Improving Automatic Speech Recognition with Decoder-Centric Regularisation in Encoder-Decoder Models

Anonymous ACL submission

Abstract

This paper proposes a simple yet effective way of regularising the encoder-decoder-based automatic speech recognition (ASR) models that enhance the robustness of the model and improve the generalisation to out-of-domain scenarios. The proposed approach is dubbed as **De**coder-**C**entric **R**egularisation in **E**ncoder-**D**ecoder (DeCRED) architecture for ASR, where auxiliary classifier(s) is introduced in layers of the decoder module. Leveraging these classifiers, we propose two decoding strategies that re-estimate the next token probabilities. Using the recent E-branchformer architecture, we build strong ASR systems that obtained competitive WERs as compared to Whisper-medium and outperformed OWSM v3; while relying only on a fraction of training data and model size. On top of such a strong baseline, we show that DeCRED can further improve the results and, moreover, generalise much better to out-of-domain scenarios, where we show an absolute reduction of 2.7 and 2.9 WERs on AMI and Gigaspeech datasets, respectively. We provide extensive analysis and accompanying experiments that support the benefits of the proposed regularisation scheme.

1 Introduction

One of the key challenges in automatic speech recognition (ASR) is the ability of the models to generalise to new or unseen domains. Large-scale training on multiple domains (Narayanan et al., 2018), data augmentation, multi-task training (Hori et al., 2017), architecture-specific regularisation (Lee and Watanabe, 2021) are some of the strategies for improving the robustness of ASR systems. Some of these techniques, such as SpecAug (Park et al., 2019), joint CTC/attention training, label smoothing (Pereyra et al., 2017; Kim et al., 2018) are now a defacto and are integrated into open source toolkits (Watanabe et al., 2018; Ravanelli et al., 2021). Recent years have seen a shift towards large-scale training (Chan et al., 2021) of speech models such as Whisper from OpenAI (Radford et al., 2023). Despite its impressive recognition accuracy on many research datasets, the lack of transparency about the training data has led the scientific community to build an *open-source equivalent* of Whisper. One such effort, dubbed as OWSM (Open Whisper-style Speech Model) (Peng et al., 2023) is trained on publicly available speech datasets, using an open source toolkit ESPnet (Watanabe et al., 2018). It is important to note that these datasets come from various domains and styles, such as conversational, lectures/talks, broadcast news, telephone-speech, read-speech, etc. The size of each dataset in the training set also varies significantly. However, it is hard to evaluate the out-of-domain generalization of these models since all the standard datasets were already seen during the training.

In this paper, we ask the question *what additional, yet, simple method can further improve the robustness of ASR systems*? To answer this, we create a setup where we train a large-scale ASR model[1] on a collection of multiple datasets, carefully leaving out a few datasets to be used for out-of-domain evaluation. The ASR model is built on the most recent E-branchformer architecture (Kim et al., 2023) and trained with all the aforementioned augmentation, label-smoothing and multi-task training techniques that are known the improve the robustness of the model. On top of that, we hypothesise that regularising the ASR model during training prevents overfitting and helps generalise better in out-of-domain scenarios. Our choice of regularisation is architecture-driven, i.e., we choose to regularise the decoder module of the encoder-decoder architecture by introducing auxiliary classifier(s) in the intermediate layers.

[1] To the extent supported by the computational budget available to us.

The decoder module in a standard encoder-decoder-based ASR can be viewed as an auto-regressive internal language model (ILM) (Zeineldeen et al., 2021)[2]. Having an auxiliary classifier adds negligible computational cost during training and no additional cost during decoding (inference) if the auxiliary classifiers are ignored. We further hypothesise that these auxiliary classifiers can be exploited for rapid adaptation to a new domain that was not seen during training.

1.1 Summary and contributions

- Section 2 discusses related works and highlights how our work complements the existing body of research.
- The decoder-centric regularisation is formally introduced in Section 3, where we also describe the proposed decoding strategies that exploit the auxiliary classifiers for joint-decoding and rapid domain adaptation.
- Experiment protocol is described in Section 4, followed by experiments on large-scale multi-domain training and evaluation on out-of-domain datasets; detail of which are presented in Section 5. Compared to the baseline, we show an absolute reduction of 2.7 and 2.9 word error rates (WERs) on AMI and Gigaspeech out-of-domain datasets, respectively.
- The analysis of the internal language model (ILM) for both the baseline and the proposed DeCRED is presented in Section 6. This analysis provides complementary evidence supporting our hypothesis that regularising the decoder module indeed helps generalise in out-of-domain scenarios.
- Experiments and results from an extensive ablation study are presented in Section 7, where we identify the various factors that affect the final performance (WER) of the systems.
- Finally, our implementations[3] are built on top of open-source `transformers` library (Wolf et al., 2020), facilitating easy replication of our results. We intend to release all model checkpoints along with the corresponding test hypotheses. Our code also allows for single-line inference within the HuggingFace ecosystem.

