used in both convolution layers and fully connected layers so that the model can learn complex patterns in the data. Some of the most common activation functions are ReLU (Rectified Linear Unit), Sigmoid, and Tanh. Dropout and batch normalization are also used to prevent overfitting and stabilize the training process.

2.4 Pre-trained Models for Audio Classification

Instead of training models from scratch for new tasks, the use of pre-trained models is steadily increasing since they are already trained for feature detection on large-scale datasets. This approach is known as transfer learning where a pre-trained model is reused on a new problem [6]. It is less computational and time consuming to fine-tune such pre-trained models.

Some of the most widely used pre-trained models include PANNs (Pre-trained Audio Neural Networks), VGGish and YAMNet. These are discussed below briefly.

PANNs are CNN-based architectures trained on AudioSet. They are developed for large-scale audio classification and have been successfully adapted for various audio tasks [7]. VGGish is a CNN model based on VGG architecture which comprises of a 24-layer-

deep network, and YAMNet is also a CNN model which is based on MobileNet-v1 architecture comprising of a 28-layer-deep network [8].

This study uses PANNs model since it trained on a large dataset (AudioSet) covering various sounds and is ideal for extracting features from new tasks. PANNs supports various architectures like CNNs, ResNet, and MobileNet, and can be easily adapted for incremental learning on multiple audio domains.

2.5 Related Work

Studies in deep learning are increasingly exploring incremental learning especially in tasks where the model needs to adapt continuously to new data without forgetting the previously learned data. Many studies cover class-incremental learning for multi-label audio classification aiming to learn new audio classes independently without interfering with previously learned ones [9].

Studies have also been done to address the issue of catastrophic forgetting commonly faced in continual learning. Karam et al. (2023) address this issue by proposing an episodic memory technique, facilitating both forward and backward knowledge transfer, which helps in learning new tasks efficiently without compromising performance on the previously learned ones [10]. This approach helps in mitigating catastrophic forgetting by maintaining a proper representation of previous and new samples.

Other studies also discuss more strategic memory-based methods like herding, as proposed in iCaRL (Incremental Classifier and Representation Learning), to use for sampling in memory buffers [13]. This method retains the most representative samples from previous class which are replayed periodically and help with overcoming the problem of forgetting during incremental learning.

Recent advancements have also explored the use of pre-trained models in audio classification tasks. Notably, the study done by Qiuqiang Kong et al. (2021) explore the use of PANNs for Audio Pattern Recognition. It discusses various CNN architectures, ResNets, MobileNets, one-dimensional CNNs too. Out of these, a 14-layer CNN achieves a mean average precision of 0.431 [7].

For the purpose of this study, we will explore a modified version of this CNN14 PANNs model and extend it for the incremental learning task on ESC-50 dataset.

3. METHODOLOGY

This section discusses the experimental setup, data preparation, model selection, and the approach taken to implement incremental learning using PANNs model on ESC-50 dataset for audio classification.

3.1 Dataset Preparation

This study focuses on ESC-50 dataset which is a collection of 2000 environmental sounds recorded at 44.1 kHz. Each recording is 5 second long and these are organized into 50 semantical classes. Each class has 40 examples, and these classes are loosely arranged into 5 categories (Animals, Natural soundscapes & water sounds, human non-speech sounds, interior/domestic sounds, and exterior/ urban noises). This audio dataset gives fair representation of different classes which ensure balanced training.

The dataset is divided into sequential tasks to create an incremental learning environment, where new sound categories are introduced over time. This study uses class-based incremental learning and splits the 50 classes into 5 tasks, each containing 10 classes. Proper class balance between train and test datasets for classes in each task has been maintained by using Stratified K-Fold method for proportional sampling.

3.2 Pre-trained Model Selection

3.2.1 PANNs and CNN14 Model Selection

Pre-trained models are models which are already trained on large datasets and can be generalized well to other similar task with little to no modifications. This method of transfer learning (reusing trained model with fine-tuning for a separate task) is cost efficient and reduces the computational load significantly.

This study explores PANNs (Pre-trained Audio Neural Networks) models which are designed to handle large-scale audio datasets. These models are trained on AudioSet, which contains over 2 million labelled audio clips consisting of 630 sound classes, out of which 527 classes are used for training these PANNs models. Their model architecture is mainly based on Convolutional Neural Networks (CNNs) which perform well with

classification and detection tasks. Since these models are already trained on a large-scale dataset, it is easier to utilize them for different audio classification tasks.

