# UFO2: A UNIFIED PRE-TRAINING FRAMEWORK FOR ONLINE AND OFFLINE SPEECH RECOGNITION

*Li Fu, Siqi Li, Qingtao Li, Liping Deng, Fangzhu Li, Lu Fan, Meng Chen, Xiaodong He*

JD AI Research, Beijing, China

## ABSTRACT

In this paper, we propose a Unified pre-training Framework for Online and Offline (UFO2) Automatic Speech Recognition (ASR), which 1) simplifies the two separate training workflows for online and offline modes into one process, and 2) improves the Word Error Rate (WER) performance with limited utterance annotating. Specifically, we extend the conventional offline-mode Self-Supervised Learning (SSL)-based ASR approach to a unified manner, where the model training is conditioned on both the full-context and dynamic-chunked inputs. To enhance the pre-trained representation model, stop-gradient operation is applied to decouple the online-mode objectives to the quantizer. Moreover, in both the pre-training and the downstream fine-tuning stages, joint losses are proposed to train the unified model with full-weight sharing for the two modes. Experimental results on the LibriSpeech dataset show that UFO2 outperforms the SSL-based baseline method by 29.7% and 18.2% relative WER reduction in offline and online modes, respectively.

***Index Terms*—** Automatic speech recognition, self-supervised learning, online and offline unified model

## 1. INTRODUCTION

In recent years, Self-Supervised Learning (SSL) has received much attention in the Automatic Speech Recognition (ASR) domain [1–7]. Generally, the SSL-based ASR approach first pre-trains a speech representation encoder on numerous unlabeled utterances via self-supervised strategies (e.g. masking, quantization, contrasting [1]), and then fine-tunes the model on labeled data with ASR objectives. It has shown great potential in improving the ASR performance with limited speech labeling, which is very promising and valuable when the human-annotated utterances are expensive or scarce [8].

Based on how the tokens are emitted, ASR systems are typically categorized by their use in: 1) *online mode* (a.k.a. streaming), which is developed to emit each hypothesized word as quickly and accurately as possible when it is spoken [9–12], and 2) *offline mode* (a.k.a. non-streaming), which aims to accurately emit the complete hypotheses after processing a full utterance [13–15]. However, most existing SSL-based ASR methods focus on the pre-training in an offline manner, i.e. each represented feature is conditioned on the full-context inputs [16]. As for the downstream online ASR model that no (or limited) future context is permitted, the accuracy performance might be hindered due to the mode inconsistency between the pre-training and fine-tuning [17]. One might pre-train the representation encoder in online manners, while the model might suffer a heavy burden on the representation learning when a large proportion of the utterance (i.e. future context) is unavailable [18].

Different from the offline-mode SSL, there are only few works about pre-training for online ASR models. Chiu et al. [17] explored replacing the learnable quantizer in [1] by a Random-projection Quantizer (RQ), to make the quantized representation independent to the recognition mode. Although the RQ method was separately evaluated on online and offline models with 0.6 billion parameters, as mentioned in [17] the random strategy of the quantizer and smaller model sizes that are more efficient for online ASR tasks still need to be further investigated. Cao et al. [19] trained an SSL-based offline ASR model (teacher model), and then adopted knowledge distillation [20] to guide the fine-tuning of an online model (student model). Nevertheless, besides introducing the additional offline model optimization and distillation strategies under the SSL framework, the online model to be fine-tuned was still initialized from an offline-mode representation encoder. Moreover, in the existing SSL-based works, the online and offline ASR systems were processed separately, causing high costs in model development and training workflows for applications in different modes [21–23].

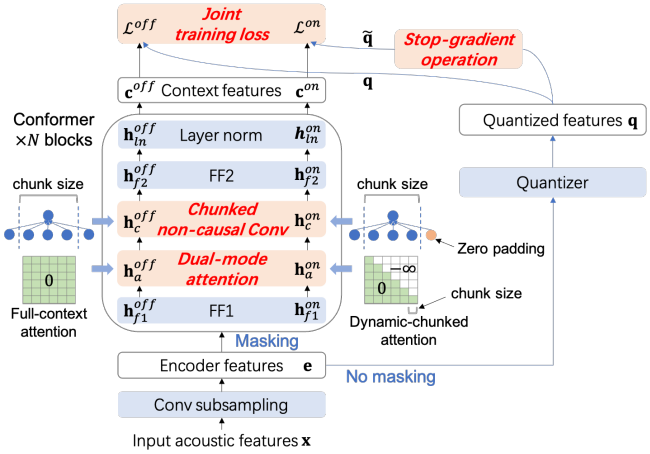


**Fig. 1**. Overview of the proposed UFO2: Encoder features $\mathbf{e}$ are masked and fed to Conformer blocks to extract offline/online latent features ($\mathbf{h}_*^{off}$/$\mathbf{h}_*^{on}$) conditioned on full-context/dynamic-chunked inputs. Then context features $\mathbf{c}^{off}$/$\mathbf{c}^{on}$ and quantized features $\mathbf{q}$/$\tilde{\mathbf{q}}$ (w/ stop gradient) are used for a joint of losses $\mathcal{L}^{off}$ and $\mathcal{L}^{on}$.