2 Related works

The idea of auxiliary classifiers or intermediate regularizers has been explored in ASR and self-supervised learning models for speech representation. Most of these works use the auxiliary classifiers in the encoder module. More specifically, Lee and Watanabe (2021) uses intermediate CTC objectives in the encoder module for ASR, while Nozaki and Komatsu (2021) extends this by adding intermediate classifier outputs to the input of the next layer, conditioning the final layer's predictions on these intermediate outputs. Similarly, Wang et al. (2021b) employs a similar scheme for training self-supervised speech encoders. Zhang et al. (2022) regularizes both the encoder and decoder modules by passing the intermediate representations from the encoder directly to the intermediate layers in the decoder. While these works have demonstrated improvements over their respective baselines, our work complements the prior work in the following ways:

- We introduce auxiliary classifier(s) only in the decoder module of the encoder-decoder architecture, essentially regularising the auto-gressive internal language model.
- We study the effect of such a regularisation scheme in the context of out-of-domain generalisation.
- We propose to exploit the auxiliary classifiers for rapid domain adaptation.

In the case of large-scale training of end-to-end ASR models, we mainly take inspiration from prior works such as SpeechStew (Chan et al., 2021) and OWSM (Peng et al., 2023), where we simply mix multiple publicly available datasets to train our models. It is important to note that simple aggregation from multiple sources (datasets) without text normalising can cause the models to *memorise* dataset-specific annotation styles (Peng et al., 2023); which is not desired for a general purpose ASR system. This also indicates a potential inefficiency, wherein model parameters are allocated towards recognising data sources

[2] Encoder with a CTC objective can also learn an internal language model, though non-autoregressive.

[3] [to ensure author anonymity, the link to the resource will be added after the review process]

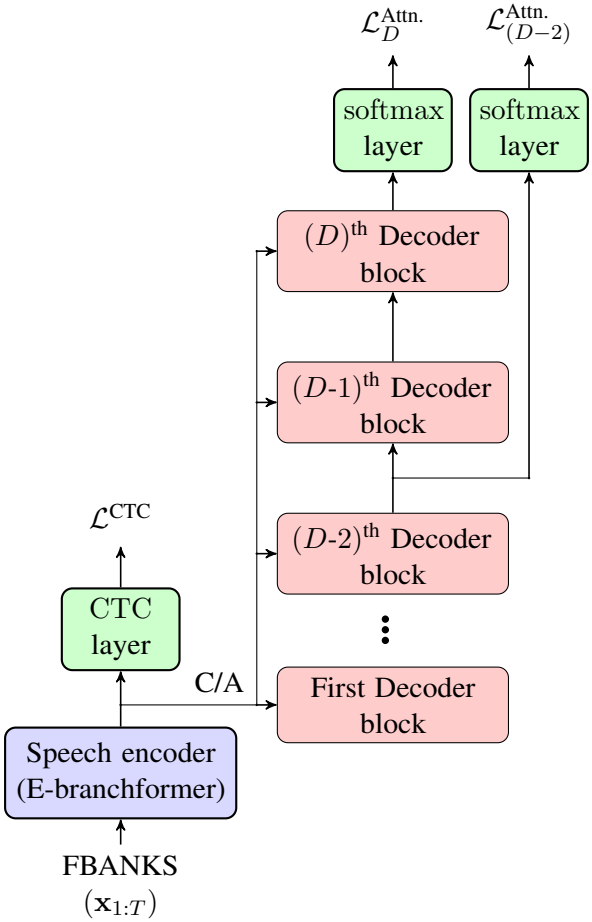

Figure 1: Architecture of the proposed DeCRED. In addition to the standard encoder-decoder framework for ASR ($\mathcal{L}_{D}^{\text{Attn}}$), with the auxiliary CTC objective ($\mathcal{L}^{\text{CTC}}$), DeCRED uses – possibly multiple – auxiliary classifiers ($\mathcal{L}_{d}^{\text{Attn}}$) attached to the decoder. In the illustration, we show one auxiliary classifier attached to $(D\text{-}2)$-th decoder block. The embedding and positional encoding layers are not depicted for brevity.

rather than solving the intended task(s). As it is inevitable, we investigate and quantify the effect of text normalisation on the model's recognition performance.

3 Decoder-centric regularization

Formally, our approach extends the training objective of encoder-decoder ASR by adding auxiliary cross-entropy loss functions. We explore two additional decoding methods that exploit these auxiliary classifiers.

3.1 Training objective

We build upon the hybrid CTC-attention-based training scheme proposed by Hori et al. (2017). Our objective function $\mathcal{L}$ is defined as:

$$\mathcal{L} = \alpha\, \mathcal{L}^{\text{CTC}} + (1-\alpha)\, \mathcal{L}^{\text{DeCRED}}, \qquad (1)$$

where $\mathcal{L}_{\text{CTC}}$ represents the standard CTC loss (Graves et al., 2006), α is a hyper-parameter, and $\mathcal{L}_{\text{DeCRED}}$ is defined as:

$$\mathcal{L}^{\text{DeCRED}} = \sum_{d=1}^{D} \beta_d \mathcal{L}_{d}^{\text{Attn}}, \qquad (2)$$

where D represents the number of layers in the decoder, $\mathcal{L}_{d}^{\text{Attn}}$ is the cross-entropy loss given a classifier layer (linear projection, followed by softmax function) attached to the d-th layer of the decoder, and β_d is the weighting factor of d-th layer. We impose constraints such that $\sum_{d=1}^{D} \beta_d = 1$ and $\beta_d \geq 0$. In practise $[\beta_1 \ldots \beta_D]$ is a sparse vector. This definition allows us to explicitly regularise the decoder (internal language model) and force earlier layers to learn discriminative features suitable for the task. Figure 1 illustrates the proposed architecture, where an auxiliary classifier is attached to the output of $(D\text{-}2)$-th decoder block.

3.2 Decoding

The decoding follows a typical auto-regressive scheme observed in encoder-decoder ASR systems, where the posterior probability of an output token is obtained by conditioning on previously decoded tokens (partial hypothesis) and the input features.

Formally, let $\mathbf{x}_{1:T}$ be a sequence of input speech (filterbank) features and let $y_{1:N}$ be a sequence of output tokens. Following the joint CTC/attention decoding (Hori et al., 2017), the posterior probability of an output token y_n is evaluated as

$$\begin{aligned} &\log p(y_n \mid y_{1:n-1}, \mathbf{x}_{1:T}) \approx \\ &\lambda \log p_{\text{CTC}}(y_n \mid y_{1:n-1}, \mathbf{x}_{1:T}) + \\ &(1-\lambda) \log p_{\text{DeCRED}}(y_n \mid y_{1:n-1}, \mathbf{x}_{1:T}), \end{aligned} \qquad (3)$$

where λ is a hyper-parameter. Now, let $\mathbf{h}_{d,n} \in \mathbb{R}^{1\times d_{\text{model}}}$ denote the hidden representation corresponding to the n-th output token obtained from the d-th layer of the decoder, and $\mathbf{W}_d \in \mathbb{R}^{d_{\text{model}}\times V}$ represent linear projection from hidden dimension d_{model} to vocabulary size V. We obtain the following decoding methods by varying the definition of p_{DeCRED}:

1. Vanilla joint CTC/attention decoding relying on representations *only* from the last layer:

$$\begin{aligned} p_{\text{DeCRED}}(y_n \mid y_{1:n-1}, \mathbf{x}_{1:T}) = \\ \text{softmax}(\mathbf{h}_{D,n}\mathbf{W}_D) \end{aligned} \qquad (4)$$

2. Sum of logits weighted by per-layer learnable scalar β_d:

$$p_{\text{DeCRED}}(\cdot) = \text{softmax}(\sum_{d=1}^{D} \beta_d \mathbf{h}_{d,n} \mathbf{W}_d) \quad (5)$$

3. Sum of logits weighted by per-layer learnable vector $\mathbf{v}_d \in \mathbb{R}^{1\times V}$, where $\odot$ is elementwise product:

$$p_{\text{DeCRED}}(\cdot) = \text{softmax}\left(\sum_{d=1}^{D} \mathbf{v}_d \odot (\mathbf{h}_{d,n} \mathbf{W}_d)\right) \quad (6)$$

Note that to obtain optimal results with methods (5) and (6), an additional held-out set is required for learning the parameters β_d, $\mathbf{v}_d$.

The above schemes can be easily integrated into any of the decoding search algorithms, such as greedy and beam-search.

4 Experimental setup

The experiments are organised into two parts. The first part (Sec. 5) compares baseline ED with the proposed DeCRED on multi-domain English datasets. The selection of datasets is inspired by those used for evaluation in OWSM. This relatively larger corpus allows us to fully exploit the proposed decoding alternatives (5) and (6), assess the effectiveness of Internal Language Model (ILM) regularisations, and compare our models' in- and out-of-domain performance with large-scale trained speech models.

The second part (Sec. 7) focuses on single in-domain datasets, studying the effects of the position (d), weight (β_d) of the auxiliary classifiers, their influence in decoding and the impact of single-domain text normalisations.

All our experiments are built on the open-source `transformers` library, accompanied by baseline models built using the ESPnet toolkit.

4.1 Baseline Encoder-Decoder (ED) model

Throughout this paper, we use the quadruplet (E, D, d_{model}, V), where E represents the number of layers in the encoder, D refers to the decoder layers, d_{model} is the hidden dimension, and V is the vocabulary size. The rest of the configuration remains fixed unless explicitly stated otherwise.

Our baseline ED small $(12, 6, 256, 5000)$ and ED base $(16, 8, 512, 5000)$ models contain 39M and 172M parameters, respectively. They receive 80-dimensional filter-bank features as input and employ an input module consisting of two Conv2d layers with 256 output channels, followed by a linear projection. This is followed by an E-layer E-Branchformer (Kim et al., 2023) encoder with relative positional embeddings (Dai et al., 2019), Macaron-like feedforward modules (Gulati et al., 2020), $d_{\text{ff}} = 4d_{\text{model}}$, four attention heads, and a dropout probability of 0.1.

In line with the E-Branchformer architecture, we incorporate a merge block followed by depth-wise convolution with a kernel size of 31. The encoder is followed by a D-layer Transformer decoder with sinusoidal positional embeddings, maintaining the same number of attention heads, d_{model}, and dropout. The decoder has fixed $d_{\text{ff}} = 2048$. We use a subword tokenizer based on the unigram algorithm.

We use the same quadruplet as in ED to define the Decoder-Centric Regularized Encoder-Decoder (DeCRED) architecture. The only difference is the number of attached classifiers and their corresponding weights. The weights of the additional classifiers are not tied to the one attached to the D-th layer. For the baseline DeCRED models (small and base), a single additional classifier with $\beta_{D-2} = 0.4$ is attached, adding only $d_{\text{model}} \times V$ new parameters.

4.2 Training details

Our models are trained on Nvidia A100 GPUs with bf16 precision using the AdamW optimiser (Loshchilov and Hutter, 2019) for 100 epochs with early stopping patience of 10, learning rate of 2×10^{-3}, weight decay of 1×10^{-6}, linear decay scheduler and 40k warm-up steps. We use a label smoothing (Pereyra et al., 2017; Watanabe et al., 2018) weight of 0.1 as an additional means of regularisation. To speed up the training, samples longer than 20 seconds are discarded from the training set.

Unlike ESPnet, where some augmentations are applied offline, we implement all augmentations online and allow for postponing some of them until later in the training, resulting in a more stable training process. For instance, while ESPnet adopts a training regime consisting of 50 epochs and three copies of the input data with speed perturbation factors 0.9, 1.0, and 1.1, we train our model for 150 epochs on the original data with speed perturbation factors $\{0.9, 1.0, 1.1\}$ randomly

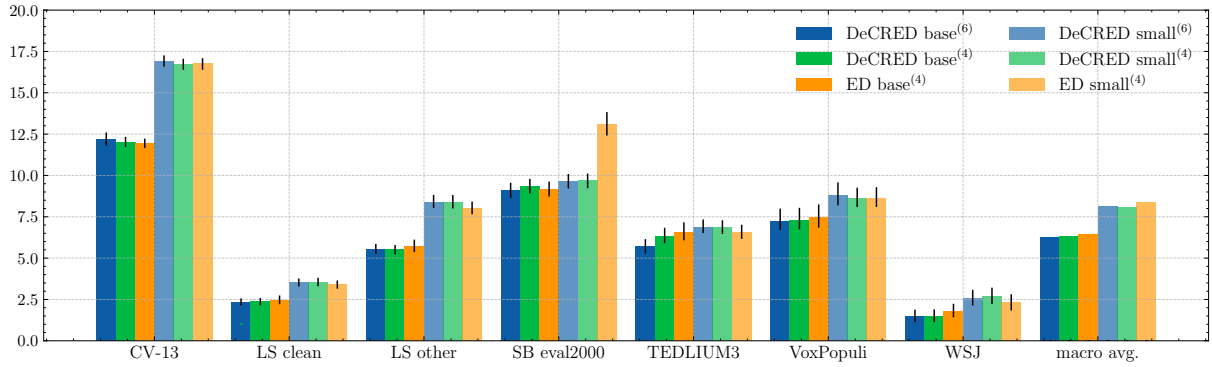

Figure 2: The impact of employment of the proposed training strategy, along with the enhanced decoding (6) on small (12, 6, 256, 5000) and base (16, 8, 512, 5000) models using greedy $\lambda = 0.3$ decoding. DeCRED-base$^{(6)}$ indicates the model with the proposed decoding technique (6), and the mixing parameters $\mathbf{v}$ tuned on development split. To compute confidence intervals, we employed bootstrapping with $\alpha = 0.05$ and $B = 1000$.

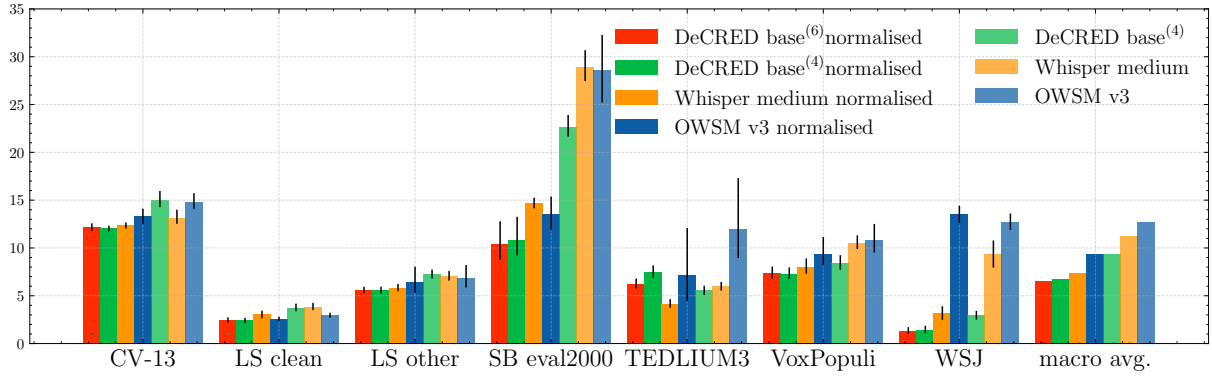

Figure 3: Comparison of the proposed model against publicly available models on original and normalised transcripts using greedy decoding (4) with $\lambda = 0$, as Whisper lacks a CTC head. Additional gains can be observed when using $\lambda = 0.3$ for both DeCRED and OWSM v3 models.

sampled on the fly. After 5k update steps, we apply SpecAug (Park et al., 2019) with two frequency-masks of maximum size 27 and five time-masks with maximum coverage of masked input of 5 %. For all experiments, we select the best-performing checkpoint based on the development WER. Additionally, we introduce a mechanism to mask special tokens, along with unfinished words[4], during error backpropagation. This strategy aimed to prevent the model from being penalised for unclear inputs.

5 ED vs DeCRED in multi-domain scenario

To build robust ASR systems that are on par with state-of-the-art, we chose a mixture of multi-domain datasets that allows for bigger training, development and test sets. The multi-domain dataset is comprised of Fisher (SWITCHBOARD) (Godfrey et al., 1992), WSJ (Paul and Baker, 1992), Common Voice en 13 (Ardila et al., 2020), LibriSpeech (Panayotov et al., 2015), VoxPopuli (Wang et al., 2021a), and TED-LIUM 3 (Hernandez et al., 2018), totalling 6k hours of training data. To study the generalisation capabilities of our models, we also evaluate our ED and DeCRED models on three unseen datasets (AMI (Carletta, 2007), FLEURS (Conneau et al., 2023) and Gigaspeech (Chen et al., 2021)).

5.1 Normalisation of multi-domain data

These datasets have different annotation styles, making learning harder and introducing undesired behaviour in the models, such as memorising the dataset-specific annotations (Peng et al., 2023). We employed a practical approach, where we used the text normalisation scheme from Whisper[5] to standardise the transcripts across all

[4] e.g. transcript "[hesitation] to re- to re- renew" is transformed into "[MASK] to [MASK] to [MASK] renew"

[5] https://github.com/huggingface/transformers/blob/main/src/transformers/models/whisper/

Table 1: Comparison of ED and DeCRED models on out-of-domain test sets. WERs are obtained using greedy decoding with $\lambda = 0$. † denotes models where $\mathbf{v}^*$ was tuned on each of the datasets separately.

Model	FLEURS	AMI-ihm	Gigaspeech
ED base$^{(4)}$	6.4	24.8	19.8
DeCRED base$^{(4)}$	6.7	22.1	16.9
DeCRED base$^{(6)}$	6.9	21.9	17.0
DeCRED base$^{(6)\dagger}$	6.8	21.4	16.4
OWSM v3	8.6	35.8	34.1
Whisper medium	**5.5**	**16.6**	**14.9**

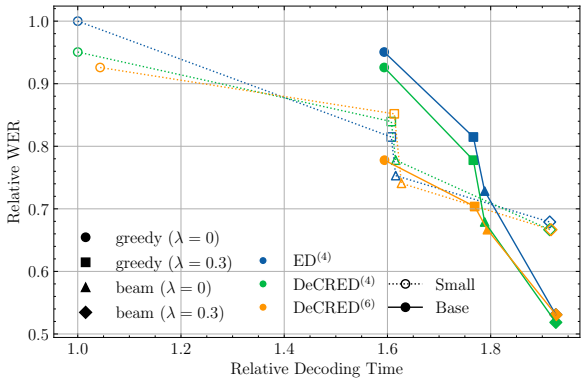

Figure 4: The impact of model size and decoding approach on the average time needed to transcribe an utterance (TEDLIUM3) and WER (macro average across datasets).

the datasets. We believe this approach allows the model to focus mainly on the recognition task. For practical applications, true casing and punctuation can be restored using a lightweight inverse text normalisation model. In addition to the Whisper text normaliser, we retained the text within parenthesis. Due to inconsistencies across datasets, we removed special tokens such as [breath], [vocalised noise], [pause], [sneeze].

Nevertheless, to enable a fair comparison with prior works, we also report results by training and evaluating the original transcripts. This allows us to quantify the effect of text normalisation on the WER.

5.2 Comparison with a fair baseline

Figure 2 compares the WER of baseline ED and the proposed DeCRED across all the in-domain datasets. Specifically, we compare both the small (12, 6, 256, 5000) and base (16, 8, 512, 5000) variants of both ED and DeCRED.

To learn the mixing parameters β_d and $\mathbf{v}$ for the respective decoding methods (Section 3.2), we followed the Platt scaling approach (Guo et al., 2017; Lee and Chang, 2021), splitting the original set development utterances into new training and development sets with a 70:30 ratio. Except for the mixing parameters, the rest of the model remains frozen. This training, or fine-tuning, is very lightweight.

We use the equation number in the superscript of the model to denote the decoding objective, i.e. DeCRED$^{(4)}$ indicates the vanilla decoding method defined by (4), DeCRED$^{(5)}$ indicates mixing the logits by learnable scalars, and DeCRED$^{(6)}$ indicates mixing the logits by learnable vectors.

The Figure also shows the macro WER computed across all the datasets. We computed a statistical significance test using the bootstrapping method, which showed that results from DeCRED-base$^{(4)}$ and DeCRED-base$^{(6)}$ are statistically significant than baseline ED-base$^{(4)}$ with p values of 0.35 and 0.3 respectively.

5.3 Comparison with Whisper and OWSM

Figure 3 provides a reference comparison between our best models, DeCRED-base$^{(4)}$ (172M parameters) and DeCRED-base$^{(6)}$, and large-scale multilingual models — Whisper-medium (Radford et al., 2023) (700M parameters) and OWSM v3 (Peng et al., 2023) (889M parameters).

It is important to note that this figure serves only as a reference and does not represent a fully fair comparison, as the models differ significantly in terms of scale and design.

Although most of our models were trained on normalised transcriptions, we also trained the DeCRED-base (16, 8, 512) model on original transcriptions to highlight the effect of text normalisation. To ensure consistency, we applied the same text normalisation used in our training pipeline to the Whisper and OWSM outputs during evaluation in the normalised setup.

5.4 Performance on out of domain

In Table 1, we compare the performance of ED and DeCRED models on out-of-domain datasets that were not seen during training. We take this as an opportunity to evaluate the effect of rapid tuning of the mixing weights $\mathbf{v}^*$ on the corresponding domain. For this, we utilised FLEURS train split and development splits of AMI and Gigaspeech, respectively, following the training protocol

`english_normalizer.py`

Table 2: Zero-Attention Internal Language Model (ILM) BPE-level perplexity estimation of ED and DeCRED models on in- and out-of-domain test sets.

Model	CV-13	LS clean	LS other	SB eval2000	TEDLIUM3	VoxPopuli	WSJ	FLEURS	AMI-ihm	Gigaspeech
DeCRED	141.0	129.1	140.4	104.1	89.0	101.4	126.0	111.5	136.6	66.3
ED	232.4	206.1	199.9	220.3	134.6	142.7	177.3	159.7	308.3	84.0

Table 3: Macro average of the WERs based on the selected decoding. All auxiliary classifier weights β_d are set to 0 by default. Parameters with an asterisk (e.g., β_d^*, $\mathbf{v}^*$) indicate tuning on a portion of the development split, i.e. training mixing weights for layers $D-2$ and D for DeCRED-base.

DeCRED-base decoding strategy	greedy $\lambda=0$	greedy $\lambda=0.3$	beam – width 10 $\lambda=0$	beam – width 10 $\lambda=0.3$
$\beta_6 = 1^{(5)}$	6.8	6.7	6.5	6.4
$\beta_8 = 1^{(4)}$	6.7	6.4	**6.4**	**6.0**
$\beta_{6,8} = (0.40, 0.60)^{(5)}$	6.7	6.6	**6.4**	6.3
$\beta_{6,8}^* = (0.50, 0.47)^{(5)}$	6.7	6.6	6.5	6.3
$\mathbf{v}^{*(6)}$	**6.5**	**6.3**	6.9	**6.0**

described in Section 3.2. In Table 1, these models are denoted as DeCRED base$^{(6)\dagger}$. In all cases, adapting $\mathbf{v}^*$ leads to a decrease in WER, and with the exception of the FLEURS dataset, this decrease is considerable, confirming that the mixing weight does provide a rapid adaptation capability.

Overall, all our models outperform the much larger OWSMv3, which has also been trained on the corresponding training data, showing that our models do generalize to unseen domains well. With the exception of the FLEURS dataset, where the difference in WER is the smallest anyway, DeCRED models outperform the ED baseline significantly, suggesting that the decoder-centric regularisation enhances the generalisation ability of the model.

5.5 Comparison across different decoding methods

Table 3 presents a comparison of different decoding methods in terms of macro WER over the in-domain datasets. We observed that integrating intermediate representations with per-layer learnable weights (5) led to minor improvement only in the greedy decoding scenario without hybrid CTC decoding. Notable improvements were observed with the incorporation of per-token-specific mixing (6), except for beam decoding on the SB eval2000 dataset, where we observed a degradation of $5.3\,\%$ in WER, primarily attributed to insertions. Interestingly, we did not observe the same behaviour with the small model. For completeness, macro WERs are also provided for early-exiting ($\beta_6 = 1$, i.e., decoding directly from the 6-th layer, while the model has 8 layers), where only minor degradations were observed.

5.6 Trade-off between performance and decoding time

Figure 4 presents the relative WER reductions of our multi-domain models on TEDLIUM3 in relation to the relative slowdown caused by additional decoding overhead. The slowdown factor is measured relatively to our fastest model ED small$^{(4)}$. It is calculated as an average time over the TEDLIUM3 test set required to emit 20 tokens on an A100 GPU with maximum VRAM memory consumption[6]. We fixed a number of decoding steps to normalise different hypothesis lengths across models.

In this setup, there is no speed difference between ED$^{(4)}$ and DeCRED$^{(4)}$. However, as shown in Figure 4, regularised models significantly reduce the WER. When using DeCRED$^{(6)}$, the only overhead is computing softmax $\left(\sum_{d=1}^{D} \mathbf{v}_d \odot (\mathbf{h}_d \mathbf{W}_d)\right)$, where $\mathbf{h}_D \mathbf{W}_D$ is already computed. It is worth noting that when using greedy decoding, DeCRED small performs similarly to ED base, being much smaller, thus consuming less computation resources and speeding up decoding significantly.

6 Analysis of internal language model

In the attention-based Encoder-Decoder (ED) ASR framework, the decoder functions as an autoregressive internal language model. In this paper, we directly regularise this component of the network by incorporating auxiliary classifiers. In the previous Section 5, we demonstrated consistent improvements of DeCRED vs ED achieved through

[6]For example, with greedy decoding and ED small, we can fit 240 samples in a batch. In contrast, with ED base and joint CTC/attention decoding with a beam size of 10, we are only able to fit 20 samples.

Table 4: Comparison of our implementation of ED and proposed DeCRED with InterCTC and ESPnet's ED baselines on the TEDLIUM3 test split.

Model	Size [M]	WER [%] greedy	WER [%] beam – width 40
ESPnet ED$^{(4)}$	35.01	8.7	8.1
Our ED$^{(4)}$	35.04	7.6	7.2
DeCRED$^{(4)}$	35.20	**7.0**	**6.8**
InterCTC$^{(4)}_{\lfloor L/2 \rfloor}$	35.20	7.5	7.1

such regularisation. Building on the work of Zeineldeen et al. (2021), we employ Zero-Attention Internal Language Model (ILM) subword-level perplexity estimation to analyse the impact of our proposed regularisation scheme across in-domain and out-of-domain datasets.

Table 2 showcases a consistent reduction in perplexity across all analysed datasets of DeCRED base versus ED base, strongly indicating that the ILM generalises much better across multiple domains, which further supports our hypothesis.

7 Ablations on in-domain dataset

To further analyse and understand the proposed regularisation scheme, we select a relatively small dataset, TEDLIUM3 (Hernandez et al., 2018), which allows for faster experiment turnout. The dataset comprises 452 hours of transcribed TED talks, with a test set containing 1155 utterances, roughly translating to 28k words. The size of this dataset enables us to train an ED (12, 6, 256, 500) baseline 35M model to full convergence in approximately 70 A100 hours.

Since we build on top of `transformers` library, to ensure a fair comparison, we adopt hyperparameters and a training setup as close as possible to the ESPnet baseline recipe[7]. For evaluating the models on TEDLIUM3, unless explicitly specified, we follow the ESPnet recipe utilising joint CTC/attention decoding (4) with a beam size of 40 and CTC decoding weight $\lambda = 0.3$. Table 4 compares our best-performing DeCRED and ED baseline models with the baseline model from ESPnet and InterCTC baseline (Lee and Watanabe, 2021). DeCRED consistently outperforms both implementations of ED. The difference is better pronounced in greedy decoding, suggesting the effectiveness of DeCRED in decoding tasks where computational resources are limited. Appendix A provides more details about the hyperparameter search.

7.1 Effect of text normalisation

To further understand the effect of text normalisation, we trained standalone models ED-small (12, 6, 256) on the TEDLIUM3 and Voxpopuli datasets with and without normalised transcripts. Notably with normalisation, we observed an improvement from 9.8 % to 9.0 % WER on VoxPopuli and from 7.2 % to 6.7 % on TEDLIUM3. The normalisation process effectively resolved contraction errors and also led to fewer errors in the most frequent confusion pairs (e.g., "the" vs "a", "in" vs "on", "in" vs "and"). By normalising, we reduced the number of words from 44.3k to 44.1k for Voxpopuli and increased this number from 27.5k to 28.2k for TEDLIUM3, which also influenced the WER.

8 Conclusion

We introduced the DeCRED regularization scheme, which effectively integrates auxiliary classifiers within the decoder of an encoder-decoder-based architecture. We further proposed decoding methods that exploit these auxiliary classifiers, which led to a significant decrease in the word error rates. We observed that DeCRED consistently improves the results when employing a simple greedy decoding scheme compared to the baseline models. Our experiments on multi-domain datasets show that DeCRED is scalable, performs competitively to much larger Whisper medium, and outperforms OWSM v3. Finally, we show that DeCRED enhances the generalisation to out-of-domain datasets, where we observed a reduction of 2.7 % and 2.9 % WER, on AMI and Gigaspeech, respectively. Using a lightweight rapid domain adaptation scheme enabled by DeCRED, the out-of-domain WERs were further reduced by 0.7 % and 0.5 % absolute on the respective datasets. In future, we intend to study DeCRED in multilingual and multi-task scenarios.

9 Limitations

We identify a few of limitations in our work. Firstly, due to our computational budget, we were only able to scale our setup to 6k hours of training data and 172M model parameters. Secondly, our models were trained on English data only, which

[7] https://github.com/espnet/espnet/tree/master/egs2/tedlium3/asr1

makes the comparison with multilingual models tricky, as these models had to invest a part of their capacity into modelling other languages as well. Yet, due to the first point, our models are exposed to one (OSWM) or even two (Whisper) orders of magnitude less English data, therefore we believe the comparison is not unfair. Also, our models use considerably smaller vocabulary; however, while this might limit model performance on domain-specific words present, for example, in the FLEURS dataset, we do not observe performance degradation there.

Next, some of the improvements from introducing DeCRED diminish when employing beam-search decoding with a wider beam, which, however, comes at a computational cost at inference time. Finally, while the proposed decoder-centric regularization is independent of the backbone architecture, we have only analysed our approach using an E-branchformer speech encoder.

References

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Jean Carletta. 2007. Unleashing the killer corpus: experiences in creating the multi-everything ami meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

William Chan, Daniel Park, Chris Lee, Yu Zhang, Quoc Le, and Mohammad Norouzi. 2021. SpeechStew: Simply Mix All Available Speech Recognition Data to Train One Large Neural Network.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. 2021. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. In *Interspeech 2021*, pages 3670–3674.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. FLEURS: FEW-Shot Learning Evaluation of Universal Representations of Speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

J.J. Godfrey, E.C. Holliman, and J. McDaniel. 1992. SWITCHBOARD: telephone speech corpus for research and development. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520 vol.1.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 369–376, New York, NY, USA. Association for Computing Machinery.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented Transformer for Speech Recognition. *Interspeech 2020*, pages 5036–5040. Conference Name: Interspeech 2020 Publisher: ISCA.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1321–1330. JMLR.org.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. Tedlium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In Alexey Karpov, Oliver Jokisch, and Rodmonga Potapova, editors, *Speech and Computer*, pages 198–208. Springer International Publishing, Cham.

Takaaki Hori, Shinji Watanabe, and John Hershey. 2017. Joint CTC/attention decoding for end-to-end speech recognition. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 518–529, Vancouver, Canada. Association for Computational Linguistics.

Kwangyoun Kim, Felix Wu, Yifan Peng, Jing Pan, Prashant Sridhar, Kyu J. Han, and Shinji Watanabe. 2023. E-Branchformer: Branchformer with Enhanced Merging for Speech Recognition. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 84–91.

Suyoun Kim, Michael Seltzer, Jinyu Li, and Rui Zhao. 2018. Improved training for online end-to-end speech recognition systems. In *Interspeech 2018*, pages 2913–2917.

Jaesong Lee and Shinji Watanabe. 2021. Intermediate Loss Regularization for CTC-Based Speech Recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6224–6228.

Mun-Hak Lee and Joon-Hyuk Chang. 2021. Deep Neural Network Calibration for E2E Speech Recognition System. In *Interspeech 2021*, pages 4064–4068.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Arun Narayanan, Ananya Misra, Khe Chai Sim, Golan Pundak, Anshuman Tripathi, Mohamed Elfeky, Parisa Haghani, Trevor Strohman, and Michiel Bacchiani. 2018. Toward Domain-Invariant Speech Recognition via Large Scale Training. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 441–447.

Jumon Nozaki and Tatsuya Komatsu. 2021. Relaxing the conditional independence assumption of ctc-based asr by conditioning on intermediate predictions. In *Interspeech 2021*, pages 3735–3739.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. ISSN: 2379-190X.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Interspeech 2019*, pages 2613–2617. ISCA.

Douglas B. Paul and Janet M. Baker. 1992. The Design for the Wall Street Journal-based CSR Corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Yifan Peng, Jinchuan Tian, Brian Yan, Dan Berrebbi, Xuankai Chang, Xinjian Li, Jiatong Shi, Siddhant Arora, William Chen, Roshan Sharma, Wangyou Zhang, Yui Sudo, Muhammad Shakeel, Jee-Weon Jung, Soumi Maiti, and Shinji Watanabe. 2023. Reproducing whisper-style training using an open-source toolkit and publicly available data. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8.

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. In *5th International Conference on Learning Representations, ICLR*. OpenReview.net.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. 2023. Robust Speech Recognition via Large-Scale Weak Supervision. In *Proceedings of the 40th International Conference on Machine Learning*, pages 28492–28518. PMLR. ISSN: 2640-3498.

Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. 2021. SpeechBrain: A general-purpose speech toolkit. *Preprint*, arXiv:2106.04624. ArXiv:2106.04624.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021a. VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 993–1003, Online. Association for Computational Linguistics.

Chengyi Wang, Yu Wu, Sanyuan Chen, Shujie Liu, Jinyu Li, Yao Qian, and Zhenglu Yang. 2021b. Self-supervised learning for speech recognition with intermediate layer supervision. *Preprint*, arXiv:2112.08778.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. Espnet: End-to-end speech processing toolkit. In *Interspeech 2018*, pages 2207–2211.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Mohammad Zeineldeen, Aleksandr Glushko, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2021. Investigating methods to improve language model integration for attention-based encoder-decoder asr models. In *Interspeech 2021*, pages 2856–2860.

Jicheng Zhang, Yizhou Peng, Haihua Xu, Yi He, Eng Siong Chng, and Hao Huang. 2022. Intermediate-layer output regularization for attention-based speech recognition with shared decoder. *Preprint*, arXiv:2207.04177.

A Position and weight of auxiliary classifiers

Even with a model with as few as $D = 6$ decoder layers, the definition of the DeCRED objective (2) leaves us with a vast configuration space. We explored this space starting with the configurations with a single auxiliary classifier, changing its position and adjusting its weight in increments of 0.1. The additional parameters introduced ($\mathbf{W}_d \in \mathbb{R}^{256\times500}$) by a single auxiliary classifier do not significantly increase the model size.

The results are summarised in Table 5. Compared to our baseline ED model with a WER of 7.2 %, we observe improvements with the additional classifier placed closer to the final layer.

Further experiments with multiple auxiliary classifiers ($\{\beta_3 = 0.2, \beta_4 = 0.3, \beta_6 = 0.5\}$ and $\{\beta_3 = 0.2, \beta_5 = 0.3, \beta_6 = 0.5\}$), did not yield significant improvements, discouraging experiments with more auxiliary classifiers. We avoided exploring very low weights (β_d) in the early layers as gradual adjustments did not yield noticeable improvements. Given the computational resources required for each experiment run, we chose to run the two most promising configurations five times to determine the optimal one. Choosing between the two most promising configurations from Table 5, i. e., $\beta_3 = 0.5$ and $\beta_4 = 0.4$, we opted for the latter for all subsequent experiments. We believe other configurations (indicated with grey colour in the lower triangle of Table 5) could yield similar results.

Table 5: Effect of the position (d) and weight (β_d) of the auxiliary classifier in DeCRED on WERs of TEDLIUM3 test set. Grey cells indicate configurations deemed reasonable for exploration. Standard deviations (σ) and best WER for the chosen configurations are displayed. For reference, the baseline ED model has a WER of 7.2 %.

Weight	Position				
	1	2	3	4	5
0.1				7.5	6.8
0.2				7.0	7.2
0.3			7.0	7.0	7.0
0.4	7.8	7.5	7.1	**6.8** [$\sigma = 0.15$]	6.9
0.5	7.2	7.1	**6.7** [$\sigma = 0.26$]	7.1	6.9