Machine Unlearning on Audio Classification

Hongyu Wu
Department of Electrical Engineering
The Cooper Union for the Advancement of Science and Art
New York, New York
hongyu.wu@cooper.edu

Diego Toribio
Department of Electrical Engineering
The Cooper Union for the Advancement of Science and Art
New York, New York
diego.toribio@cooper.edu

***Abstract*— Machine unlearning is a critical emerging field aimed at addressing data privacy concerns and enhancing the adaptability of machine learning models by enabling selective forgetting of specific data subsets. In this study, we focus on machine unlearning within the domain of audio processing, specifically for digit classification tasks. Leveraging a Convolutional Neural Network (CNN) trained on Mel-Frequency Cepstral Coefficient (MFCC) representations, we investigate techniques such as noise injection and Batch Normalization re-initialization to effectively remove the influence of targeted data while preserving the model's overall performance. Our work provides a novel perspective on adapting unlearning techniques for audio data, offering a foundation for privacy-centric applications in speech and audio analysis.**

Keywords— audio classification, MFCC, machine unlearning, deep learning, finetuning

1. Introduction

Machine learning has revolutionized numerous domains, including audio classification, where it enables tasks such as speech recognition, speaker identification, and environmental sound analysis. Leveraging advanced architectures like Convolutional Neural Networks (CNNs), machine learning models can extract intricate patterns from audio data represented through features such as Mel-Frequency Cepstral Coefficients (MFCCs). These capabilities have made audio classification a cornerstone for applications in virtual assistants, call centers, and other speech-driven technologies.

However, the widespread use of machine learning raises significant data privacy concerns. In many cases, models inadvertently retain sensitive information from the training data, making compliance with regulations challenging when users request their data to be erased. This concern has driven the development of machine unlearning, a process that enables models to forget specific subsets of data without requiring complete retraining.

In the context of audio classification, unlearning becomes particularly challenging due to the complexity and variability of audio signals. Developing techniques to selectively remove the influence of specific data—such as a particular user's voice or speaker information—while maintaining the model's accuracy for other tasks is critical. This work explores these challenges by focusing on digit classification using audio data and proposing strategies for effective unlearning that balance privacy requirements with model performance.

2. Background

Audio processing relies on several foundational concepts that underpin the techniques used in this work. Mel-Frequency Cepstral Coefficients (MFCCs) are a widely used feature extraction method that transforms audio signals into compact and perceptually relevant representations. By mimicking the human auditory system's sensitivity to different frequencies, MFCCs effectively capture key characteristics of audio signals, making them well-suited for tasks like speech and sound classification. These features are commonly used as input to machine learning models.

Convolutional Neural Networks (CNNs) are particularly advantageous in audio processing due to their ability to learn hierarchical representations of input data. Through layers of convolution and pooling, CNNs capture local patterns and progressively build more abstract representations. This ability makes them highly effective for complex classification tasks, including digit recognition, where the variability in audio signals poses a significant challenge. Additionally, Batch Normalization [1] is often employed in CNNs to stabilize training and improve convergence, which is critical when adapting models for specialized tasks such as unlearning.

While machine unlearning has seen significant exploration in computer vision, particularly for tasks like image classification and generation [2], its application to audio processing remains underexplored. Existing unlearning methods, such as gradient-based adjustments [3], weight saliency [3], and memory matrix multiplication [2], have demonstrated success in selectively removing specific data influences from models in visual domains. However, these approaches have not been comprehensively applied to audio data, where the temporal and spectral complexity introduces unique challenges.

This work bridges this gap by applying machine unlearning techniques to audio classification, a fundamental task in audio processing. By developing and testing unlearning strategies tailored to audio data, this study expands the scope of unlearning research, offering insights into its application in a domain that has received limited attention.

3. Methods and Model

A. *Model*

The proposed system employs a Convolutional Neural Network (CNN) architecture for classifying audio using MFCC spectrograms [4] of audio data. The network is designed with sequential layers to facilitate robust feature extraction and classification. Convolutional layers are used to extract local features from the MFCC spectrograms, capturing patterns in the time-frequency domain. Batch Normalization is incorporated to stabilize training by normalizing activations, which reduces internal covariate shift and accelerates convergence [1]. Activation functions introduce non-linearity, allowing the network to learn complex feature mappings. Dense layers are used to perform the final classification of spoken digits. To mitigate overfitting and enhance generalization to unseen data, dropout [5] is applied throughout the network and progressively increases. Dropout is applied more heavily on the fully connected layers due to their vulnerability to overfitting [5]. The model's input is a 20x20 MFCC spectrogram, which serves as a compact representation of the audio signal's time-frequency characteristics. The size of the network is purposely kept small to ensure that the model learns the set of useful features and nothing extraneous that affects the generalization performance.