For the purpose of this study, we have chosen CNN14 architecture from the PANNs models since it has shown to be one of the most accurate models for audio classification tasks. They have shown high accuracy when trained on the AudioSet dataset. This high accuracy is likely to transfer when working with smaller datasets like ESC-50. The CNN14 model is capable of understanding temporal and spectral patterns which is essential for differentiating between different audio classes. This model creates a good balance between complexity and computational efficiency compared to other CNN models. This model is also ideal for incremental learning since it can handle new tasks without compromising previously learned tasks, thus mitigating catastrophic forgetting.

3.2.2 PANNs CNN14 Model Architecture

The PANNs CNN14 Model is designed for large-scale audio classification tasks. Its architecture consists of an input layer, 14 convolutional layers, fully connected layers, and finally the output layer. These are discussed briefly below.

The input layer takes the audio spectrogram (Log-Mel Spectrogram) representations as input. This is followed by 6 convolution blocks, each consisting of two convolution layers (with kernel size 3x3), batch normalization and ReLU activation. Batch normalization helps in reducing variance and improve performance. The ReLU (Rectified Linear Unit) activation function, which introduces non-linearity, is used since it is computationally cheap and helps with vanishing gradient problems. These convolution blocks extract features such as edges in frequencies, energy fluctuations, harmonics, tonal and percussive sounds, frequency modulations, etc.

Pooling operation is done after each convolution block to reduce the dimensionality. The fully connected layer uses the sigmoid activation function for multi-label classification of 527 AudioSet classes. Finally, the output layer consists of 527 neurons, each representing the probability of one audio class.

3.2.3 Incremental Learning PANNs CNN14 Model Architecture

The basic architecture and feature extraction is adapted from the original PANNs model, but some modifications are made in the fully connected and output layer for the incremental learning task in this study.

The fully connected layer is modified to use Log-Softmax activation function for single label (multi-class) classification tasks as opposed to the Sigmoid activation function used in original PANNs model for multi-label classification tasks. The output layer is also modified to output custom number of neurons (in this case 50, since ESC-50 dataset has 50 classes), instead of the 527 neurons in the original model (for AudioSet dataset).

3.3 Incremental Learning Setup

Apart from the changes made to the model architecture, certain more modifications are required to create an incremental learning environment.

3.3.1 Implementation Details

The classifier layer (fully connected) is expanded to add new classes dynamically without forgetting previously learned ones to ensure proper incremental learning.

The model adapts the original PANNs CNN14 model by freezing the pre-trained base layers to retain the general-purpose audio representations learned and then implementing a gradual unfreezing strategy to facilitate efficient learning over sequential tasks. This is a good practice especially when working with smaller datasets like ESC-50 where overfitting is a common phenomenon and thus it becomes essential to ensure proper incremental training process.

Training during the first task is limited to the uppermost convolution block and the modified fully connected classification head. The convolution blocks are then gradually unfrozen beginning from the highest level (closest to the classifier) and moving incrementally towards the lower layers based on the model performance. If the performance of the model stagnates and the improvement falls below a certain threshold value (here 0.01), then the convolution block next in line is unfrozen. This evaluation is done after every third epoch, again to facilitate smooth learning and avoid overfitting. Training on each new task does not begin from the topmost frozen configuration, instead it resumes from the exact state in which the model was left at the end of the previous task. This approach allows the model to progressively adapt to the new tasks without having to train the model from scratch.

A frozen copy of the previous model is also saved as a teacher network which generates soft targets (probability distributions) over previous classes. A knowledge distillation loss is calculated for the predictions made by the new model on the old classes. This process

of knowledge distillation guides the student model (new model) to give similar outputs as the teacher model while learning new tasks.

Elastic Weight Consolidation (EWC) is also incorporated into the training process for incremental learning. After the training in a task has been done, the parameters learned so far are stored and the Fisher Information Matrix (FIM) is computed. A EWC loss is added to the total loss function which penalizes any changes made to important parameters from previous task. This prevents losing critical knowledge and allows long-term retention over tasks.

