Bai du USA

ICLR

W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

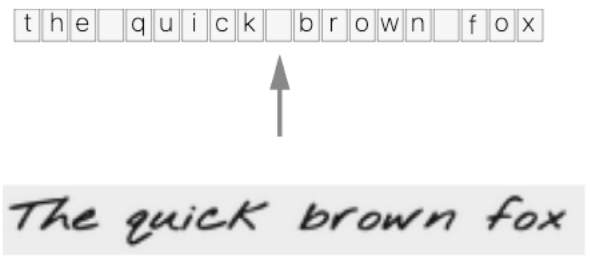
Xingyu Cai, Jiahong Yuan, Yuchen Bian,

Guangxu Xun, Jiaji Huang, Kenneth Church

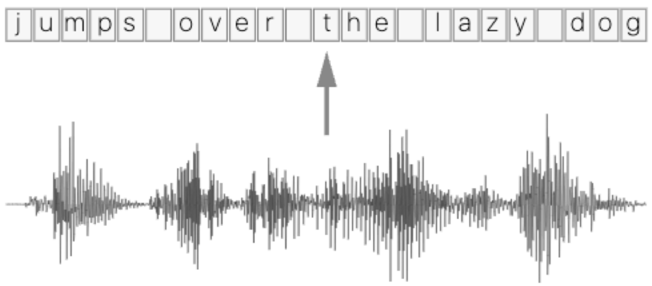
W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

- Connectionist Temporal Classification (CTC) was proposed in [1] to train end- to-end sequence learning models.

- CTC is widely used to train automatic speech recognition (ASR) [2], optical character recognition (OCR) [3], sign language translation [4], and many other tasks.



Handwriting recognition: The input can be (x, y) coordinates of a pen stroke or pixels in an image.



Speech recognition: The input can be a spectrogram or some other frequency based feature extractor.

https://distill.pub/2017/ctc/

- Let X be the input sequence of length T, e.g. audio frames, image segments, etc
- Let Y be the labels of length N.
- Typically T >= N, i.e. many-to-one mapping.

$$\mathcal{L}_{\text{CTC}}(X, Y) = -\log P(Y|X) = -\log \sum_{\pi \in \mathcal{Z}} P_{\pi}(Y|X)$$

[1] Graves, et al, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks", ICML, 2006
[2] Battenberg, et al, "Exploring neural transducers for end-to-end speech recognition", ASRU, 2017
[3] Chen, et al, "Text recognition in the wild: A survey", CSUR, 2021
[4] Camgoz, et al, "Sign language trans- formers: Joint end-to-end sign language recognition and translation", CVPR, 2020

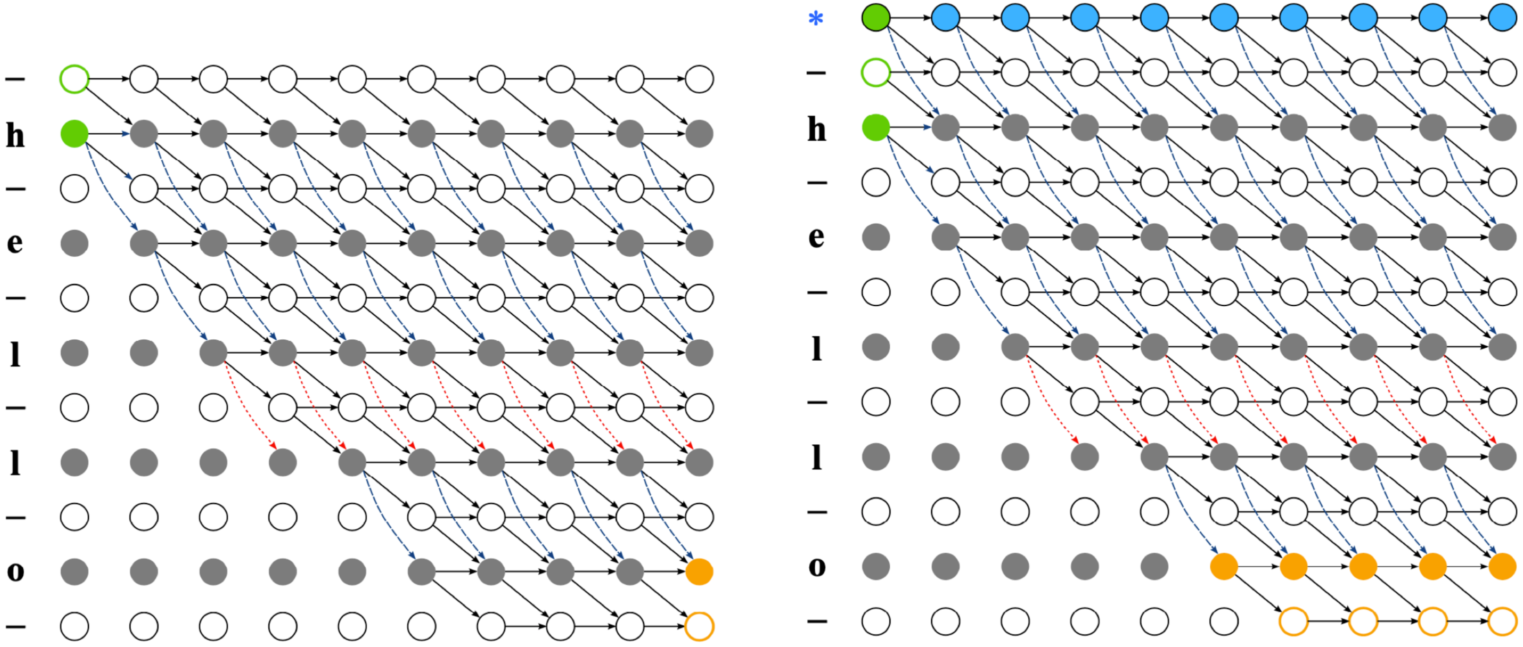
W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