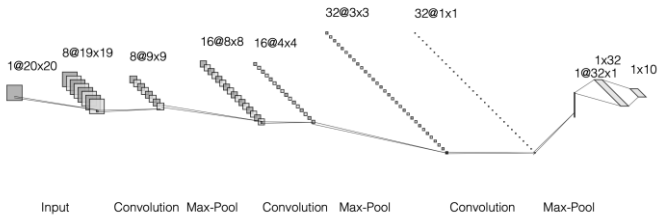

Figure 1: Model Architecture

B. *Preprocessing Methods*

The raw audio data undergoes a preprocessing pipeline to create a consistent input representation suitable for model training. All audio files are first resampled to 8,000 Hz to ensure uniformity across samples. MFCC spectrograms are then generated with a fixed length of 20 frames, where padding or truncation is applied to standardized dimensions. The dataset is divided into training, validation, and testing subsets. For the unlearning experiments, data from specific speakers is excluded and set aside to evaluate the effectiveness of the unlearning techniques. MFCC spectrograms are generated with the Python Librosa [6] library.

C. *Unlearning Techniques*

To enable machine unlearning, two distinct techniques are implemented, both designed to remove the influence of excluded data while maintaining overall model performance. The first technique, Weight Perturbation, involves introducing random noise to the model's weights. By carefully tuning the scale of injected noise, the method disrupts learned representations associated with the excluded data without significantly impairing the model's ability to classify the remaining data. The second technique, Selective Layer Reset, targets specific layers of the network by resetting their parameters to their initial state. This effectively removes the influence of the excluded data while preserving the integrity of other learned features. The third technique resets the moving average of the batch normalization layers in the network.

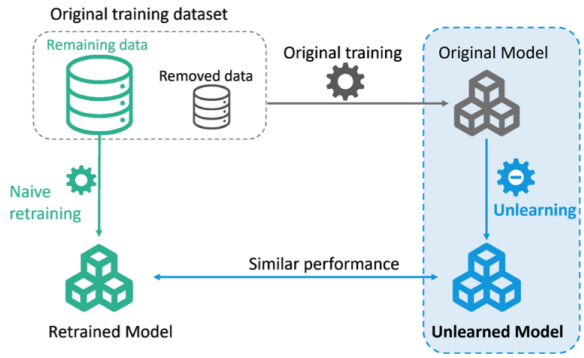

Figure 2: Machine Unlearning [7]

D. *Evaluation Criteria*

The performance of these unlearning techniques is evaluated using a combination of standard classification metrics and unlearning-specific measures. Classification metrics, including accuracy, precision, recall, and F1-score, are used to assess the model's ability to generalize on the retained dataset. To evaluate unlearning effectiveness, the model's ability to "forget" the excluded speaker data is measured by analyzing its performance on this excluded subset. Metrics such as accuracy degradation or reduced predictive capability on excluded data are used to quantify the success of the unlearning process. By combining these evaluations, the study ensures that the unlearning techniques effectively remove unwanted data influences while preserving the model's performance on the retained dataset and compared to a baseline naively retrained model.

4. Experiments

1) Dataset

The Free Spoken Digit Dataset (FSDD) [8] was selected due to its suitability for audio and its structure, which facilitates the evaluation of machine unlearning techniques. The dataset consists of 3,000 audio recordings, representing 10 digits (0–9) spoken by six distinct speakers. Each speaker contributes 50 samples per digit, resulting in a balanced dataset ideal for training and evaluation. The recordings are mostly clear and sampled consistently making it an effective choice for exploring the impact of unlearning techniques in audio classification. Additionally, the segmentation by speaker identity provides a straightforward framework for defining retained and excluded subsets, critical for testing the selective forgetting process.

The dataset is split in a specific way to prevent potential information leak across the multiple phases of training and unlearning. The full dataset is first split into two parts, the retained set and the excluded set. Excluded set contains one speaker with 500 samples. The retained set is then further split into three folds, the training set, validation set and test set. The

excluded set is then concatenated to the training set forming 2525 samples. The validation set consists of 225 samples and the test set contains 250 samples.

2) Unlearning Baseline

To properly evaluate the effectiveness of unlearning techniques, a baseline model is trained from scratch. From now on this model will be referred to as the gold [7] model. The gold model is trained exclusively on the retained set to minimize the effect of samples from the excluded set so that it acts as a fair baseline model. Evaluations of the effectiveness of the unlearning techniques will be compared to the gold model.

3) Experimental Setup

All experiments are performed on the Kaggle platform with a P100 GPU. To perform unlearning, a baseline model is pretrained on the FSDD dataset for 100 epochs with a batch size of 16. The gold model is trained for 120 epochs to mimic the number of training steps to prevent underfitting due to less data. Both models are trained with the Adam optimizer. [9] To ensure robustness and reliability of the results, each experiment was repeated three times using different random seeds, allowing for statistical validation and maintain consistency in findings. Source codes are implemented in the Keras [10] framework.