An Experience Replay (ER) mechanism is also implemented by adding a buffer to store previously learned data (here, 10 samples per class). This prior knowledge is sampled periodically after every alternate epoch to remind the model of the previously learned data while training on new data. In this study, centroid-based herding is done where samples are selected based on their distance from the mean (centroid) of the class vector. Top k samples with the shortest distance are selected. This method tries to maintain class balance by selecting the most relevant samples from each class based on their distance from the centroid of the class vector.

Each task is trained using 30 epochs and an early stopping mechanism is also implemented where the evaluation on a task is ended at an early stage if the change in accuracy is very less ($\delta < 0.01$) and the average accuracy over three previous tasks is above a threshold value (90%). This optimizes the training process and avoids overfitting with too much training.

3.3.2 Loss Functions

A combination of Cross Entropy Loss, Knowledge Distillation (KD) Loss and Elastic Weight Consolidation (EWC) Loss is used for the optimization process.

Cross-Entropy Loss measures the difference between the true label distribution and predicted probability distribution using the formula shown in Equation 1 below:

$$L_{CE} = - \sum_{i=1}^c y_i \log(\hat{y}_i) \quad (1)$$

where, y_i is true one-hot encoded label for class i , and \hat{y}_i is the predicted softmax probability for class i . This loss is used for training on new classes during incremental learning.

Knowledge Distillation (KD) Loss measures the predictions of the new model on old classes and learns to mimic the previously trained model soft predictions using the formula shown in Equation 2 below:

$$L_{KD} = \sum_i q_i^T \log \left(\frac{q_i^T}{p_i^S} \right) \quad (2)$$

where, q^T is the teacher model's soft predictions, and p^S is the student model's soft predictions. This helps in incremental learning by training the new model to give similar predictions to the teacher model.

Elastic Weight Consolidation (EWC) Loss prevents any drastic changes in important parameters by introducing a penalty as shown in Equation 3 below:

$$L_{EWC} = \frac{1}{2} \sum_i F_i (\theta_i - \theta_i^*)^2 \quad (3)$$

where, θ_i is the current value of the parameter i , θ_i^* is the value of the parameter i after learning previous tasks, and F_i is the Fisher Information Matrix. This helps in knowledge retention by penalizing shifts in important weights.

The final loss is a combination of all three losses which is shown in Equation 4 below:

$$L_{total} = L_{CE} + \alpha L_{KD} + \lambda L_{EWC} \quad (4)$$

where, α controls the influence of knowledge distillation, and λ is a hyperparameter that controls how strong the EWC Loss penalty is. This total loss along with Experience Replay facilitates efficient training of the model for incremental learning.

3.4 Evaluation Metrics

To evaluate the performance of the incremental learning approach used in this study, metrics like Accuracy, F1 score, Mean Average Precision (mAP), Area Under the Curve (AUC), and Forgetting Measure are evaluated. These metrics help in analysing the overall performance of audio classification and also test the effectiveness of the incremental learning algorithm used. These metrics are discussed in brief below.

The accuracy score measures the overall classification performance over all classes by using the proportion of correctly predicted samples over total samples. The formula for the same is shown in Equation 5 below:

$$Accuracy = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i = y_i) \quad (5)$$

where, N is the number of samples, \hat{y}_i is the predicted class label, and y_i is the true class label.

The F1 Score is a harmonic mean of precision and recall which is calculated separately for each class. This is very useful for detecting class imbalance effects. F1 score is calculated according to the formula shown in Equation 6 below:

$$F1 = \frac{1}{N} \sum_{i=1}^N \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (6)$$

where, N is the total number of classes, and Precision and Recall are shown in Equation 7 and Equation 8 respectively.

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

where, TP is true positives, FP is false positives and FN is false negatives for class C .

The Mean Average Precision (mAP) is particularly helpful in multi-class classification as it is a trade-off between precision and recall across thresholds and measures the mean of the average precision scores for each class. This is shown in Equation 9 below:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (9)$$

where, N is the total number of classes, and AP is the value of Average Precision for each class. This value of AP is calculated from the precision-recall curve by integrating precision values over recall levels.

The Area Under the Curve (AUC) measure how well the model is able to distinguish between classes by using one-vs-rest comparison strategy (treating one class as positive and others as negative) for multi-class settings. It calculates the area under the Receiver Operation Characteristic (ROC) curve, which plots the True Positive Rate (TPR) against the False Positive Rate (FPR). A higher value of AUC indicates better separability between the predicted probabilities of the true and false classes.

