ENHANCING AUTOMATIC CHORD RECOGNITION VIA PSEUDO-LABELING AND KNOWLEDGE DISTILLATION

Nghia Phan[1]*, Rong Jin*[1]*, Gang Liu*[2]*, and Xiao Dong*[3]

[1]California State University, Fullerton, CA, USA [2]Microsoft, Redmond, WA, USA [3]Beijing Normal-Hong Kong Baptist University, China
ptnghia@csu.fullerton.edu, rong.jin@fullerton.edu, liu.gang@microsoft.com, xiaodong@bnbu.edu.cn

ABSTRACT

Automatic Chord Recognition (ACR) is constrained by the scarcity of aligned chord annotations, which are costly to acquire. At the same time, open-weight pre-trained models are more accessible than their proprietary training data. In this work, we present a two-stage training pipeline that leverages pre-trained models together with unlabeled audio. The proposed method decouples training into two stages. In the first stage, we use the pre-trained BTC model [1] as a teacher to generate pseudo-labels for over 1,000 hours of diverse unlabeled audio and train a student model solely on these pseudo-labels. In the second stage, the student is continually trained on ground-truth labels as they become available. To prevent catastrophic forgetting of the representations learned in the first stage, we apply selective knowledge distillation (KD) from the teacher as a regularizer. In our experiments, two models (BTC, 2E1D) were used as students. In Stage 1, using only pseudo-labels, the BTC student achieves about 99% of the teacher's performance, while the 2E1D model achieves about 97% of the teacher's performance across seven standard `mir_eval` metrics. After continual training with labeled data in Stage 2, the resulting BTC student model consistently surpasses both the traditional supervised learning baseline and the original pre-trained teacher model across all metrics. The resulting 2E1D student model also outperforms the supervised baseline and approaches teacher-level performance, with both models demonstrating substantial gains on rare chord qualities.

1. INTRODUCTION

Automatic Chord Recognition (ACR) from audio signals remains a fundamental task in Music Information Retrieval (MIR) that aims to identify the harmonic content of audio recordings by outputting a sequence of chord labels. While large-scale labeled datasets are readily available for many machine learning domains, ACR faces significant data constraints: publicly available labeled chord datasets remain limited in both size and diversity [2]. This scarcity stems from the substantial manual effort required for precise audio-label alignment and consistent harmonic interpretation, as chord boundaries are inherently ambiguous and context-dependent [3, 4]. Furthermore, chord vocabularies are large and highly imbalanced; models excel on frequent major/minor chords but underperform on rare seventh and extended chords [2, 4, 5].

Recent semi-supervised ACR methods [5, 6] have attempted to integrate unlabeled data into training. They use pseudo-labeling or contrastive learning but require ground-truth labels from the outset to fuse labeled and unlabeled data within a single training run. This tight coupling limits their utility when labeled data is initially unavailable or when training data remains proprietary.

To overcome these limitations, we experiment with two semi-supervised learning paradigms: *pseudo-labeling* [7] and *knowledge distillation* (KD) [8]. Pseudo-labeling is a self-training technique where a pre-trained "teacher" model generates hard or soft target predictions on unlabeled audio, which serve as synthetic ground truth to train a "student" network [7]. Knowledge distillation is a model regularization and compression framework wherein a student is optimized to match the full output probability distribution (soft logits) of a teacher model, transferring knowledge about inter-class relationships that hard annotations omit [8]. By combining these paradigms in a decoupled, two-stage pipeline (see Figure 3), we train a student model that can match or exceed the teacher's capabilities. In the first stage, a pre-trained teacher model generates pseudo-labels for over 1,000 hours of diverse unlabeled audio, and a student model is trained solely on these pseudo-labels until convergence, without requiring any ground-truth annotations. In the second stage, when labeled data becomes available, the pseudo-label-trained student is continually trained on ground-truth labels. To retain the representations acquired in the first stage, we apply selective KD from the teacher as a regularizer throughout the second stage.

Our two-stage training approach decouples labeled and unlabeled data, making the training viable even when labeled data is initially unavailable. When data collection is expensive, this offers a practical alternative by training a model entirely without labels at only a minor accuracy cost. Notably, open-weight pre-trained models are often more readily available than their training data, offering a practical way to leverage teacher knowledge without access to proprietary datasets. Additionally, we show that the accuracy gains are disproportionately concentrated on rare chord qualities.

We also introduce a compact, dual-encoder architecture (2E1D) that is based on the Transformer architecture [9] and is lighter-weight than the teacher model. We demonstrate that our method can generalize across architectures. Our experiments show that the best resulting student surpasses both the traditional supervised learning approach and the pre-trained teacher model across all standard `mir_eval` metrics [10], with particularly large gains on rare chord qualities. We open-source a chord-recognition web application[1] that lets practitioners run and validate our models on their own audio locally.

2. RELATED WORK

Pseudo-labeling, where a trained model generates labels for unlabeled data, has become a cornerstone of semi-supervised learn-

[1] `https://github.com/ptnghia-j/ChordMiniApp`

ing [7]. The Noisy Student framework [11] demonstrated that iteratively training larger student models on pseudo-labeled data with noise injection can surpass teacher performance, establishing a paradigm for leveraging unlabeled data at scale. FixMatch [12] combined consistency regularization with pseudo-labeling using confidence thresholds, while Mean Teacher [13] maintained an exponential moving average of student weights to produce consistency targets. Meta Pseudo Labels [14] further improved teacher–student training by jointly optimizing the teacher based on student feedback. Within audio signal processing, pseudo-labeling has been successfully applied to various tasks including speech recognition [15], piano transcription [16], and music tagging [17]. For ACR specifically, Bortolozzo et al. [5] adapted Noisy Student for rare chord recognition, employing confidence filtering and iterative teacher–student training to address class imbalance. Li et al. [6] combined contrastive pre-training with noisy-student semi-supervision for large-vocabulary chord recognition. However, these approaches typically require ground-truth labels from the outset and train on mixed pseudo-label and ground-truth signals within a single pipeline. In contrast, we investigate whether separate incremental learning stages can achieve comparable or superior performance by first training on pseudo-labels alone, then adapting to labeled data.