4) Unlearning Procedures

For the basic unlearning scheme, two unlearning techniques are applied sequentially. We also examine the effect of a third technique on batch normalization re-initialization. First, all fully connected layers have their weights reset, erasing any influence of the excluded set on the original model on these layers. The model is then trained on the retained set for 5 epochs. Second, the model is frozen, and a random selection of layers is injected scaled gaussian noise and unfrozen to train for 1 epoch on the retained data without the excluded set. This step is repeated 5 times each time with a random selection of both the number of layers and the layers themselves to ensure enough diversity of noise is added to the model to minimize the effect of the excluded set. Finally, the entire model is unfrozen and finetuned for 5 epochs. The entire length of the unlearning process is only 12.5% of the naive retraining approach.

5. Results

a) Model Performance

To evaluate the proposed unlearning techniques, we analyzed the model's classification performance on both the retained test dataset and the excluded dataset. Standard metrics such as accuracy, precision, recall, and F1-score were computed for the baseline model (trained without unlearning) and the unlearned models (after applying Weight Perturbation and Selective Layer Reset and/or Batch Normalization Reset).

The classification results demonstrate minimal degradation in performance on the test dataset, indicating that the proposed techniques effectively preserved the model's ability to generalize to retained data. However, for the excluded dataset, a significant drop in accuracy and confidence scores was observed, highlighting the success of the unlearning techniques in reducing the influence of excluded data.

A detailed comparison of one metric is presented in Table 1, showing the performance across different unlearning schemes. Note that for batch normalization reset, two different sets of experiments are tested. The experiment titled "All BN" refers to the reset of all batch normalization layers while the experiment titled "Conv BN" only resets the batch normalization layers after the convolutional layers.

Dataset	Base	All BN	Conv BN
Test	-0.0223	-0.0207	-0.0193
Excluded	-0.1596	-0.1667	-0.1793

Table 1: Performance Loss after unlearning

Table 1 reports the difference in macro F1 scores of the models' performance before and after the unlearning procedures. In both batch normalization reset cases the unlearning effect is better since the gap is somewhat larger than the base scheme. The performance degradation on the test set is also smaller albeit much less substantial.

Model	Test (vs Gold)	Excluded (vs Gold)
Base	-0.04	0.14
All BN	-0.03	0.13
Conv BN	-0.03	0.12

Table 2: Performance comparison to ideal case

Table 2 reports the performance differences in macro F1 scores compared to the gold model. A more negative gap on the test set indicates worse performance compared to ideal unlearning where all effects of the excluded set have been removed since the gold model has never seen those samples and can effectively act as one example of ideal unlearning. The performance gap on the excluded set is expected to be positive since the unlearning schemes are far from perfect and will inadvertently include some influence of the unlearned data. Nonetheless, a smaller gap indicates better unlearning performance.

The batch normalization reset method again demonstrates small yet significant improvements to both the unlearning and performance retention capabilities when compared to the base unlearning scheme. It is noteworthy that the difference between the Conv BN approach and the basic approach is statistically significant with a P-value around 0.02.

b) Analysis

The main hyperparameters of the unlearning process are the length of each step, the number of iterative steps, and the noise scale in the Weight Perturbation step. The permutation of the Weight Perturbation step could also be changed but no notable change in performance is observed.

To maximally limit the length of the unlearning process, the noise scale is chosen to be of significant ratio to the magnitude of the layer weights, especially the convolutional layers. Typical magnitude of the convolutional weights is less than or equal to 0.2. This value is obtained through randomly selecting and averaging the values of some weights of the feature extraction layers. To balance the purpose of unlearning as well as efficiency of the process, an initial value of 0.04 for the noise scale is chosen and the number of steps is 3. This

results in quite minimal unlearning performance and it is deemed that stronger noise and more steps are needed to achieve better results. Finally, the noise scale is selected to be 0.08 and the number of steps increased to 5. This not only balances the length of the unlearning steps to each take 1/3 of the total unlearning process and also ensures that enough perturbation to the weights is introduced so that the model can indeed unlearn the excluded dataset.

The improvement through initializing the batch normalization layers along with the classifier head can is perhaps not too surprising due to the behavior of the layer even though it is a non-trainable layer. As stated in [4], batch normalization keeps track of the moving average of the batches the layer sees during training, which includes the excluded set. This information is retained in the basic unlearning scheme and left untouched. Although the unlearning process does change the moving average, it is difficult to quantify the portion of information that the moving average retains. Therefore, simply re-initializing the batch normalization layers should erase all information of the excluded set carried by the layer.