The Forgetting measure helps in keeping a track of the performance degradation on previously learned tasks after training on new class. Forgetting for each task is found as the difference between the maximum accuracy of task t over all tasks and accuracy of task t after training on the final task. This is shown in Equation 10 below:

$$Forgetting = \max(Acc_t^k) - Acc_t^T \quad (10)$$

where, Acc_t^k is the accuracy of task t after training the model on task k and Acc_t^T is the accuracy of task t after training the model on the final task T . A positive value of forgetting indicates a decline in performance on old tasks after training on new ones which helps in assessing the continual learning setup.

4. RESULT AND DISCUSSION

In this section, first we analyse the baseline performance of the pre-trained model without incremental learning. Then the model is trained incrementally by dividing the dataset into 5 sequential tasks with 10 classes each. The catastrophic forgetting for incremental learning is also analysed. The dataset used for testing these performances is ESC-50.

4.1 Baseline Performance

Before starting the incremental learning process, the model is first tested on the ESC-50 dataset without any incremental learning setup. The model is trained on all 50 classes together and the accuracy and losses are tracked after each epoch (iteration).

Figure 2 below shows the curves for test accuracies and losses plotted against each epoch, tracking the performance of the model. The loss starts at a high value of 3.9004 but goes down gradually and reaches 3.0569 by the end of training. The accuracy on the other hand begins at a low value of 11.25 percent, as expected, and slowly grows to a good accuracy of 89.25 percent. The value of train accuracies and test accuracies also remain similar throughout the training process, so there is no major overfitting.

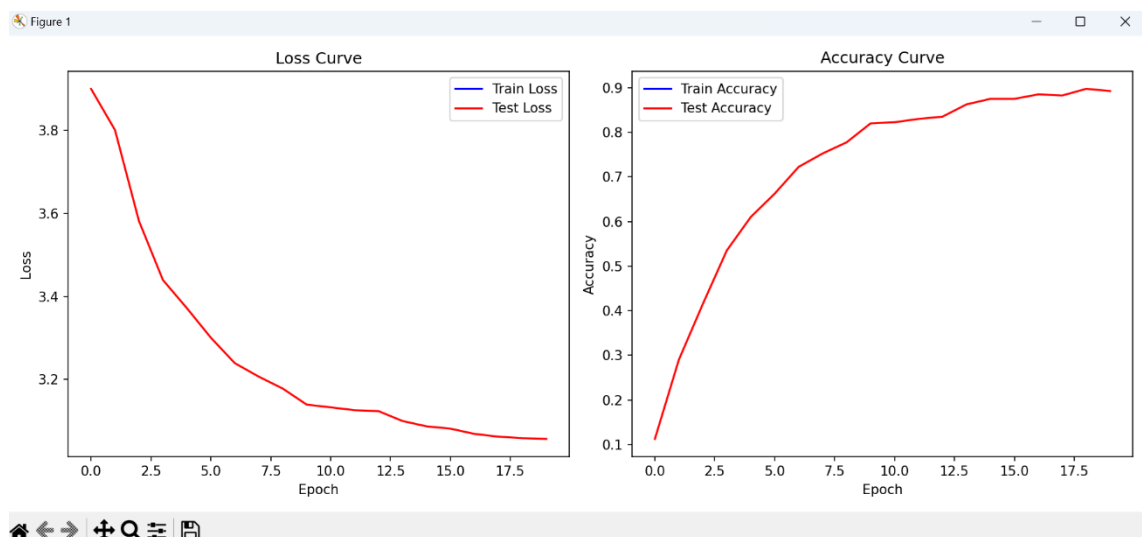


Figure 2 Loss and Accuracy Curves for Baseline Performance

These metrics show that the model is learning efficiently on the ESC-50 dataset when done without incremental learning. It also shows the effectiveness of using pre-trained models as it is very easy to adapt them to new tasks without the need for training from scratch.

4.2 Incremental Learning Performance

The model is trained using the incremental learning setup introduced in the previous section. The 50 classes of the ESC-50 dataset are divided into 5 separate tasks of 10 classes each and this is fed into the model sequentially. The evaluation is done by analysing the accuracies, F1 scores, mAP, and AUC values after each task to understand the performance of the model during incremental learning. The forgetting curve along with the evaluation metrics values over each task are plotted at the end to analyse the overall performance of the model.