- In this paper, we tackle the incomplete label problem, i.e. Y only matches the middle part of X,

$$\mathcal{L}'_{\text{CTC}}(X, \hat{Y}) = \sigma\{\mathcal{L}_{\text{CTC}}(\hat{X}_{i,\tau}, \hat{Y}) \mid i + \tau < T\}$$

- Standard CTC uses Dynamic Programming (DP). A good tutorial: https://distill.pub/2017/ctc/

- We borrow the idea from DTW literature [1], and propose the **wild card enhanced CTC, W-CTC:**
  - ✓ Prepend a wild card symbol to Y
  - ✓ Making the DP procedure starting with either wild-card or Y
  - ✓ The final states are the entire last two rows rather than the two bottom-right nodes.



(a) Standard CTC: The two green nodes at top-left corner are initial states for DP recursion. The two orange nodes at bottom-right corner are ending nodes for final loss calculation. The black and blue arrows are allowed transitions based on Equation 3. The red arrows are forbidden.

(b) W-CTC: The first row corresponds to the prepended wild-card "*" symbol (the blue nodes). There are three initial states. The ending nodes are the entire last two rows, rather than only two right-bottom nodes. These changes enable $\hat{Y}$ to match only a fraction of X.

Figure 1: Illustration of Dynamic Programming based CTC and proposed W-CTC loss calculation.

[1] Sakurai, et al, "Stream monitoring under the time warping distance", ICDE, 2007

W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

The Key Summary:

- The problem reduces to a classical problem: dynamic programming with unconstraint endpoints.
- Prepend the wild card symbol: Ensure the transcription can start anywhere in speech frames, because P(*|X) = 1. There is no punishment if skipping the beginning frames.
- End with the entire last two rows: Ensure the transcription can end anywhere in speech frames. We combine the last two rows using weighted sum:

$$\mathcal{L}_{\text{W-CTC}} = \sum_{N-1}^{T-1} w_j \mathcal{L}_{\text{CTC}}^{(j)} \ \text{, where } [w_{N-1}, \ldots, w_{T-1}] = \text{softmax}\left([-\mathcal{L}_{\text{CTC}}^{(N-1)}, \ldots, -\mathcal{L}_{\text{CTC}}^{(T-1)}]\right)$$

NOTE:

- The wild card symbol “*”, is **NOT equivalent** to the ”blank” symbol in CTC. “blank” has physical meaning, e.g. silence, noise, such that the model still needs to predict P(blank|X). But wild card assigns probability 1 regardless of X.
- The method only solves missing Y problem, but cannot solve missing X cases.
- The remaining part of Y, must satisfy (taking “hello” as an example):
  - ➢ Corresponds to the middle part of X: “?ell?” YES “h???o” NO
  - ➢ Must be continuous subsequence: ”?el??” YES ”?e?l?” NO

W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

- Comprehensive experiments on automatic speech recognition (ASR), optical character recognition (OCR) and continuous sign language recognition (CSLR) are carried out.
- By randomly masking part of the label, we obtain the corrupted labels and train the models with complete X, but fraction of Y. We study the model performance vs masking ratio.

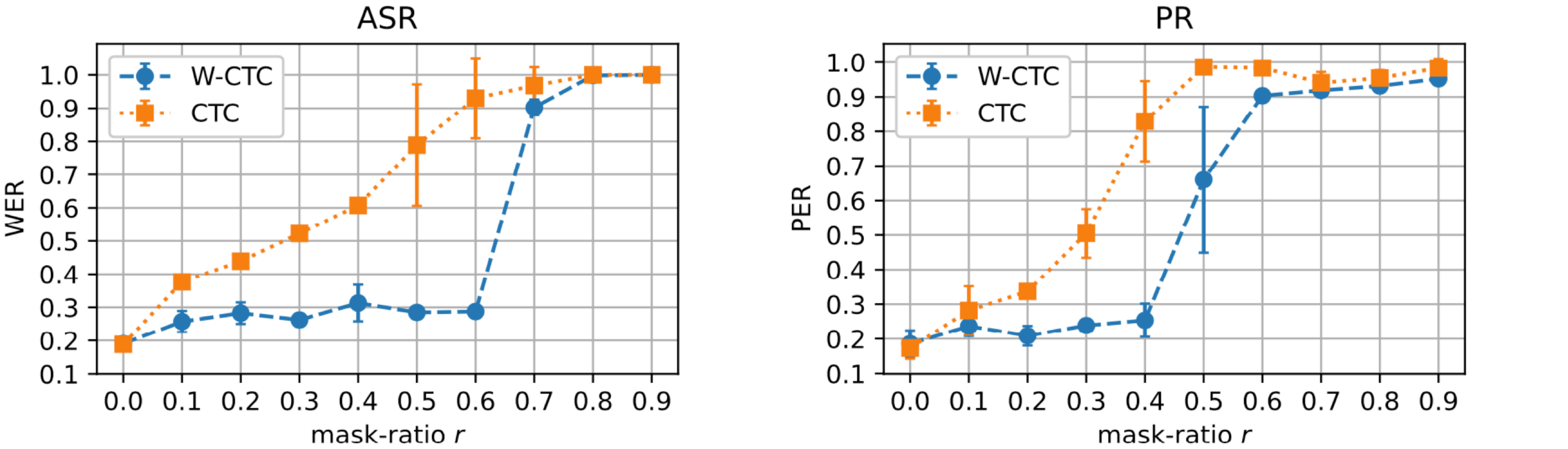
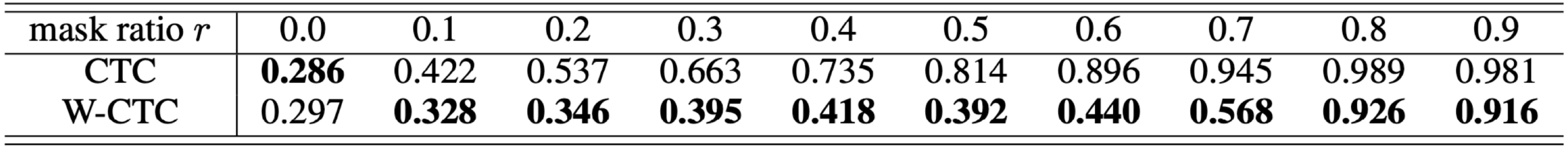


Figure 2: WER / PER vs mask-ratio in ASR and PR tasks, on TIMIT test set.

Table 1: WER on CSLR task. W-CTC significantly outperforms CTC when label corrupted.



mask ratio r	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
CTC	**0.286**	0.422	0.537	0.663	0.735	0.814	0.896	0.945	0.989	0.981
W-CTC	0.297	**0.328**	**0.346**	**0.395**	**0.418**	**0.392**	**0.440**	**0.568**	**0.926**	**0.916**

W-CTC: a Connectionist Temporal Classification Loss with Wild Cards

OCR Experiments and illustrative examples:

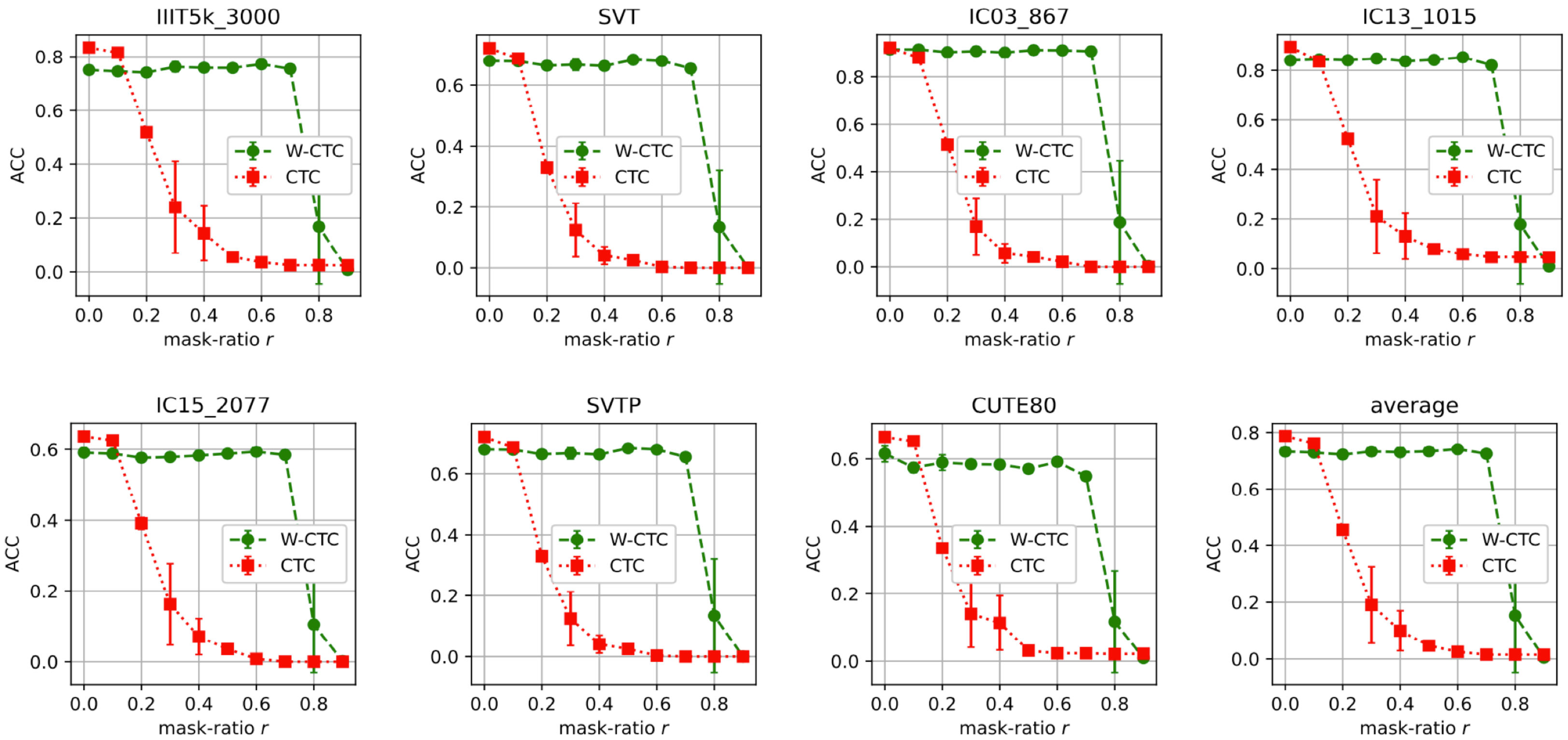
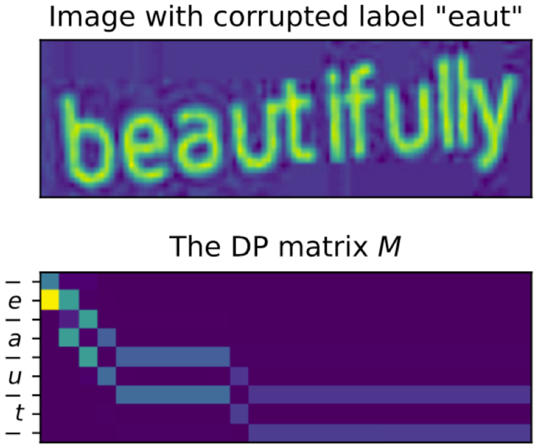
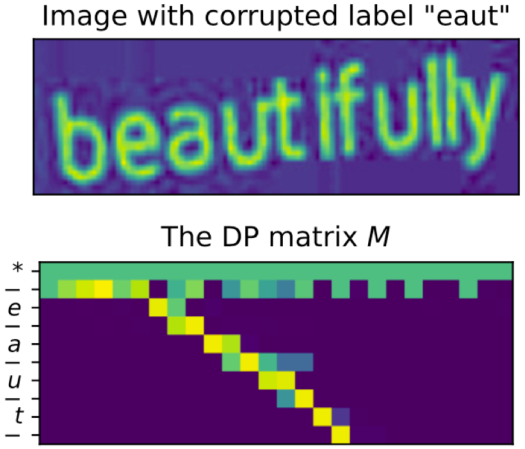


Figure 3: Test accuracy on 7 standard test sets, as a function of r (mask ratio). The last plot is the average. The proposed W-CTC has generally better accuracy than the standard CTC.



(a) Model A, standard CTC.



(c) Model C, W-CTC.

Thank You

The code can be found at:

https://github.com/TideDancer/iclr22-wctc