Though the performance difference between the basic unlearning method and the "All BN" method is not statistically significant, and the "All BN" approach and the "Conv BN" approach results are also not statistically significant, the statistical significance of the difference between the basic method and the "Conv BN" method is clearly demonstrated amongst the experiments. The speculation of the "All BN" and "Conv BN" methods having progressively better performance is the difference in the underlying inductive biases of the layers. The "All BN" approach includes batch normalization layers after fully connected layers whereas the "Conv BN" only includes those after convolutional layers.

6. Future Work and Conclusion

The speculation of the difference between the methods can be tested and verified with the more newly introduced normalization methods, mainly the popular layer normalization [11] and instance normalization [12]. These methods do not perform normalization across batches or have parameters accumulated through the training process and therefore should be invariant to the existence and absence of the excluded data.

Another area of future work in machine unlearning is the introduction of explainability into the influence of unlearned data. Incorporating explainability methods to provide interpretable insights into how and why certain data influences were removed, further improving user trust in privacy-preserving models.

This work explores the application of machine unlearning techniques in the domain of spoken digit classification, leveraging the Free-Spoken Digit Dataset and a Convolutional Neural Network trained on MFCC representations. The unlearning techniques—Weight Perturbation, Selective Layer Reset and Batch Normalization Reset—successfully removes the influence of excluded speaker data while preserving overall model performance on the retained dataset. Key findings indicate these methods effectively "forget" specific data with minimal impact on test accuracy, precision, recall, and F1-score, as highlighted in the experimental results.

The implications of this study extend beyond spoken digit classification, demonstrating the potential for machine unlearning to address privacy-sensitive applications and adapt models dynamically in scenarios where data deletion or exclusion is required. These techniques contribute to the growing body of research on ensuring compliance with privacy regulations, such as the General Data Protection Regulation (GDPR), while maintaining the utility of machine learning models in practical settings.

Acknowledgment

The authors would like to thank J. T. Colonel for his invaluable guidance and technical support throughout this work. We also acknowledge the contributions of the Kaggle platform for providing access to the computational resources used in this study, including the P100 GPUs and the cloud hosted data services for making collaboration easier. Further, we would like to acknowledge the NeurIPS 2023 Machine Unlearning Challenge, hosted on Kaggle, for inspiring this research. The competition provides valuable insights into the application of machine unlearning techniques, which directly influences the approach taken in this study. Lastly, we express gratitude to the developers of the Free Spoken Digit Dataset and the developers of the Keras and Librosa frameworks for making this work possible.

References

[1] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 2015, pp. 448–456.

[2] S. Poppi, S. Sarto, M. Cornia, L. Baraldi, and R. Cucchiara, "Multi-Class Unlearning for Image Classification via Weight Filtering," IEEE Intelligent Systems, vol. 38, no. 2, pp. 50–59, Mar. 2023. [Online]. Available: https://arxiv.org/abs/2304.02049

[3] C. Fan, J. Liu, Y. Zhang, E. Wong, D. Wei, and S. Liu, "SalUn: Empowering Machine Unlearning via Gradient-based Weight Saliency in Both Image Classification and Generation," arXiv preprint arXiv:2310.12508, Oct. 2023. [Online]. Available: https://arxiv.org/abs/2310.12508

[4] Z. K. Abdul and A. K. Al-Talabani, "Mel Frequency Cepstral Coefficient and its Applications: A Review," in IEEE Access, vol. 10, pp. 122136-122158, 2022, doi: 10.1109/ACCESS.2022.3223444.

[5] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, vol. 15, no. 56, pp. 1929–1958, 2014.

[6] B. McFee, C. Raffel, D. Liang, D. P. W. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in Python," Proceedings of the 14th Python in Science Conference (SciPy), vol. 8, no. 1, pp. 18–25, 2015.

[7] J. Xu, Z. Wu, C. Wang, and X. Jia, "Machine unlearning: Solutions and challenges," arXiv, Aug. 2023. [Online]. Available: https://arxiv.org/abs/2308.07061.

[8] J. Klievink, "Free Spoken Digit Dataset (FSDD)," GitHub repository, 2016. [Online]. Available: https://github.com/Jakobovski/free-spoken-digit-dataset.

[9] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 2015.

[10] F. Chollet et al., "Keras," 2015. [Online]. Available: https://keras.io

[11] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," arXiv preprint arXiv:1607.06450, 2016.

[12] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance Normalization: The Missing Ingredient for Fast Stylization," arXiv preprint arXiv:1607.08022, 2016.

[13] E. Triantafillou, F. Pedregosa, J. Hayes, P. Kairouz, I. Guyon, M. Kurmanji, G. K. Dziugaite, P. Triantafillou, K. Zhao, L. S. Hosoya, J. C. S. Jacques Jr., V. Dumoulin, I. Mitliagkas, S. Escalera, J. Wan, S. Dane, M. Demkin, and W. Reade, "NeurIPS 2023 - Machine Unlearning," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/competitions/neurips-2023-machine-unlearning.