Figure 3 below shows the accuracy of the model after training on each task. The overall accuracy of the model over all tasks remains in a good range of 70% in Task 4 to 92.5% in Task 2.

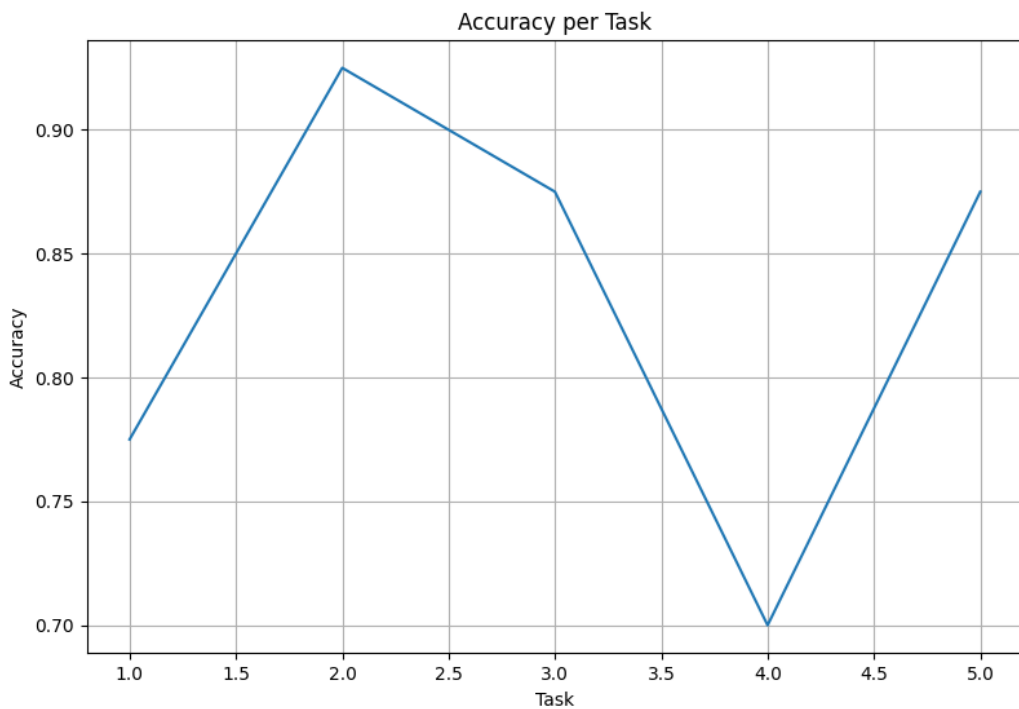


Figure 3 Accuracy Per Task for Incremental Learning

All the tasks begin with a low accuracy around 10-20% during initial epochs but steadily increase to good accuracies during training as the weights adapt. Table 1 below summarizes all the evaluation metric values after training on each task.

Table 1: Evaluation metric values for all tasks

Task No.	Accuracy	F1 Score	mAP	AUC
1	77.5%	73.83%	95.47%	99.11%
2	92.5%	92.23%	98.50%	99.81%
3	87.5%	83.92%	91.70%	98.77%
4	70.00%	62.62%	87.17%	96.86%
5	87.5%	83.18%	92.75%	98.61%

It can be seen that model performs well on Task 1 and Task 2 which is very common in incremental learning since initial tasks get the full network capacity. Task 2 even has early stopping at epoch 20 which is very impressive. Task 3 also had good accuracy but the performance drops for Task 4. This could be because of catastrophic forgetting, or it might be that it has a different and more complex class distribution than previous tasks. Task 5 has a stable performance during 25-30 epochs indicating good retention of previous data. The forgetting analysis for the model after each task is done in the next section.

4.3 Catastrophic Forgetting Analysis

The forgetting curve for the model plotted over each task is shown in Figure 4 below.