Transferring knowledge via temperature-scaled soft targets, or knowledge distillation [8], has been shown to provide richer supervision than hard labels. Yuan et al. [18] studied conditions under which biased soft labels can still improve student generalization. Chen [19] showed that KD can be interpreted as a form of output regularization, while Mansourian et al. [20] detailed its general regularizing effects. In continual learning, KD has been used to preserve prior knowledge while adapting to new data, mitigating catastrophic forgetting [21, 22]. Learning without Forgetting (LwF) [21] pioneered using distillation from the model's own previous state to retain old knowledge when learning new tasks. KD has primarily been used for model compression, training smaller, more efficient students [17]. In real-time audio systems, such compression is vital for deploying computationally demanding deep learning models onto low-latency digital audio workstations (DAWs) or embedded hardware targets, as explored in recent studies on real-time inference engines [23] and structured network pruning [24]. While we similarly optimize a compact student model (2E1D), we apply KD differently: as a regularization mechanism during continual learning that anchors the student to the teacher's generalized representations. Consequently, KD enables the student to adapt to new labels while retaining prior pseudo-label knowledge, reducing catastrophic forgetting.

Continual learning addresses the challenge of learning from non-stationary data streams without forgetting previously acquired knowledge [25]. In the MIR domain, this challenge is particularly relevant given the ongoing creation of new music and evolving annotation standards. While task-incremental and class-incremental scenarios have received significant attention, data-incremental continual learning remains underexplored for ACR. In this setting, the task and label space remain unchanged but new data arrive.

3. METHODOLOGY

3.1. Problem Formulation

Let $\mathcal{D}_l = \{(x_i, y_i)\}_{i=1}^{N_l}$ denote a small labeled dataset where $x_i \in \mathbb{R}^{T\times F}$ represents time–frequency features (e.g., Constant-Q Transform) with T frames and F frequency bins, and $y_i \in \{1, 2, \ldots, V\}^T$ are frame-wise chord labels over a vocabulary of size V. Let $\mathcal{D}_u = \{x_j\}_{j=1}^{N_u}$ represent large-scale unlabeled datasets where $N_u \gg N_l$. Our objective is to train an effective student model f_s by leveraging pseudo-labels generated from $\mathcal{D}_u$. This mitigates reliance on expensive manual annotations while maintaining competitive performance.

3.2. Training Method

We employ a pre-trained teacher model $f_t : \mathbb{R}^{T\times F} \to \mathbb{R}^{T\times V}$ to generate pseudo-labels for unlabeled data. Any open-weight ACR model can serve as the teacher. In our experiments, we use the BTC model from Park et al. [1] as the teacher model and adopt the standard vocabulary size $V = 170$ for all training settings. For each unlabeled sequence $x_j \in \mathcal{D}_u$, pseudo-labels are generated via frame-wise argmax over teacher outputs:

$$\hat{y}_j^{(n)} = \arg\max_{c\in\{1,\ldots,V\}} \big[f_t(x_j)\big]_{n,c}, \quad n = 1, \ldots, T. \tag{1}$$

To preserve temporal coherence, we keep sequence boundaries intact during teacher inference (e.g., padding only at segment ends) and convert 100% of frames to pseudo-labels without confidence filtering, yielding a pseudo-labeled dataset:

$$\mathcal{D}_u^{(p)} = \{(x_j, \hat{y}_j)\}_{j=1}^{N_u}. \tag{2}$$

Our method uses M complementary unlabeled datasets where $\mathcal{D}_u = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \cdots \cup \mathcal{D}_M$. The specific datasets used in our experiments are described in Section 4.1.

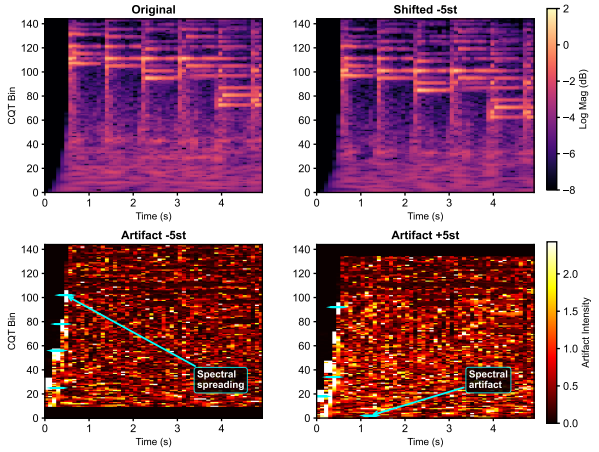

Figure 1: Constant-Q Transform (CQT) comparison revealing pitch-shifting artifacts. Annotations indicate spectral spreading (low-frequency energy diffusion) and spectral artifacts (high-frequency noise) introduced by the phase-vocoder shift.

3.2.1. Data Augmentation vs. Natural Chord Root Coverage

The supervised chord recognition task typically requires pitch-shifting data augmentation to address severe chord root imbalance in labeled datasets. Studies show that widely used datasets exhibit strong biases toward C and G major keys, comprising a substantial portion of the total duration, while keys typically notated with many

accidentals (e.g., F♯, D♭) are underrepresented [2]. Without augmentation, models overfit to key-specific spectral patterns and fail to generalize chord templates across all 12 pitch classes. Recent semi-supervised methods similarly rely on pitch-shifting when training on the labeled subset. However, pitch-shifting introduces audible artifacts. Phase-vocoder-based methods, including the Rubber Band library commonly used for music augmentation, produce transient smearing, phasiness, and spectral discontinuities [26]. These artifacts manifest as temporal blurring in Constant-Q Transform (CQT) features and artificial energy spreading across frequency bins (Figure 1).

We observe that large-scale pseudo-labeling provides an alternative that naturally eliminates the need for pitch-shifting. We analyzed chord root distributions across 101,575 pseudo-labeled tracks (∼1,300 hours) from FMA, DALI, and MAESTRO. As shown in Figure 2, all 12 pitch classes are well-represented in the accumulated root note distribution. The Shannon entropy ($H = -\sum_{i=1}^{12} p_i \log_2 p_i$) reaches 3.53 bits (the maximum being $\log_2(12) \approx 3.58$ bits), corresponding to 98.4% uniformity ($3.53/\log_2(12) \approx 98.4\%$). This natural coverage arises because diverse unlabeled corpora span multiple genres, artists, and production contexts, each contributing different key preferences that aggregate to a near-uniform distribution. Consequently, pseudo-label pre-training exposes the model to patterns in all keys without requiring pitch-shifting, establishing key-invariant representations. In our experiments, applying pitch-shifting during Stage 2 ground-truth fine-tuning shows no improvement for our pre-trained models (Section 5), confirming that Stage 1 pre-training already builds key-invariant representations. This allows us to completely omit signal manipulation and its associated artifacts throughout our training pipeline.

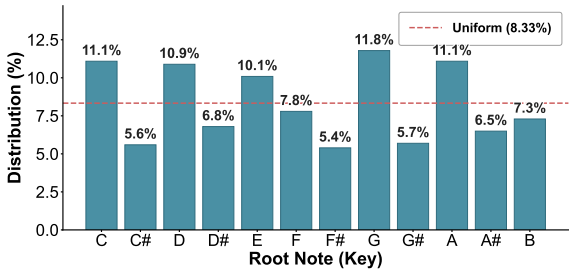

Figure 2: Duration-weighted chord root distribution across pseudo-labeled datasets (Section 4.1). The dashed line indicates uniform distribution (8.33%). Pitch classes are well-represented with 98.4% uniformity.

3.2.2. *The Training Pipeline*

Figure 3 illustrates the complete two-stage training pipeline. In Stage 1, unlabeled data is first preprocessed into Constant-Q Transform spectrograms following the procedure described in Section 4.1. A pre-trained model is used as a teacher to infer frame-wise pseudo-labels (PL) on the unlabeled audio for each track, producing paired data (Spectrogram, PL). A student model is then trained on these pseudo-labeled pairs until convergence. Optionally, knowledge distillation (KD) from the teacher's soft targets can be applied during this stage to accelerate convergence; this optional path is depicted as a dashed line in the figure. We denote the resulting model after the first stage of training as the *Student model PL*. Stage 2 begins

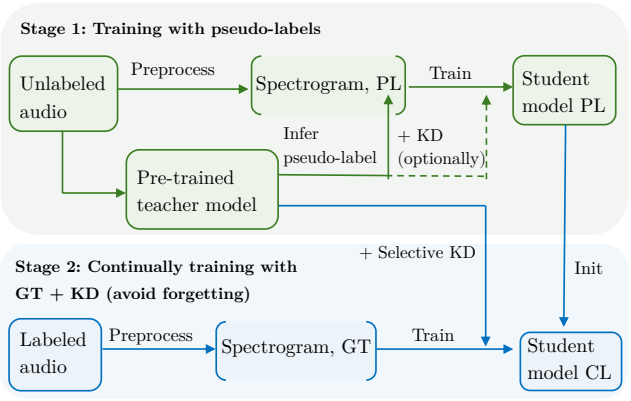

Figure 3: Illustration of the proposed two-stage training pipeline. Details of audio data are described in Section 4.1. Note: The resulting Student model CL from Stage 2 can be continually trained when additional labeled data is available.

when the labeled data becomes available. Labeled audio undergoes the same preprocessing, yielding paired data (spectrogram, GT), where GT denotes ground-truth chord annotations. In this stage, *Student model PL* serves as the weight initialization for a new model called *Student model CL*. Then the *Student model CL* serves as the initialization for subsequent rounds of continual learning as additional labeled datasets are acquired. The selective KD signal (Section 3.2.4) from the original pre-trained teacher is applied throughout the training of *Student model CL*. KD acts as regularization, anchoring the student to the teacher's distributional knowledge when ground-truth labels conflict with teacher predictions, while still allowing adaptation when they agree. When KD is applied, the training loss is a weighted sum of a classification term and a KD term as follows:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{KD}} + (1-\alpha)\mathcal{L}_C, \tag{3}$$

where $\alpha \in [0,1]$ controls the balance between teacher regularization and label supervision, and $\mathcal{L}_C$ denotes the classification loss. Specifically:

$$\mathcal{L}_C = \begin{cases} \mathcal{L}_{\text{PL}} = \frac{1}{|\mathcal{D}_u^{(p)}|} \sum_{(x,\hat{y})\in\mathcal{D}_u^{(p)}} \ell_{\text{CE}}(f_s(x), \hat{y}) & \text{if Stage 1} \\ \mathcal{L}_{\text{CE}} = \frac{1}{|\mathcal{D}_l|} \sum_{(x,y)\in\mathcal{D}_l} \ell_{\text{CE}}(f_s(x), y) & \text{if Stage 2} \end{cases}, \tag{4}$$

where $\mathcal{D}_u^{(p)}$ denotes the pseudo-labeled unlabeled data and $\mathcal{D}_l$ denotes the ground-truth labeled data. Setting α=0 reduces Eq. (3) to pure classification training without KD. The per-frame KD loss is:

$$\mathcal{L}_{\text{KD}} = \tau^2 \cdot D_{\text{KL}}\left(\sigma\left(\frac{\mathbf{z}_t}{\tau}\right) \middle\| \sigma\left(\frac{\mathbf{z}_s}{\tau}\right)\right), \tag{5}$$

where $\sigma(\cdot)$ denotes the softmax function, $\tau > 0$ is the temperature parameter that controls the smoothness of probability distributions, D_{KL} is the Kullback-Leibler divergence, and $\mathbf{z}_s, \mathbf{z}_t \in \mathbb{R}^V$ are the student and teacher logits, respectively.

3.2.3. *KD as Regularization*

We denote the temperature-softened probability distributions as $\mathbf{p}^{(s)} = \sigma(\mathbf{z}_s/\tau)$ and $\mathbf{p}^{(t)} = \sigma(\mathbf{z}_t/\tau)$, with $p_k^{(s)}$ and $p_k^{(t)}$ denoting

the k-th class probability. The τ^2 scaling ensures gradient magnitudes remain comparable to cross-entropy [8]. The KD gradient with respect to student logits is:

$$\nabla_{\mathbf{z}_s} \mathcal{L}_{\text{KD}} = \tau \left(\mathbf{p}^{(s)} - \mathbf{p}^{(t)} \right). \tag{6}$$

This gradient "pulls" student predictions toward the teacher's distribution. Combining with the classification term from Eq. (3):

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial z_{s,k}} = (1-\alpha) \frac{\partial \mathcal{L}_C}{\partial z_{s,k}} + \alpha\tau \left(p_k^{(s)} - p_k^{(t)} \right). \tag{7}$$

The second term acts as a regularizer that anchors student predictions to the teacher's distribution. When ground-truth labels conflict with teacher predictions (e.g., due to annotation noise or misalignment), this regularization prevents overfitting to erroneous labels. Conversely, when labels align with teacher predictions, the regularization does not impede adaptation. This property is particularly beneficial for ACR, where annotation inconsistencies are common due to subjective harmonic interpretation [3, 4].

3.2.4. *Selective KD*

To further improve robustness, we introduce *selective KD* that filters the teacher signal based on prediction confidence. Let γ denote the teacher's maximum temperature-softened probability for a given frame. We define an asymmetric weighting function $w(\gamma) \in [0, 1]$:

$$w(\gamma) = \begin{cases} 0 & \text{if } \gamma < \theta_{\min} \\ 1 & \text{if } \theta_{\min} \le \gamma \le \theta_{\max} \\ 1 - \lambda \cdot \frac{\gamma - \theta_{\max}}{1 - \theta_{\max}} & \text{if } \gamma > \theta_{\max} \end{cases}, \tag{8}$$

where $\theta_{\min}$ filters out uninformative low-confidence samples, $\theta_{\max}$ down-weights overconfident predictions that may bias toward majority classes, and the factor $\lambda \in [0, 1]$ controls how strongly overconfident predictions are down-weighted. The weighted KD loss becomes $\mathcal{L}_{\text{KD}}^{\text{sel}} = \frac{1}{T} \sum_{n=1}^{T} w(\gamma_n) \cdot \mathcal{L}_{\text{KD}}^{(n)}$ over all T frames, where $\mathcal{L}_{\text{KD}}^{(n)}$ denotes the per-frame KD loss of Eq. (5) evaluated at frame n and γ_n the corresponding teacher confidence. Samples near decision boundaries (moderate confidence) contain valuable uncertainty information, so we preserve full weight in the informative range $[\theta_{\min}, \theta_{\max}]$. We apply selective KD throughout all Stage 2 continual learning experiments to stabilize training by reducing gradient variance from extreme-confidence samples. We set the hyperparameters to $\theta_{\min} = 0.1, \theta_{\max} = 0.9, \lambda = 0.8$ as robust defaults based on the observed teacher-confidence distribution.

3.2.5. *Experimental Models*

BTC [1] serves as both the pseudo-labeling teacher and the baseline for self-distillation experiments. The model uses a direct frame projection followed by a stack of bi-directional transformer layers with position-wise feed-forward blocks, representing a *deeper* architecture with sequentially stacked attention layers.

For cross-architecture validation, we introduce a new compact dual-encoder architecture (2E1D) [2] for chord recognition that is purely transformer-based [9] (without any CNN layer). As shown in Figure 4, 2E1D adopts a *wider* design: (1) a frequency encoder that groups CQT bins into spectral clusters and applies self-attention to learn harmonic relationships across frequency bands; (2) a temporal encoder that processes full-band features to model chord progression patterns over time; and (3) a cross-attention fusion block that combines the two streams for final chord classification. At inference time, we apply a temporal smoothing pipeline over output logits to reduce frame-level prediction jitter. Each class logit channel is convolved with a normalized 1D Gaussian kernel $g[n] = \exp(-n^2/(2\sigma_g^2))/Z$, where $\sigma_g = K/6$ for a kernel of width K, normalization factor Z, and with replicate padding at segment boundaries. We further process the input using overlapping sliding windows with a stride of $s = \lfloor L_w(1-r) \rfloor$, where L_w is the segment length and r is the overlap ratio, accumulating per-class votes across all windows before taking the frame-wise argmax.

[2]Model: `https://github.com/ptnghia-j/ChordMini`

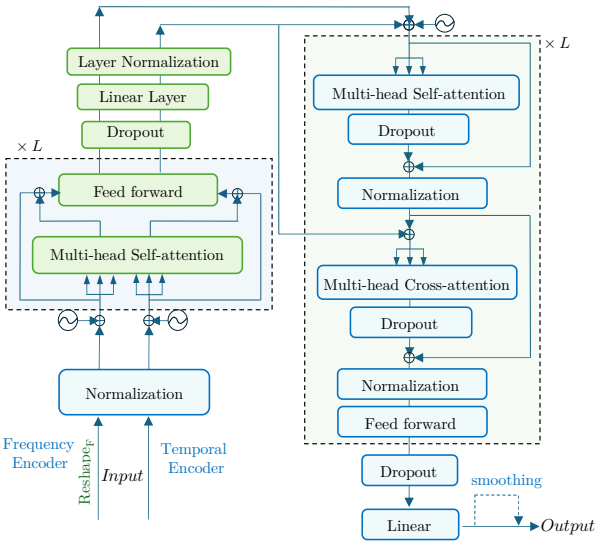

Figure 4: Experimental Dual Encoder Architecture (2E1D): The model consists of separate temporal and frequency encoders that process CQT features independently before fusion for classification.

The wider architecture design trades depth for parallel processing capacity: the 2E1D model generally runs faster than BTC. As shown in Section 5, the wider 2E1D architecture is more susceptible to degradation when adapting to noisy labels compared to the deeper architecture of BTC. Thus, the 2E1D model requires stronger KD regularization to maintain stability.

4. EXPERIMENTS

4.1. Datasets and Preprocessing

Unlabeled Datasets. We use three large-scale unlabeled datasets for pseudo-label generation, totaling over 1,000 hours of audio. The Free Music Archive [27] ($\mathcal{D}_{FMA}$) provides over 100,000 short-form tracks (∼30s) with extensive genre diversity. The MAESTRO dataset [28] ($\mathcal{D}_{MAESTRO}$) contributes approximately 200 hours of high-quality piano recordings with precise MIDI alignment. The DALI dataset [29] ($\mathcal{D}_{DALI}$) supplies over 5,000 full-length music tracks.

Labeled Datasets. We aggregate annotations from the Isophonics [30], the McGill Billboard corpus [31], the RWC Pop dataset [32], and the USPop annotations distributed with [33], collecting a fixed subset of 600 songs. The dataset is then split in a ratio of 7:1:2 into train/validation/test sets (420/60/120 songs). The "50%" and "full" conditions in our continual learning experiments refer to using 210

		mir_eval Metrics							Segmentation			Frame-wise vs. teacher			
Model	**Training Data**	**Root**	**Thirds**	**Triads**	**7ths**	**Tetrads**	**Majmin**	**MIREX**	**Over**	**Under**	**Seg**	**Acc**	**Prec**	**Rec**	**F1**
BTC [1]	Pre-trained weights	81.89	78.49	76.85	66.29	63.72	79.00	78.65	82.16	90.33	80.85	–	–	–	–
Our two student models trained only with pseudo-labels (generated by the BTC teacher from unlabeled data)															
2E1D (2.2M)	FMA (short-form)	77.28±1.3	73.70±1.2	72.15±1.1	61.78±1.4	58.94±1.5	74.49±1.2	73.68±1.3	83.55	83.96	78.03	57.27	34.43	20.29	23.51
	DALI (long-form)	74.32±1.1	65.53±1.3	63.73±1.4	52.68±1.5	50.27±1.4	65.91±1.3	65.37±1.4	77.09	**85.92**	74.00	78.25	48.06	37.41	39.52
	MAESTRO+DALI	79.50±1.0	75.80±1.0	74.11±1.1	63.18±1.3	60.28±1.2	76.53±1.0	75.50±1.1	83.14	84.85	78.53	79.44	52.23	41.95	44.50
	All (FMA+M+D)	**80.37±1.2**	**76.63±1.1**	**74.91±1.2**	**64.37±1.4**	**61.50±1.3**	**77.29±1.1**	**76.35±1.2**	**84.34**	85.63	**80.15**	80.15	57.01	45.20	47.49
	All + KD (α=0.3)	80.17±1.0	76.36±1.0	74.68±1.0	63.93±1.3	61.06±1.4	77.09±1.0	76.08±1.0	84.23	85.27	79.69	**81.23**	**58.14**	**46.33**	**48.62**
BTC (3.03M)	FMA (short-form)	78.85±1.2	74.76±1.0	73.12±1.1	62.88±1.3	60.09±1.4	75.37±1.0	75.07±1.1	80.92	87.57	79.44	61.12	36.53	24.11	26.37
	DALI (long-form)	81.01±1.0	77.24±0.9	75.35±1.0	65.46±1.2	62.59±1.1	77.66±0.9	77.14±1.0	79.50	90.60	79.51	85.88	49.23	50.37	47.32
	MAESTRO+DALI	80.65±1.1	76.37±1.0	74.61±1.2	64.19±1.0	61.35±1.3	76.93±0.9	76.69±1.0	77.42	**90.67**	77.66	88.01	58.84	**64.29**	59.16
	All (FMA+M+D)	**81.54±0.9**	**77.86±0.7**	**76.08±0.8**	**66.29±1.1**	**63.54±1.0**	**78.29±0.7**	**77.84±0.9**	**80.97**	90.15	**80.76**	89.14	62.54	63.89	59.34
	All + KD (α=0.3)	81.23±1.0	77.75±0.8	75.94±0.9	66.00±1.0	63.22±1.1	78.15±0.8	77.82±1.0	80.81	90.25	80.60	**89.47**	**63.21**	64.05	**60.12**

Table 1: Pseudo-labeling results across dataset configurations. The mir_eval metrics are computed against the ground-truth test set, while frame-wise accuracy, precision, recall, and F1 measure agreement with the teacher's predictions. FMA contains short-form clips (∼30s), while DALI and MAESTRO provide long-form full tracks. The pre-trained BTC teacher [1] serves as the reference/upper bound on mir_eval metrics. Best results for BTC and 2E1D student models in the table are highlighted in blue and green, respectively.

and 420 training songs from this subset, respectively.

Clean vs. Noisy Annotations. To evaluate KD's robustness to annotation noise, we prepare two versions of the labeled data: (1) *clean labels* with careful manual alignment, and (2) *noisy labels* sourced online without alignment correction, which primarily affect non-chord ("N") label boundaries. This setup enables an ablation study of KD's regularization effect under different noise conditions.

Preprocessing. All audio undergoes identical preprocessing. The CQT features [34] are extracted with $F = 144$ frequency bins, 24 bins per octave, and a hop length of $h = 2048$ samples, yielding a temporal resolution of $\Delta t \approx 93\,\text{ms}$ at sampling rate $f_{sr} = 22.05\,\text{kHz}$. Features undergo z-score normalization using teacher model statistics (μ_t, σ_t) to maintain identical input distributions.

4.2. Training Configuration

In the first stage, pseudo-labeling training employs the AdamW optimizer with a batch size of 256 and a sequence length of 108 frames (∼10 seconds). The learning rate is warmed up from 10^{-4} to a peak of $3 \cdot 10^{-4}$ over 10 epochs, followed by cosine-annealed decay. Early stopping monitors validation accuracy with a patience of 10 epochs. We reserve 10% of the pseudo-labeled data for validation and 10% as a held-out test set. In the second stage, continual learning, we use a reduced learning rate of 10^{-5} with decay upon observing a validation plateau. KD weight α=0.3 balances adaptation and regularization. The temperature τ=3.0 is empirically selected as the optimal value in all settings. Training continues until early stopping triggers. No data augmentation is applied in our two-stage pipeline (Section 3.2.1).

For the supervised learning (SL) baseline, both BTC and 2E1D are trained from scratch on the full labeled training set (420 songs). The SL baseline uses the same configuration as in pseudo-labeling training, except for the learning schedule. The learning rate decays by a factor of 0.9 when validation accuracy does not improve. Pitch-shifting augmentation via the Rubber Band library transposes both audio and labels from -5 to $+6$ semitones.

4.3. Evaluation Metrics

We employ standard metrics from the mir_eval library [10]. The Root metric compares the root note. The Thirds metric adds major and minor third intervals. The Triads metric evaluates all triadic qualities, while Majmin focuses on major and minor qualities. The Sevenths metric measures a predefined set of seventh chords. The Tetrads metric extends evaluation to four tones. MIREX considers an estimation accurate when at least three pitch classes are correct. Frame-wise accuracy (Acc), precision (Prec), recall (Rec), and F1 are computed using standard True Positive, False Positive, and False Negative counts. Additionally, we report Chord Symbol Recall (CSR) and Weighted Chord Symbol Recall (WCSR) following [5]:

$$\text{CSR}_{c,i} = \frac{|S^{\text{pred}}_{c,i} \cap S^{\text{ref}}_{c,i}|}{|S^{\text{ref}}_{c,i}|}, \quad \text{WCSR}_c = \frac{\sum_i T_i \cdot \text{CSR}_{c,i}}{\sum_i T_i}, \quad (9)$$

where, for a chord class c and track index i, $S^{\text{pred}}_{c,i}$ denotes predicted chord labels, $S^{\text{ref}}_{c,i}$ denotes ground-truth labels for the respective intervals, and T_i is the duration of the i-th track. The Average Chord Quality Accuracy (ACQA), with the chord set $\mathcal{V}$, is calculated as:

$$\text{ACQA} = \frac{\sum_{c \in \mathcal{V}} \text{WCSR}_c}{|\mathcal{V}|}. \quad (10)$$

WCSR weights accuracy by class distribution, while ACQA gives equal weight to all chord qualities. This makes ACQA more sensitive to rare chord performance. Finally, we report segmentation metrics using the standard library implementation [10]. The over-segmentation score (Over) measures how well estimated boundaries avoid fragmenting the ground-truth reference segments, while the under-segmentation score (Under) measures how well predicted segments avoid merging them; higher values indicate better boundary agreement. The overall segmentation score (Seg) is the minimum of the two scores, macro-averaged across tracks.

5. RESULTS

In this section, we present results for our two-stage training method across the metrics described in Section 4.3, including an ablation study on the KD regularization against noisy labels. All reported metrics are evaluated on the held-out test set (120 songs) from the clean labeled dataset described in Section 4.1.

Method	Root	Thirds	Triads	7ths	Tetrads	Majmin	MIREX	Seg	WCSR	ACQA
Prior Work (Noisy Student Framework)										
Bortolozzo et al. [5] †	–	–	–	–	–	–	–	–	46.8	24.5
Li et al. [6] †	78.05	73.56	70.71	61.11	56.84	74.64	74.26	–	–	–
Traditional supervised learning (SL) baseline (trained on full labels with data augmentation)										
2E1D (SL) (2.2M)	79.39±1.4	76.14±1.2	74.53±0.6	64.53±1.9	61.67±1.0	76.89±1.4	76.24±0.7	78.50±1.2	68.7±1.0	24.4±1.5
BTC (SL) (3.03M)	81.52±1.2	78.00±1.0	76.12±0.5	65.93±1.7	63.44±0.9	78.11±1.2	77.79±0.9	79.38±1.1	68.6±1.1	29.0±1.3
Results of our student models after Stage 2 with Data-Incremental Continual Learning (with clean labels, α=0.3)										
2E1D (CL) (50%)	81.02±0.08	77.54±0.18	75.76±0.20	66.39±0.18	63.68±0.23	77.93±0.17	77.57±0.09	80.23±0.21	70.4±0.3	34.2±0.6
2E1D (CL) (full)	**81.55±0.05**	**78.52±0.12**	**76.85±0.13**	**68.19±0.12**	**65.49±0.15**	**78.98±0.11**	**78.52±0.06**	**80.73±0.14**	**71.9±0.2**	**36.1±0.4**
BTC (CL) (50%)	82.14±0.06	78.58±0.17	76.67±0.18	66.77±0.15	64.28±0.20	78.62±0.15	78.87±0.08	81.17±0.18	68.7±0.2	37.1±0.6
BTC (CL) (full)	**83.03±0.04**	**80.17±0.11**	**78.33±0.12**	**69.38±0.10**	**67.00±0.13**	**80.24±0.10**	**80.16±0.05**	**81.71±0.12**	**71.6±0.1**	**39.5±0.4**

Table 2: Data-incremental continual learning results with overall evaluation metrics. After pseudo-labeling, we fine-tune on ground-truth labels with KD (α=0.3). Results for the student models are reported as the average and standard deviation (Mean ± Std). Best results for BTC and 2E1D are highlighted in blue and green, respectively. † Evaluated on our test split using our re-implementation (reporting only the metrics compatible with the model's prediction vocabulary).

5.1. Stage 1: Training with pseudo-labels

Increasing unlabeled data diversity consistently improves performance across the `mir_eval` metric hierarchy and segmentation quality (Table 1). Training on all three datasets (FMA, MAESTRO, DALI) yields the strongest results for both architectures, indicating that pseudo-label pre-training benefits from coverage of varied genres and recording conditions. Long-form datasets consisting of full-length tracks (DALI, MAESTRO) produce more stable pseudo-labels than short-form clips (FMA): DALI achieves higher frame-wise agreement with the teacher than FMA. Longer musical contexts provide consistent harmonic progressions, reducing boundary jitter and spurious chord predictions. However, FMA achieves better `mir_eval` results than DALI for 2E1D due to its larger size. The best BTC student reaches about 99% of teacher performance on ground-truth evaluation across all `mir_eval` metrics, while the purely transformer-based 2E1D achieves 96–98% of the teacher model results, demonstrating cross-architecture knowledge transfer. When all training targets are pseudo-labels, adding KD better aligns the student to the teacher's soft-label distribution, resulting in improvements across frame-wise metrics despite a decrease in `mir_eval` metrics. In addition, KD accelerates optimization, as all pseudo-label training runs using KD converge in 30–40 epochs versus 50–70 for other settings; this comes with a modest training-time memory overhead to store and backpropagate full soft targets.

5.2. Stage 2: Continually training with GT + KD

In Stage 2, the student weights from Stage 1 pseudo-label pre-training are continually trained on ground-truth labeled data with selective KD (α=0.3). BTC with full labels consistently surpasses both the teacher (Table 1) and the supervised learning (SL) baseline across all seven `mir_eval` metrics (Table 2). The 2E1D student similarly improves over its SL baseline across the board. Among the recognition hierarchy, 7ths and Tetrads benefit the most, indicating that combining broad pseudo-label pre-training with targeted ground-truth fine-tuning is particularly effective for complex chords. For comparison with prior semi-supervised approaches [5, 6], we re-implemented their training procedures as described in the respective original publications and evaluated the resulting models on our held-out test split under identical preprocessing settings. Prior work relies on weaker supervised models to generate pseudo-labels, constraining label quality; in contrast, leveraging a highly capable teacher in Stage 1 provides a stronger initialization for Stage 2 adaptation. Furthermore, we observe a substantial drop in seed-to-seed variance in Stage 2 (±0.04 to 0.23) compared to Stage 1 (±0.7 to 1.5) on the `mir_eval` metrics. This stability arises because Stage 2 optimizes over unambiguous ground-truth labels rather than noisy pseudo-labels, while selective KD further acts as a regularizer to prevent the model from overfitting to noise in the smaller labeled dataset.

Notably, the BTC student model is capable of surpassing the teacher model that generated its pre-training pseudo-labels. We attribute this to two complementary mechanisms. First, in Stage 1, the student is trained on a substantially larger unlabeled dataset than the labeled data available to the teacher. The larger amount of data allows the student to approximate the teacher's underlying knowledge more accurately. Second, in Stage 2, ground-truth annotations allow the student to correct systematic errors inherited from the teacher, particularly on rare chord qualities where the teacher's predictions are least reliable. Additionally, selective KD simultaneously anchors the student to the teacher's well-calibrated representations on common chords, preventing overfitting to the small labeled set. The combination of broader pre-training coverage and targeted error correction thus enables the student to exceed the teacher's overall performance.

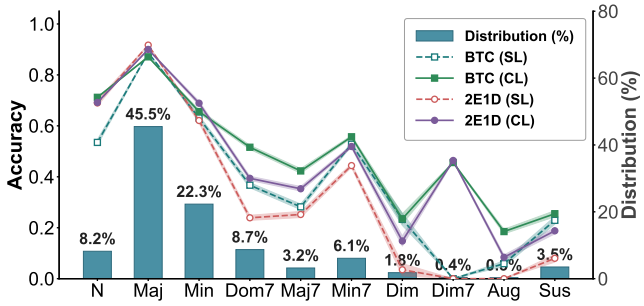

Figure 5: Chord quality distribution (bars) and recognition accuracy (lines) of models trained with supervised learning (BTC (SL), 2E1D (SL)) versus our approach (BTC (CL), 2E1D (CL)).

Figure 5 provides a per-quality view of this improvement and serves as a central illustration of our approach. The training distribution is heavily imbalanced: Major and Minor chords dominate the training frames, while rare qualities (Dim, Dim7, Aug, Sus) constitute less than 3%. The SL baselines achieve 0% accuracy on Dim7 and 0% (2E1D) or 5.9% (BTC) on Aug due to insufficient training examples; our pipeline yields measurable improvements on these classes. The absolute test-sample counts reported in Table 3 (e.g., 212 frames for Dim7, 517 for Aug) confirm that these gains reflect consistent frame-level corrections rather than statistical noise. The ACQA metric, which weights all qualities equally, captures this disproportionate rare-chord improvement more clearly than distribution-weighted WCSR. The pre-training on pseudo-labels also explains the sample-efficiency effect: the student trained on only 50% of labeled data already outperforms the supervised baseline trained on the full dataset (Table 2).

Method	N	Maj	Min	Dom7	Maj7	Min7	Dim	Dim7	Aug	Sus
Count	14.7k	137k	36.4k	32.6k	11.4k	25.3k	540	212	517	5.6k
Bortolozzo et al. [5]†	–	55.5	54.4	51.0	12.0	47.5	30.4	43.3	–	–
2E1D (SL)	69.0	**91.7**	62.1	23.9	25.2	44.4	3.5	0.0	0.0	8.0
BTC (SL)	53.5	**88.7**	63.1	36.7	28.2	52.8	23.3	0.0	5.9	22.9
2E1D (CL) (50%)	69.2	90.9	63.0	33.3	29.5	49.0	12.7	**46.7**	5.2	16.6
2E1D (CL) (full)	**69.2**	90.0	**68.9**	**39.4**	**35.3**	**51.9**	**14.8**	46.4	**8.4**	**18.8**
BTC (CL) (50%)	71.2	84.5	61.2	50.2	42.1	51.6	18.3	**47.4**	14.8	22.3
BTC (CL) (full)	**71.2**	87.1	**65.6**	**51.6**	**42.3**	**55.7**	**23.5**	45.6	**18.5**	**25.5**

Table 3: Subset chord quality accuracy results after Stage 2 with Data-Incremental Continual Learning (α=0.3). Best results for BTC and 2E1D are highlighted in blue and green, respectively. †Evaluated on our test split using our re-implementation.

KD as regularization under label noise. The above results use clean, manually aligned ground-truth labels. To isolate KD's role as a regularizer, we repeat Stage 2 using the same Stage 1 initialized weights but with noisy labels sourced online without alignment correction, which primarily affect non-chord ("N") label boundaries. Table 4 reports results across varying α values. Without KD (α=0), both architectures degrade substantially: BTC drops consistently across all metrics, while the wider 2E1D collapses more severely. Increasing α progressively recovers performance; BTC peaks at α=0.3, while 2E1D requires stronger regularization (α=0.5) to stabilize. Figure 6 shows the corresponding training dynamics: without KD, validation loss rises as the model overfits to noisy labels, whereas $\alpha > 0$ anchors the student to the teacher's distribution, preserving the pseudo-label knowledge acquired in Stage 1. Crucially, when labels are clean (Table 2), KD does not impede adaptation. This confirms that KD selectively mitigates noise while preserving adaptation capacity.

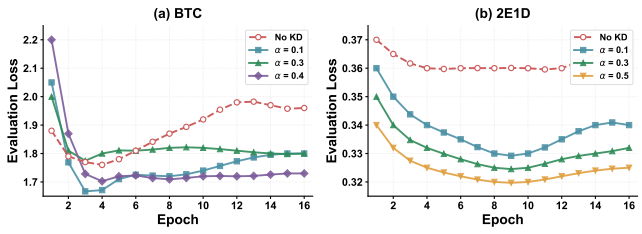

Figure 6: Evaluation loss of student models during continual training with different KD weights α.

Model	α	Root	Thirds	Triads	7ths	Tetrads	Majmin	MIREX	Seg
2E1D	0	52.47	50.70	49.88	42.52	41.00	51.12	50.99	66.72
	0.1	55.85	54.16	53.41	47.68	45.78	54.88	54.07	67.70
	0.3	66.05	63.89	62.79	55.78	53.39	64.70	63.87	73.32
	0.5	**74.21**	**71.25**	**69.78**	**61.20**	**58.35**	**72.09**	**71.54**	**76.84**
BTC	0	71.06	68.05	66.73	58.11	55.66	68.69	68.20	77.05
	0.1	73.88	70.86	69.43	60.91	58.25	71.52	70.96	78.59
	0.3	**77.34**	**73.95**	**72.44**	**62.97**	**60.25**	**74.60**	**74.29**	**79.61**
	0.4	76.95	72.88	71.30	61.30	58.57	73.59	73.18	78.98

Table 4: KD regularization effect during continual training with misaligned labels. The KD weight α controls the balance between teacher soft targets and ground-truth hard labels. Best results for BTC and 2E1D are highlighted in blue and green, respectively.

6. CONCLUSION

Since model weights are often more readily available than proprietary training data, we present a practical training strategy for the ACR problem that leverages open-weight pre-trained models when high-quality labels are scarce. We show that students trained solely on pseudo-labels can approach teacher-level performance across seven `mir_eval` metrics. We further demonstrate that continual learning can improve performance without catastrophic forgetting when the teacher provides sufficiently general representations. Under our pipeline, the best student model ultimately surpasses the teacher, with major gains on rare chord qualities (e.g., Dim, Dim7, Aug). Knowledge distillation improves robustness to noisy labels while preserving adaptability when ground-truth annotations are clean. We also find that the wider 2E1D architecture requires stronger KD regularization than the deeper BTC, underscoring how architectural choices influence continual-learning stability. A key limitation is reliance on teacher quality: biased or weakly generalizable teacher representations can transfer these shortcomings to the student. Future work will explore stronger teacher models and model architectures, ensembles of multiple teachers, scaling to additional unlabeled corpora, and extending the framework to related MIR tasks such as beat tracking and key estimation.

7. REFERENCES

[1] J. Park, K. Choi, S. Jeon, D. Kim, and J. Park, "A Bi-directional Transformer for Musical Chord Recognition," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Delft, The Netherlands, 2019, pp. 620–627.

[2] E. J. Humphrey and J. P. Bello, "Four Timely Insights on Automatic Chord Estimation," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Málaga, Spain, Oct. 2015, pp. 673–679.

[3] C. Harte, "Towards Automatic Extraction of Harmony Information from Music Signals," Ph.D. dissertation, Queen Mary, University of London, Aug. 2010.

[4] J. Pauwels, K. O'Hanlon, E. Gómez, and M. B. Sandler, "20 Years of Automatic Chord Recognition from Audio," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Delft, The Netherlands, Nov. 2019, pp. 54–63.

[5] M. Bortolozzo, R. Schramm, and C. R. Jung, "Improving the Classification of Rare Chords With Unlabeled Data," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 3390–3394.

[6] C. Li, J. Jiang, Y. Li, and L. Tian, "Large-Vocabulary Chord Recognition Based on Contrastive Learning and Noisy Student," *IEEE Transactions on Consumer Electronics*, vol. 71, no. 2, pp. 3695–3706, May 2025.

[7] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *ICML Workshop on Challenges in Representation Learning*, 2013.

[8] G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," in *NIPS Deep Learning and Representation Learning Workshop*, 2014.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 5998–6008.

[10] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis, "mir_eval: A transparent implementation of common MIR metrics," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2014, pp. 367–372.

[11] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with Noisy Student improves ImageNet classification," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 687–10 698.

[12] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 596–608.

[13] A. Tarvainen and H. Valpola, "Mean Teachers are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Learning Results," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 1195–1204.

[14] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta Pseudo Labels," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 557–11 568.

[15] T. Likhomanenko, R. Collobert, N. Jaitly, and S. Bengio, "Continuous Soft Pseudo-Labeling in ASR," in *Proceedings of I Can't Believe It's Not Better! – Understanding Deep Learning Through Empirical Falsification at NeurIPS 2022 Workshops*, ser. Proceedings of Machine Learning Research, vol. 187, 2023, pp. 66–84.

[16] S. Strahl and M. Müller, "Semi-Supervised Piano Transcription Using Pseudo-Labeling Techniques," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, San Francisco, CA, USA, 2024, pp. 173–181.

[17] Y.-N. Hung, J.-C. Wang, M. Won, and D. Le, "Scaling Up Music Information Retrieval Training with Semi-Supervised Learning," *arXiv preprint arXiv:2310.01353*, 2023. [Online]. Available: https://arxiv.org/abs/2310.01353

[18] H. Yuan, Y. Shi, N. Xu, X. Yang, X. Geng, and Y. Rui, "Learning From Biased Soft Labels," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[19] D. Chen, "A Note on Knowledge Distillation Loss Function for Object Classification," arXiv preprint arXiv:2109.06458, Sep. 2021.

[20] A. M. Mansourian, R. Ahmadi, M. Ghafouri, A. M. Babaei, E. B. Golezani, Z. yasamani ghamchi, V. Ramezanian, A. Taherian, K. Dinashi, A. Miri, and S. Kasaei, "A comprehensive survey on knowledge distillation," *Transactions on Machine Learning Research*, 2025.

[21] Z. Li and D. Hoiem, "Learning without Forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, Dec. 2018.

[22] R. M. French, "Catastrophic Forgetting in Connectionist Networks," *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, Apr. 1999.

[23] D. Stefani, S. Peroni, and L. Turchet, "A Comparison of Deep Learning Inference Engines for Embedded Real-Time Audio Classification," in *Proc. Int. Conf. on Digital Audio Effects (DAFx)*, 2022, pp. 256–263.

[24] C. J. Clarke and J. Chowdhury, "Inference-Time Structured Pruning for Real-Time Neural Network Audio Effects," in *Proc. Int. Conf. on Digital Audio Effects (DAFx)*, 2025, pp. 358–365.

[25] G. M. van de Ven and A. S. Tolias, "Three Scenarios for Continual Learning," *arXiv preprint arXiv:1904.07734*, 2019.

[26] J. Driedger and M. Müller, "A Review of Time-Scale Modification of Music Signals," *Applied Sciences*, vol. 6, no. 2, p. 57, 2016.

[27] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A Dataset for Music Analysis," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2017, pp. 316–323. [Online]. Available: https://arxiv.org/abs/1612.01840

[28] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C.-Z. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck, "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2019.

[29] G. Meseguer-Brocal, A. Cohen-Hadria, and G. Peeters, "DALI: A Large Dataset of Synchronized Audio, Lyrics and Notes, Automatically Created Using Teacher-Student Machine Learning Paradigm," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Paris, France, 2018.

[30] M. Mauch, C. Cannam, M. Davies, S. Dixon, C. Harte, S. Kolozali, D. Tidhar, and M. Sandler, "OMRAS2 Metadata Project 2009," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Kobe, Japan, Oct. 2009.

[31] J. A. Burgoyne, J. Wild, and I. Fujinaga, "An Expert Ground Truth Set for Audio Chord Recognition and Music Analysis," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Miami, Florida, USA, Oct. 2011, pp. 633–638.

[32] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC Music Database: Popular, Classical and Jazz Music Databases," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Paris, France, Oct. 2002, pp. 287–288.

[33] B. McFee and J. P. Bello, "Structured Training for Large-Vocabulary Chord Recognition," in *Proc. Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Suzhou, China, Oct. 2017, pp. 188–194.

[34] C. Schörkhuber and A. Klapuri, "Constant-Q Transform Toolbox for Music Processing," in *Proc. Sound and Music Computing Conf. (SMC)*, Barcelona, Spain, Jul. 2010.