In this paper, a novel Unified pre-training Framework is proposed to improve the speech representation learning for downstream Online and Offline (UFO2) ASR tasks. In particular, UFO2 simplifies the training workflows by unifying the online and offline modes into a single model. As shown in Fig. 1, different from the most representative SSL-based approach – Wav2vec2 [1] and its variants, e.g. Wav2vec2-Conformer [25], we extend the offline-mode approach to a unified manner via four strategies on the feature extraction and training objectives. **1) Dual-mode attention**. To train a unified representation encoder, the full-context Multi-Headed Self-Attention (MHSA) in the Conformer block [24] is used to extract offline-mode features conditioned on the complete utterance. Simultaneously, the dynamic-chunked MHSA [21] is adopted to mimic different latency

**Table 1**. System performance in WER (%) (best performance is marked in bold).

| | Methods | Training dataset | | test clean | | | | test other | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Unlabeled | Labeled | offline | | online | | offline | | online | |
| | | $\boldsymbol{D}_u$ | $\boldsymbol{D}_l$ | CPBS | AR | CPBS | AR | CPBS | AR | CPBS | AR |
| S1 | Conformer-offline [24] | NA | 100h | 8.82 | 8.11 | NA | NA | 23.33 | 22.19 | NA | NA |
| S2 | Wav2vec2-base [1] | 960h | 100h | 6.10 | NA | NA | NA | 13.30 | NA | NA | NA |
| S3 | Wav2vec2-Conformer [25] | 960h | 100h | 5.80 | 5.49 | NA | NA | 13.21 | 12.50 | NA | NA |
| S4 | Conformer-unified [21] | NA | 100h | 10.64 | 9.83 | 11.39 | 10.53 | 26.58 | 25.55 | 28.40 | 27.34 |
| S5 | Wav2vec2-Conformer-unified | 960h | 100h | 7.38 | 6.78 | 8.27 | 7.66 | 18.47 | 17.65 | 21.08 | 20.31 |
| S6 | UFO2 | 960h | 100h | **5.46** | **4.99** | **7.06** | **6.35** | **12.38** | **11.75** | **16.87** | **16.05** |

ranges for online-mode learning. **2) Chunked non-causal Convolution (Conv)**. Instead of using the popular causal-Conv Conformer to build a unified model [21], we leverage non-causal Convs to enhance the local feature extraction for online mode, while the positions in the future chunks are padded with zero to make the latency strictly controlled within the chunk size. **3) Stop-gradient operation.** The online and offline representation models share all the encoder and quantizer weights. However, to further improve the representation, stop gradient is operated to decouple the impact of the online-mode objectives to the quantizer. **4) Joint training loss.** The online and offline objectives are aggregated to train the unified model, which encourages the two modes to promote each other. Experimental results show that UFO2 achieves obvious accuracy improvements in comparison with the baseline methods in both the online and offline modes. Our main contributions are summarized as follows:

- To the best of our knowledge, this is the first work exploring a unified pre-training framework for online and offline ASR.
- We propose a new SSL-based ASR approach, named UFO2, by introducing simple and effective strategies for unified training.
- We show the effectiveness of our method on the LibriSpeech 100 hours experiments with significantly lower Word Error Rates (WERs) compared with the baseline methods.

## 2. RELATED WORK

**Unified ASR in supervised learning.** Recently, it has been shown favorable to unify online and offline ASR systems. Conformer-unified [21] randomly sampled one from the two modes when training a single causal-Conv Conformer model. To improve the model quality, the online and offline ASR losses were joint for dual-mode learning with weight sharing [22]: the non-causal Conv in the Conformer was used for offline feature extraction, while the Conv's right-half weights were masked to mimic a causal Conv for online mode. Inspired by [22], a self-pruning method was proposed to unify a compact sparse on-device online ASR model and a large dense offline model [23]. The existing unified systems are mainly discussed in the supervised setting, while we focus on pre-training the unified model on unlabeled utterances. Note that the strategies in our method could also be extended for the supervised ASR task.

**SSL-based ASR.** To overcome the need for labeled training data, SSL-based ASR has been a hot topic in recent years. Wav2vec [26] adopted contrastive learning to train speech representations and improved the ASR accuracy. To enhance the performance, a quantizer was designed in Vq-Wav2vec [27] to learn discrete speech representations. A further performance gain was achieved in Wav2vec2 [1] by combining masking, quantizer, and contrasting strategies in the pre-training. To optimize the model architecture, Wav2vec2-Conformer [25] was proposed by replacing the Transformer block of Wav2vec2 with the Conformer block. More work about the SSL-based ASR approach can be referred to [8]. However, different from the existing works that assume an offline-mode setting [1, 25] or optimize the offline and online models separately [17, 19], our UFO2 unifies online and offline modes in both the pre-training and fine-tuning stages to simplify the training workflows, and encourages the two modes to promote mutually during the training.

## 3. OUR PROPOSED APPROACH

### 3.1. Problem formulation

**Unlabeled dataset:** $\boldsymbol{D}_u = \{\mathbf{x}^i | i \in \{1, \cdots, N_u\}\}$ with $N_u$ the number of utterances; $\mathbf{x}^i \in \mathbf{R}^{F \times T_i}$ is the $i^{th}$ sample which is a sequence of $F$-dimensional acoustic features with length $T_i$.
**Labeled dataset:** $\boldsymbol{D}_l = \{\mathbf{x}_l^i, \mathbf{y}_l^i | i \in \{1, \cdots, N_l\}\}$ with $N_l$ the number of labeled samples; $\mathbf{y}_l^i \in \mathbf{L}^{U_i}$ is the label sequence (with length $U_i$) of utterance $\mathbf{x}_l^i$, where $\mathbf{L}$ is the finite label character.
**Aim of UFO2:** Leverage the unlabeled dataset $\boldsymbol{D}_u$ to learn speech representations and then fine-tune the model on $\boldsymbol{D}_l$ to obtain a single ASR model that performs well in both the online and offline modes.

### 3.2. Model architecture

Our model mainly contains a Conv subsampling encoder, a Conformer context encoder, and a quantizer module, as shown in Fig. 1. Given input acoustic features $\mathbf{x}$, the Conv subsampling encoder outputs encoder features $\mathbf{e}$ with a $4\times$ reduction in the sequence length, which are then processed in the following two ways. 1) The encoder features are masked on sampled 6.5% time-steps and the 10 consecutive steps [1], and then fed to a stack of $N$ Conformer blocks to obtain offline and online context features $\mathbf{c}^{off}$ and $\mathbf{c}^{on}$ simultaneously. Each Conformer block contains two Feed Forward (FF1, FF2) modules sandwiching the MHSA and non-causal Conv modules [24], where the offline and online latent features ($\mathbf{h}_*^{off}$, $\mathbf{h}_*^{on}$) are extracted to adapt the model for the two modes. 2) The encoder features $\mathbf{e}$ (without masking) are quantized to features $\mathbf{q}$ selected from $G$ codebooks with $V$ entry vectors via the quantizer module [1]. Finally, the model is pre-trained with a self-supervised contrastive loss, which is applied to distinguish the quantized feature at the same time-step of each masked context feature from other masked samples [1].

### 3.3. Strategies for unified model

To enhance the performance of the unified model, four strategies (highlighted in red, in Fig. 1) on the feature extraction and training objectives are proposed as follows.

**1) Dual-mode attention.** One of the key parts for unifying the online and offline modes is to enable the model extracting dual-mode features [22]. In the proposed UFO2, a bias matrix $\mathbf{B}$ (unlearnable) is added to the attention logits [28] $\mathbf{A}$ of the MHSA module, yields the dual-mode attention matrix $\mathbf{A}_d = \mathrm{Softmax}(\mathbf{A} + \mathbf{B})$ as below.

- *Full-context attention:* $\mathbf{B}$ is set to a zero matrix to output the normal attention conditioned on the full utterance for offline mode.
- *Dynamic-chunked attention:* We first segment the inputs into non-overlapping chunks with size $c_s$, and then set the top-right elements of $\mathbf{B}$ to $-\infty$. The $-\infty$ values lead to zero attention scores via the Softmax function, i.e. each online-mode feature cannot attend to frames belonging to the future chunks [29]. To enhance the online-mode learning, $c_s$ is dynamically selected for each training mini-batch to mimic latency ranges from 0 to 1 second [21].

**2) Chunked non-causal Conv.** Typically, causal-Conv [21] is used for Conformer-based online ASR models to extract local features being independent on its future inputs. It mitigates the latency increasing problem when stacking multiple non-causal-Conv Conformer

**Table 2**. Ablation studies with different training losses and model choices in WER (%).

| | Pre-training △: offline loss ✓: joint loss | Conv in Conformer ASR △: causal ✓: chunked non-causal | Fine-tuning △: random-mode loss ✓: joint loss | test clean | | | | test other | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | offline | | online | | offline | | online | |
| | | | | CPBS | AR | CPBS | AR | CPBS | AR | CPBS | AR |
| A1 | △ | △ | △ | 7.38 | 6.78 | 8.27 | 7.66 | 18.47 | 17.65 | 21.08 | 20.31 |
| A2 | △ | △ | ✓ | 6.15 | 5.65 | 8.10 | 7.57 | 14.39 | 13.62 | 19.71 | 18.87 |
| A3 | △ | ✓ | △ | 6.11 | 5.62 | 7.71 | 7.05 | 14.36 | 13.62 | 18.96 | 18.11 |
| A4 | △ | ✓ | ✓ | 5.85 | 5.45 | 7.63 | 7.08 | 14.16 | 13.39 | 18.82 | 18.13 |
| A5 | ✓ | ✓ | ✓ | **5.46** | **4.99** | **7.06** | **6.35** | **12.38** | **11.75** | **16.87** | **16.05** |

blocks [30]. However, the causal-Conv may hinder the dynamic-chunked approach since the future inputs belonging to the current chunk are not used (see Fig. 2). To address this issue, we design a chunked non-causal Conv for the unified model. In online mode, the positions in the future chunks are padded with zero to ensure the latency strictly controlled within the chunk size; the Conv's weights are fully shared to offline mode without chunk restriction. Compared with the causal Conv, the main advantages of the chunked non-causal Conv are: a) more local features within the current chunk are leveraged to enhance the performance; b) model weights are fully shared for the two modes to encourage them to promote mutually [22].


(a) Causal Conv (b) Chunked non-causal Conv

**Fig. 2**. Advantages of chunked non-causal Conv, over causal Conv, in leveraging more local features in online and offline modes.

**3) Stop-gradient operation.** Empirically, an offline representation would outperform an online one, since the former is conditioned on more future context. Considering that the quantizer will be removed in the downstream fine-tuning stage [1], it can be completely turned into offline-mode to enhance the performance during the pre-training. Thus, we apply the stop-gradient operation to $\mathbf{q}$ obtaining $\tilde{\mathbf{q}}$, which decouples the impact of online-mode objectives to the quantizer. Our experiments show that the stop gradient operation largely improves the ASR performance (shown in Sec. 4.4).

**4) Joint training loss.** As for the self-supervised pre-training, the contrastive losses [1] in offline ($\mathcal{L}_{\text{con}}^{off}$) and online ($\mathcal{L}_{\text{con}}^{on}$) modes are proposed and jointly trained as $\mathcal{L}_{\text{con}} = \lambda\mathcal{L}_{\text{con}}^{off} + (1-\lambda)\mathcal{L}_{\text{con}}^{on}$, where $\lambda = 0.5$ assigning equal importance for the two items. As for the downstream task, the ASR model is initialized from the pre-trained encoder and then fine-tuned on the labeled dataset $\boldsymbol{D}_l$ in a unified manner same as the pre-training. In this stage, the masking and quantizer of the pre-training are removed. Instead, a fully connected layer is added to the top of the encoder, and a multi-layer Transformer decoder is applied for the ASR task learning [21]. To train the ASR model, we adopt the representative hybrid loss [31] $\mathcal{L}_{\text{ctc/att}}(\mathbf{x}_l, \mathbf{y}_l) = \epsilon\mathcal{L}_{\text{ctc}}(\mathbf{x}_l, \mathbf{y}_l) + (1-\epsilon)\mathcal{L}_{\text{att}}(\mathbf{x}_l, \mathbf{y}_l)$, with $\mathcal{L}_{\text{ctc}}(\mathbf{x}_l, \mathbf{y}_l)$ the Connectionist Temporal Classification (CTC) loss and $\mathcal{L}_{\text{att}}(\mathbf{x}_l, \mathbf{y}_l)$ the attention-based loss, and with $\epsilon = 0.3$ [21]. Finally, the offline and online hybrid losses ($\mathcal{L}_{\text{ctc/att}}^{off}(\mathbf{x}_l, \mathbf{y}_l)$ and $\mathcal{L}_{\text{ctc/att}}^{on}(\mathbf{x}_l, \mathbf{y}_l)$) are aggregated for training a unified ASR model as

$$\mathcal{L}_{\text{asr}} = \alpha\mathcal{L}_{\text{ctc/att}}^{off}(\mathbf{x}_l, \mathbf{y}_l) + (1-\alpha)\mathcal{L}_{\text{ctc/att}}^{on}(\mathbf{x}_l, \mathbf{y}_l) \quad (1)$$

where $\alpha$ is used to balance the two items (studied in Sec. 4.4).

## 4. EXPERIMENTS AND DISCUSSION

### 4.1. Experimental setup

**Data preparation.** The public LibriSpeech (LS) [32] utterances ($\boldsymbol{D}_u$, 960 hours) and the transcriptions of the train-clean subset ($\boldsymbol{D}_l$, 100 hours) are used in our experiments. The WER performance of the trained models is evaluated on the original test-clean (considered easier) and test-other (considered harder/more noisy) datasets. The 80-dimensional Mel-spectrograms of each speech are pre-processed as the input to the model with 25ms window size and 10ms step size.

**Table 3**. WERs (%) for Causal-Conv (CC) and Chunked-Non-Causal-Conv (CNCC) Conformers in pre-training and fine-tuning.

| | Pre-training (offline) | Fine-tuning (unified) | test clean (AR) | | test other (AR) | |
|---|---|---|---|---|---|---|
| | | | offline | online | offline | online |
| B1 | CNCC | CC | 6.78 | 7.66 | 17.65 | 20.31 |
| B2 | CC | CC | 6.11 | 7.53 | 15.08 | 19.68 |
| B3 | CC | CNCC | 5.90 | 7.47 | 15.04 | 19.82 |
| B4 | CNCC | CNCC | **5.62** | **7.05** | **13.62** | **18.11** |

**Model.** The ASR model consists of 2 Conv sub-sampling layers, 12 Conformer layers for the encoder, and 6 Transformer layers for the decoder [21]. For the Conv sub-sampling layers, the kernel sizes are (3,3) and the strides are 2. All Conformer and Transformer layers have 8 multi-heads and are 512-dimensional; the kernel sizes of the non-causal Convs are 15. In addition, 5003 modeling units are utilized including 5000 Byte-Pair Encoding (BPE) [33] units and 3 non-verbal symbols (blank, unknown-unit, and sos-eos). Following [1], we set the multiple codebooks in the quantizer with groups 2, entries 320 and vector dimension 512. The number of model parameters is about 0.1 billion, which is comparable with the Wav2vec2-base model [1]. Both the pre-training and fine-tuning are optimized with a mini-batch of 96 and using the same learning rate scheduler as [21].

**Decoding.** The models are evaluated with two decoding chunk sizes: 1) $c_s = \infty$ for offline mode; and 2) $c_s = 16$ for online mode. The latency of $c_s = 16$ is uniformly distributed from 0 to 640ms, with an average latency of 320ms [21]. The beam size of the CTC Prefix Beam Search (CPBS) decoding [34] is 10, and the parameters for the Attention Rescore (AR) decoding are the same with the training [21]. Note that no language model is applied in the decoding.

### 4.2. Main experiment

**Baseline methods**[1]. We implement two unified baseline models (with 0.1-billion parameters) consist of: 1) a supervised baseline – Conformer-unified [21]: a State-Of-The-Art (SOTA) unified ASR system using the conventional causal-Conv Conformer; and 2) an SSL-based baseline – Wav2vec2-Conformer-unified: we implement the Conformer-unified model while initializing the encoder network from pre-trained Wav2vec2-Conformer [25].

**System performance.** As shown in Table 1, we compare UFO2 with SOTA supervised and SSL-based ASR approaches in both offline ($c_s = \infty$) and online ($c_s = 16$) modes. As for the systems that are trained for offline uses alone (S1-S3), both the Wav2vec2-base system (S2) [1] and Wav2vec2-Conformer system (S3) [25] largely outperform the supervised Conformer-offline method (S1) [24] due to the benefits from SSL on the 960 hours of unlabeled utterances. Compared with S1, the Conformer-unified system (S4) [21] is compatible with the two modes, while obtaining higher WERs in offline mode due to the use of causal-Conv Conformer and random-

[1]Note that previous works [17, 19] are not fully comparable because they are not unified models. [17] optimized two modes separately on a much larger 60k-hour dataset, while [19] distilled an offline model to the online model.

**Table 4**. An ablation study on the stop gradient in WER (%).

| Methods | test clean (AR) | | test other (AR) | |
|---|---|---|---|---|
| | offline | online | offline | online |
| UFO2 w/o stop-grad | 5.59 | 6.87 | 13.52 | 17.09 |
| UFO2 w/ stop-grad | **4.99** | **6.35** | **11.75** | **16.05** |

**Table 5**. WERs (%) for different fine-tuning hyperparameters.

| Weights of offline mode | test clean (AR) | | test other (AR) | |
|---|---|---|---|---|
| | offline | online | offline | online |
| $\alpha = 0$ | 6.19 | 6.44 | 13.81 | 16.60 |
| $\alpha = 0.25$ | 5.23 | 6.39 | 12.59 | 16.36 |
| $\alpha = 0.5$ | 5.15 | **6.35** | 12.05 | **16.04** |
| $\alpha = 0.75$ | **4.99** | **6.35** | **11.75** | 16.05 |
| $\alpha = 1$ | 5.13 | 15.38 | 12.00 | 25.67 |

mode training. As for the unified ASR method based on SSL, the Wav2vec2-Conformer-unified system (S5) improves the performance compared with S4 via parameters initializing from the pre-trained representation model of S3. Note that since S3 uses non-causal-Conv Conformer, only the left half of the non-causal-Conv kernels are inherited to the downstream S5. Compared with the existing methods, our proposed UFO2 (S6) achieves the best performance with obvious WER reduction in both the two modes. Numerically, compared with the Wav2vec2-Conformer-unified baseline (S5), UFO2 (S6) achieves an average of 29.7% and 18.2% relative WER reduction in offline and online modes, respectively.

### 4.3. Ablation study

As shown in Table 2, ablation studies with different training losses and model choices are conducted by comparing our UFO2 (A5) with the Wav2vec2-Conformer-unified baseline (A1) and the variants (A2-A4). First, when comparing A2 with A1, the joint of online and offline ASR losses outperforms the random-mode strategy in the fine-tuning stage. The finding is consistent with the previous study [22]. Compared with A1 that fine-tunes a causal-Conv Conformer initialized from Wav2vec-Conformer [25], A3 uses the chunked non-causal-Conv Conformer (all parameters of Wav2vec-Conformer are inherited), which enhances the ASR performance by leveraging more local features within the chunk. A4 is the combination of A2 and A3, which achieves a further performance improvement. Note that the A4 method can also be extended to 1) other released pre-trained models since most of the existing SSL-based models are pre-trained in an offline manner [8], and 2) the supervised setting. Finally, UFO2 (A5) combines all of the proposed strategies in both the pre-training and fine-tuning stages, and significantly reduces the WERs compared with the baseline system A1.

### 4.4. Discussion

To further evaluate the effectiveness of UFO2, we analyze the proposed method from the following four perspectives. Due to space limitations, the WER results of the AR decoding are reported for the discussion (We omit the CPBS decoding which is the same trend).

**Chunked non-causal Conv in Conformer.** To analyze the effect of the Conv in Conformer blocks, we conduct three experiments (B2-B4) based on the Wav2vec2-Conformer-unified baseline (B1), with causal/chunked non-causal Convs for the pre-training and fine-tuning stages (shown in Table 3). Compared with B1, B2 obtains lower WERs with the causal-Conv Conformer in pre-training. We infer that the consistency of model structures between the pre-training and fine-tuning would bring more advantage than only using non-causal Conv for pre-training. The performance of B3 is similar to B2, although a chunked non-causal-Conv Conformer (only the left-half Conv kernels are pre-trained) is adopted for the downstream ASR task in B3. A possible reason is that the chunked non-causal-Conv would bring little improvement if it is not fully pre-trained.

**Table 6**. WERs (%) for Fine-Tuning (FT) on LS-960h.

| Methods | Pre-training | FT on LS-960h | | | |
|---|---|---|---|---|---|
| | | LS-test-clean | | LS-test-other | |
| | | offline | online | offline | online |
| Conformer-unified [21] | NA | 3.6 | 4.0 | 9.3 | 11.1 |
| Wav2vec2-base [1] | LS-960h | 3.4 | NA | 8.5 | NA |
| Wav2vec2-Conformer [25] | | 3.1 | NA | 7.4 | NA |
| Wav2vec2-Conformer-unified | | 3.5 | 4.0 | 8.5 | 10.8 |
| UFO2 | | **3.0** | **3.8** | **7.1** | **9.4** |

**Table 7**. WERs (%) for Fine-Tuning (FT) on GS-250h and TL-200h.

| Methods | Pre-training | FT on GS-250h | | | | FT on TL-200h | |
|---|---|---|---|---|---|---|---|
| | | GS-dev | | GS-test | | TL-test | |
| | | offline | online | offline | online | offline | online |
| Conformer-unified [21] | NA | 24.5 | 26.1 | 23.7 | 25.3 | 9.6 | 10.8 |
| Wav2vec2-base [1] | LS-960h | 20.4 | NA | 20.0 | NA | 8.6 | NA |
| Wav2vec2-Conformer [25] | | 18.5 | NA | 18.2 | NA | 6.7 | NA |
| Wav2vec2-Conformer-unified | | 21.3 | 22.6 | 21.0 | 22.2 | 8.3 | 9.3 |
| UFO2 | | **17.8** | **20.3** | **17.4** | **19.8** | **6.0** | **7.7** |

Compared with B1-B3, B4 (proposed in UFO2) achieves the best performance via using chunked non-causal-Conv in both the two training stages, which leverages more local features and also fully inherits the pre-trained weights for the downstream fine-tuning.

**Stop gradient in pre-training.** Inspired by [17], we decouple the online-mode objectives to the quantizer via stop gradient operation in the pre-training, which largely reduces WERs, as shown in Table 4. The results indicate that the model would learn a high quality quantizer with full-context inputs, while the online-mode objectives might degrade the quantized representation due to a heavy burden on the representation learning when a large proportion of the utterance (i.e. future context) is unavailable [18].

**Hyperparameter tuning for fine-tuning.** We analyze the hyperparameter choice for the fine-tuning loss in Eq. 1, which balances the offline and online terms. As shown in Table 5, $\alpha = 0$ and $\alpha = 1$ imply fine-tuning an online model and an offline model respectively, which suffer from high WERs when the decoding mode is inconsistent with the training. $\alpha = 0.25, 0.5, 0.75$ imply fine-tuning unified models, which significantly outperform both the online model ($\alpha = 0$) and offline model ($\alpha = 1$). The results also indicate that our UFO2 achieves the best performance when $\alpha = 0.75$. We infer that the online and offline mode promote each other with the joint training manner: the offline model that usually achieves better results would help the online model learning; while the dynamic-chunked strategies might bring advantage like spectrogram augmentation [35] for the offline model via masking the future context.

**Fine-tuning on large and out-of-domain datasets.** To further verify the effectiveness of our approach, we fine-tune UFO2 on three different datasets: 1) all of the 960-hour LS dataset, 2) 250-hour GigaSpeech (GS) subset-S for quick research experiments [36], and 3) 200-hour TedLium2 (TL) dataset collected from TED talks [37]. As shown in Table 6-7, UFO2 consistently outperforms the baseline systems, with fine-tuning on large and out-of-domain datasets. Numerically, compared with the SSL-based baseline Wav2vec2-Conformer-unified, UFO2 achieves relative WER reductions by 15.3%/9.0%, 16.8%/10.5%, 27.7%/17.2% in offline/online modes when fine-tuning on LS-960h, GS-250h, TL-200h, respectively.

## 5. CONCLUSION

In this paper, a novel framework named UFO2 is proposed to unify the online and offline ASR models based on SSL, which simplifies the two separate training workflows into a single one, and improves the recognition accuracy. Our results on the LibriSpeech dataset show that the proposed UFO2 significantly enhances the performance compared to the baseline methods. However, we also find that the performance in online mode still underperforms the offline mode, which will be further addressed in the future work.

## 6. REFERENCES


[1] A. Baevski, Y. Zhou, A. Mohamed, et al., "Wav2vec2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, 2020.

[2] W. Hsu, B. Bolte, Y. Tsai, et al., "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[3] S. Chen, C. Wang, Z. Chen, et al., "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[4] S. Yang, P. Chi, Y. Chuang, et al., "Superb: Speech processing universal performance benchmark," in *Proc. Interspeech*, 2021.

[5] M. Ravanelli, J. Zhong, S. Pascual, et al., "Multi-task self-supervised learning for robust speech recognition," in *Proc. ICASSP*, 2020.

[6] A. Babu, C. Wang, A. Tjandra, et al., "Xls-r: Self-supervised cross-lingual speech representation learning at scale," in *Proc. Interspeech*, 2022.

[7] C. Gao, G. Cheng, T. Li, et al., "Self-supervised pre-training for attention-based encoder-decoder asr model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1763–1774, 2022.

[8] A. Mohamed, H. Lee, L. Borgholt, et al., "Self-supervised speech representation learning: A review," *arXiv preprint arXiv:2205.10643*, 2022.

[9] H. Miao, G. Cheng, C. Gao, et al., "Transformer-based online ctc/attention end-to-end speech recognition architecture," in *Proc. ICASSP*, 2020.

[10] Y. He, T. Sainath, R. Prabhavalkar, et al., "Streaming end-to-end speech recognition for mobile devices," in *Proc. ICASSP*, 2019.

[11] B. Li, S. Chang, T. Sainath, et al., "Towards fast and accurate streaming end-to-end asr," in *Proc. ICASSP*, 2020.

[12] T. Doutre, W. Han, M. Ma, et al., "Improving streaming automatic speech recognition with non-streaming model distillation on unsupervised data," in *Proc. ICASSP*, 2021.

[13] J. Li, Y. Wu, Y. Gaur, et al., "On the comparison of popular end-to-end models for large scale speech recognition," in *Proc. Interspeech*, 2020.

[14] L. Fu, X. Li, L. Zi, et al., "Incremental learning for end-to-end automatic speech recognition," in *Proc. ASRU*, 2021.

[15] H. Braun, J. Luitjens, R. Leary, et al., "Gpu-accelerated viterbi exact lattice decoder for batched online and offline speech recognition," in *Proc. ICASSP*, 2020.

[16] M. Karimi, C. Liu, K. Kumatani, et al., "Deploying self-supervised learning in the wild for hybrid automatic speech recognition," *arXiv preprint arXiv:2205.08598*, 2022.

[17] C. Chiu, J. Qin, Y. Zhang, et al., "Self-supervised learning with random-projection quantizer for speech recognition," in *Proc. ICML*, 2022.

[18] L. Fu, X. Li, R. Wang, et al., "Scala: Supervised contrastive learning for end-to-end speech recognition," in *Proc. Interspeech*, 2022.

[19] S. Cao, Y. Kang, Y. Fu, et al., "Improving streaming transformer based asr under a framework of self-supervised learning," in *Proc. Interspeech*, 2021.

[20] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. NeurIPS*, 2015.

[21] Z. Yao, D. Wu, X. Wang, et al., "Wenet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit," in *Proc. Interspeech*, 2021.

[22] J. Yu, W. Han, A. Gulati, et al., "Dual-mode asr: Unify and improve streaming asr with full-context modeling," in *Proc. ICLR*, 2021.

[23] C. Liu, Y. Shangguan, H. Yang, et al., "Learning a dual-mode speech recognition model via self-pruning," *arXiv preprint arXiv:2207.11906*, 2022.

[24] A. Gulati, J. Qin, C. Chiu, et al., "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech*, 2020.

[25] Y. Zhang, J. Qin, D. Park, et al., "Pushing the limits of semi-supervised learning for automatic speech recognition," in *Proc. NeurIPS*, 2020.

[26] S. Schneider, A. Baevski, R. Collobert, et al., "Wav2vec: Unsupervised pre-training for speech recognition," in *Proc. Interspeech*, 2019.

[27] A. Baevski, S. Schneider, M. Auli, et al., "Vq-wav2vec: Self-supervised learning of discrete speech representations," in *Proc. ICLR*, 2020.

[28] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," in *Proc. NeurIPS*, 2017.

[29] X. Chen, Y. Wu, Z. Wang, et al., "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," in *Proc. ICASSP*, 2021.

[30] Y. Shi, C. Wu, D. Wang, et al., "Streaming transformer transducer based speech recognition using non-causal convolution," in *Proc. ICASSP*, 2022.

[31] S. Watanabe, T. Hori, S. Kim, et al., "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[32] V. Panayotov, G. Chen, D. Povey, et al., "Librispeech: An asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015.

[33] D. Gowda, A. Garg, K. Kim, et al., "Multi-task multi-resolution char-to-bpe cross-attention decoder for end-to-end speech recognition," in *Proc. Interspeech*, 2019.

[34] A. Graves, S. Fernandez, F. Gomez, et al., "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, 2006.

[35] D. Park, W. Chan, Y. Zhang, et al., "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019.

[36] G. Chen, S. Chai, G. Wang, et al., "Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio," in *Proc. Interspeech*, 2021.

[37] A. Rousseau, P. Deléglise, Y. Esteve, et al., "Enhancing the ted-lium corpus with selected data for language modeling and more ted talks," in *Proc. LREC*, 2014.