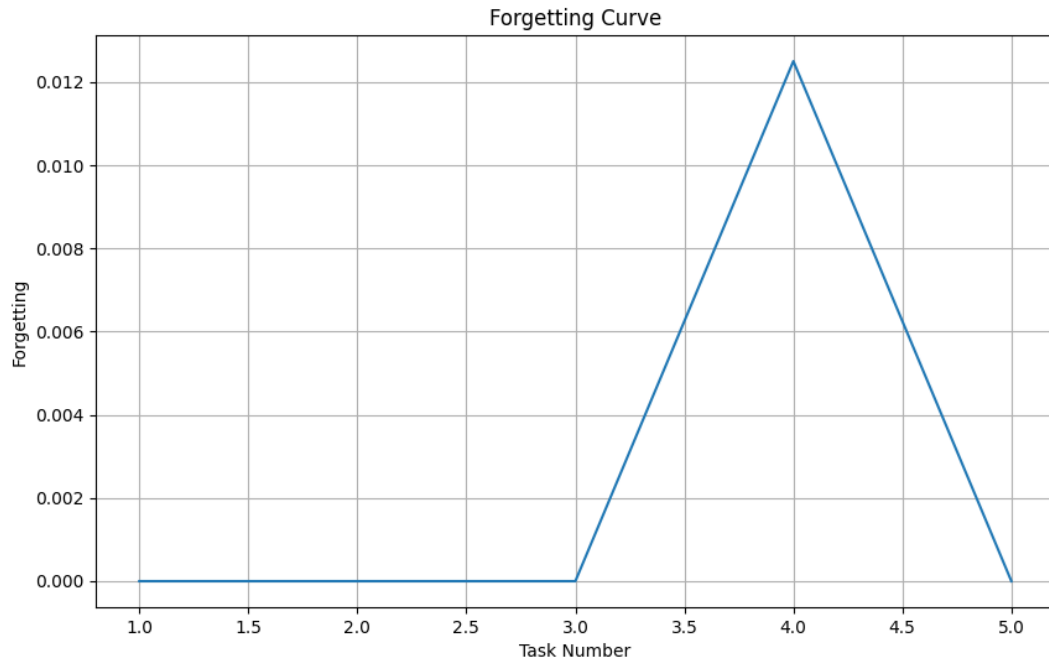


Figure 4 Plot for Catastrophic Forgetting Value Per Task

The graph depicts a 0 forgetting value for Tasks 1, 2, and 3 indicating that the model performs almost similar on these tasks in the beginning and after training on the final task. This shows that the model is retaining previous knowledge well which can be credited to the Experience Replay mechanism and EWC regularization. The forgetting value for Task 4 increases which might be due to shift in data distribution. The model is either overwriting prior knowledge or struggling to integrate new classes. Forgetting for task 5 is not defined since it is measured as the difference between the maximum accuracy achieved on a task and accuracy of the task after training on the final task.

Overall, the forgetting curve follows a typical non-linear trend for incremental learning tasks where the model copes well to initial tasks, struggles in the middle with training on new classes, and later stabilizes as it adapts to the incremental learning setup. Better incremental learning strategies can be used to overcome this catastrophic forgetting and improving results.

5. CONCLUSION

This study explored an incremental learning setup for audio classification using pre-trained models. The effectiveness of using pre-trained model weights is well depicted by the baseline performance accuracy achieved when PANNs CNN14 Model is used to train on the ESC-50 dataset without incremental learning. While the incremental learning model discussed in Section 3 adapts well to initial tasks, its performance degrades over Task 3 and Task 4 but improves again over Task 5. This could be due to catastrophic forgetting during learning of Task 4 as they might vary a lot from previous tasks.

This behaviour can be tackled by implementing more strategies and optimizing the training process to mitigate catastrophic forgetting. The scaling of EWC and KD losses can be experimented with to find the most optimum solution. The Experience Replay method could also be improved since it is selecting previous samples using centroid based selection in this study. Better strategies like True Herding and Gradient Sample Selection are discussed in incremental learning literature that give better results and help in reducing catastrophic forgetting [12].

Another factor affecting these results could be the size of the ESC-50 dataset. Since the dataset has only 40 samples per class, it can hurt the generalization capability of the model. Experience Replay buffer also has memory limitations since the idea of incremental learning is to learn new tasks while storing minimal information of previous tasks. If the size of the dataset is larger, the model would stabilize faster, and the replay buffer would benefit from more diverse representation of samples. The chances of overfitting would reduce, and the generalization capability of the model can be better assessed.

Future works in this study can explore efficient and more optimal model solutions to achieve higher accuracies with minimal forgetting of previous tasks. Smarter buffer management and alternative regularization techniques could give better results for incremental learning task for audio classification.

REFERENCES

- [1] Wu Y, Chen Y, Wang L, Ye Y, Liu Z, Guo Y, Fu Y. Large scale incremental learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition 2019 (pp. 374-382). Available: http://openaccess.thecvf.com/content_CVPR_2019/html/Wu_Large_Scale_Incremental_Learning_CVPR_2019_paper.html
- [2] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A.A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, & R. Hadsell, Overcoming catastrophic forgetting in neural networks, Proc. Natl. Acad. Sci. U.S.A. 114 (13) 3521-3526, (2017). Available: <https://doi.org/10.1073/pnas.1611835114>
- [3] Z. Li and D. Hoiem, "Learning without Forgetting," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 12, pp. 2935-2947, 1 Dec. 2018, doi: 10.1109/TPAMI.2017.2773081. Available: <https://ieeexplore.ieee.org/abstract/document/8107520>
- [4] Hu, Guannan, et al. "Prioritized Experience Replay for Continual Learning." 2021 6th International Conference on Computational Intelligence and Applications (ICCIA), IEEE, 2021, pp. 16–20, Available: <https://doi.org/10.1109/ICCIA52886.2021.00011>
- [5] Zaman, Khalid & Sah, Melike & Direkoglu, Cem & Unoki, Masashi. (2023). A Survey of Audio Classification Using Deep Learning. IEEE Access. 11. 1-1. 10.1109/ACCESS.2023.3318015. K. Ruohonen, Matemaattisen tekstin kirjoittaminen, Tampereen teknillinen yli-opisto, 2009, 7 s. Available: https://www.researchgate.net/publication/374101086_A_Survey_of_Audio_Classification_using_Deep_Learning#:~:text=Deep%20learning%20models%20are%20able,represented%20in%20a%20suitable%20form.
- [6] Hosna, A., Merry, E., Gyalmo, J. et al. Transfer learning: a friendly introduction. *J Big Data* 9, 102 (2022). Available: <https://doi.org/10.1186/s40537-022-00652-w>
- [7] Kong Q, Cao Y, Iqbal T, Wang Y, Wang W, Plumbley MD. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. IEEE/ACM Transactions on Audio, Speech, and Language Processing. 2020 Oct 19;28:2880-94.

Available:

<https://ieeexplore.ieee.org/abstract/document/9229505>

- [8] Tsalera, E., Papadakis, A., & Samarakou, M. (2021). Comparison of Pre-Trained CNNs for Audio Classification Using Transfer Learning. *Journal of Sensor and Actuator Networks*, 10(4), 72.

Available:

<https://doi.org/10.3390/jsan10040072>

- [9] M. Mulimani and A. Mesaros, "A Closer Look at Class-Incremental Learning for Multi-Label Audio Classification," in *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 1293-1306, 2025, doi: 10.1109/TASLPRO.2025.3547233.

Available:

<https://ieeexplore.ieee.org/abstract/document/10909318>

- [10] Karam, S., Ruan, S.J., Haq, Q.M.u. et al. Episodic memory based continual learning without catastrophic forgetting for environmental sound classification. *J Ambient Intell Human Comput* 14, 4439–4449 (2023).

Available:

<https://doi.org/10.1007/s12652-023-04561-5>

- [11] Konasani, Venkata Reddy, and Shailendra Kadre. 2021. *Machine Learning and Deep Learning Using Python and TensorFlow*. 1st ed. New York: McGraw Hill.

Available:

<https://www.accessengineeringlibrary.com/content/book/9781260462296>

- [12] Brignac D, Lobo N, Mahalanobis A. Improving replay sample selection and storage for less forgetting in continual learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision 2023 (pp. 3540-3549).

Available:

https://openaccess.thecvf.com/content/ICCV2023W/VCL/html/Brignac_Improving_Replay_Sample_Selection_and_Storage_for_Less_Forgetting_in_ICCVW_2023_paper.html

- [13] Rebuffi SA, Kolesnikov A, Sperl G, Lampert CH. icarl: Incremental classifier and representation learning. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition 2017 (pp. 2001-2010).

Available:

https://openaccess.thecvf.com/content_cvpr_2017/html/Rebuffi_iCaRL_Incremental_Classifier_CVPR_2017_paper.html

- [14] Keita, Z. (2023, November 14) An Introduction to Convolution Neural Networks (CNNs). *DataCamp*.

Available:

<https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns>