Mini-Project

An Informal Review of Relevant Literature

Robert Flynn

wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations (Baevski et al., 2020)

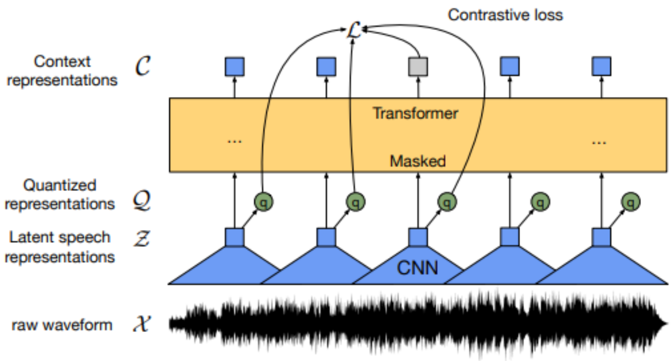

Figure 1: Diagram detailing the contrastive self-supervised pre-training for Wav2vec 2.0

- This model uses a self-supervised learning framework to build acoustic representations from unlabeled speech
- The models architecture consists of a feature encoder that uses a convolutional architecture, and a transformer architecture which intakes the features and outputs contextual speech representations
- For self-supervised pre-training the model employs a contrastive learning approach. A subset of the latent speech features representations are masked, and the model is trained with the objective of identifying the correct latent quantized representation when marginalizing over a set of "distractors" sampled from the sequence
- Figure 1 depicts the pre-training task. This approach is inspired by the masked language model pre-training used by the BERT model (Devlin et al., 2018)
- Unlabeled speech data is often readily available. This work shows that once good acoustic representations have been learnt, the model can be fine-tuned to transcribe text with a small amount of labeled data. More labeled data is always better, however this approach shows reasonable results can be obtained when fine-tuning on as little of 10 minutes of labelled audio: "when using only 10 minutes of labeled data, our approach achieves word error rate (WER) 4.8/8.2 on the clean/other test sets of Librispeech"
- For fine-tuning, a linear-projection layer is initialized on top of the (tranformer) context layer. This layer is trained to predict a set of classes, which consists of the vocabulary of the target text, for librespeech they use 29 tokens, to denote characters and word boundaries.
- For fine-tuning the model uses Connectionist Temporal Classification (CTC) (Graves et al., 2006) as its loss function. An intuitive explanation of this process is provided by Hannun (2017).

- This is an encoder only model with the linear projection layer predicting characters based on the acoustic representations at each time-step; consequently, the probabilities of the outputted characters are conditionally independent. Language models are often used with such architectures to find a more probable character sequence given the outputted logits.

ROBUST WAV2VEC 2.0: ANALYZING DOMAIN SHIFT IN SELF-SUPERVISED PRE-TRAINING (Hsu et al., 2021)

- The Wav2vec2 paper showed that once good acoustic representations have been learnt, models can be fine-tuned to transcribe audio effectively with a much smaller amount of labelled data than if no pre-training was performed.
- Pre-training for Wav2vec2 took place on audio in the same domain as the audio for fine-tuning, and evaluation. Often there will be situations where there is not enough transcribed audio from the desired target domain to fine-tune a model such as Wav2vec2.
- This paper examines whether pre-training on in-domain data then fine-tuning on Out-Of-Domain (OOD) data is beneficial for producing an ASR system that is effective in the target domain

"In the supervised learning paradigm, practitioners who would like to build a system for a new domain, can either train on existing OOD labeled data or build a corpus of labeled data in the new domain. With pre-training, we have a third option: collect unlabeled data in the new domain and fine-tune on existing labeled OOD data. This has the clear advantage of unlabeled in-domain being often much easier to obtain than transcribed in-domain data."

X	*TED-LIUM (TD) dev WER*					
	FT on TD-10h		FT on LS-10h		FT on SB-10h	
	PT on X	X+TD	PT on X	X+TD	PT on X	X+TD
None	diverge	9.93	diverge	10.99	diverge	11.32
SF	12.12	9.60	14.82	11.08	99.63	11.04
LS	9.81	8.59	12.92	8.91	13.08	10.39
SF+LS	9.13	8.91	10.61	9.67	12.25	10.75

X	*LibriSpeech (LS) dev-other WER*					
	FT on TD-10h		FT on LS-10h		FT on SB-10h	
	PT on X	X+LS	PT on X	X+LS	PT on X	X+LS
None	diverge	14.60	diverge	10.53	diverge	17.92
SF	28.91	14.30	20.36	10.44	94.38	15.53
TD	23.44	12.81	15.36	9.71	27.50	15.46
SF+TD	20.50	13.58	14.42	10.39	21.99	13.89

X	*Switchboard (SB) RT03 WER*					
	FT on TD-10h		FT on LS-10h		FT on SB-10h	
	PT on X	X+SF	PT on X	X+SF	PT on X	X+SF
None	diverge	18.90	diverge	19.30	diverge	10.80
TD	35.70	16.20	34.60	17.40	18.70	11.00
LS	33.60	17.80	36.50	16.10	18.20	11.00
TD+LS	29.70	17.40	28.90	16.90	15.60	10.80

Figure 2: Validation WER on TD, LS, and SB of models pre-trained (PT) on various subsets of TD, LS, SB, and fine-tuned (FT) on TD-10h, LS-10h, or SB-10h.

- The table in figure 2 presents the key findings of this work. The main takeaway from these results is that pre-training on data for the target domain can lead to sizeable improvements in WER, despite only fine-tuning on OOD data.
- Further results from this work show that pre-training on multiple domains improves the models ability to perform on previously unseen domains at evaluation time.

What are the implications of this work with regard to the mini-project?

It is likely that we will be able to obtain a relatively large amount of unlabelled data for our target domain of oral history. While it may be possible to acquire some data that shares similar characteristics to our target domain, it may not be an optimal amount to train/fine-tune a large ASR model on its own. Work from this paper suggests that pre-training an acoustic model with data that includes large amounts of audio from our target domain, then fine-tuning on a combination of: any near-target domain transcribed audio we can acquire, and on other datasets commonly used in ASR (librespeech e.t.c.), may yield good results.

Our current baseline model uses a Robust Wav2vec2 model that has been finetuned on librespeech[1]. A version of Robust Wav2vec2 that has not been finetuned is also available on the hugging face library[2], this model has been pre-trained in an unsupervised fashion on a variety of commonly used used corpora (libri light, Common voice, Switchboard and Fisher). We could potentially use this model as the starting point for further pre-training on Oral History unlabeled data. Note that contrastive predictive coding (van den Oord et al., 2019), that is used for wav2vec2's pre-training task, can be very computationally expensive due to its need for large batch sizes (Baevski et al., 2020). Additionally, the model would require fully fine-tuning after pre-training.

Effective Sentence Scoring Method Using BERT for Speech Recognition (Shin et al., 2019)

- **A given sentence**:

  `move the vat over the hot fire`

- **A set of instances we create**:

  1. Input = `[MASK] the vat over the hot fire`

     Label = `move`
  2. Input = `move [MASK] vat over the hot fire`

     Label = `the`

     ...
  7. Input = `move the vat over the hot [MASK]`

     Label = `fire`

Figure 3: A depiction of an masked language model being used to predict word probabilities, when each word in the sentence is masked iteratively

[1]`https://huggingface.co/facebook/wav2vec2-large-robust-ft-libri-960h`

[2]`https://huggingface.co/facebook/wav2vec2-large-robust`

- Language Models (LMs) are commonly incorporated into ASR acoustic models to improve performance. These models can either be implicitly integrated in a fully sequence-to-sequence (seq-to-seq) fashion, or utilized through beam search decoding in CTC encoder only models.
- This paper uses the Listen, Attend and Spell (Chan et al., 2015) ASR model, which employs an attention based BiLSTM type architecture. In this paper the LAS model was trained with a CTC objective, as slightly more recent work demonstrated that the left-to-right constraints of CTC help the model learn speech text alignments (Hori et al., 2017). This is because regular attention mechanisms do not take advantage of the monotonic alignment between the inputted audio and outputted text (the order of words in the outputted text should always be given in the same order in which they are spoken) (Lugosch, 2020).
- LMs used for decoding and beam search re-scoring in ASR models are often unidirectional, processing the input sequence from left-to-right. Bi-directional models offer the advantage of being able to assess the probability of a word in the context of the entire sentence, not just the words prior.
- Bi(directional)-LSTM LMs have been previously applied to ASR however "there is no interaction between the past and the future words in the biLMs" as the forward and backward representations are not fused
- The BERT (Devlin et al., 2018) architecture combats these issues. BERT model is also trained on extremely large corpus, with multi-headed attention blocks, allowing the model to better capture long-range token level dependencies (it can accurately model the relationships between words that are far away from each other)

- This paper looks at using ~~the BERT model~~ a Bi-directional self-attention based model, to re-rank candidate transcribed sentences. Their technique utilizes BERT's Pre-training task, iteratively masking each word in a sentence, and summing the probabilities of each target to attain an overall "likelihood" of the sentences. This process is shown in figure 3
- This score is linearly combined with the prior score attributed to the sentences by the LAS model, with some weighting attributed to each probability.
- These new scores are then used to re-rank the candidate sentence and attain a new (hopefully) more likely sentence. In this paper they trained ~~the BERT model~~ a model similar to BERT on the libre-speech corpus, along with a range of other uni-directional LMs, with their proposed methodology showing the best performance.

On the first few passes of this paper I assumed they used **THE** BERT model, however on close examination it seems they use the same/or similar architecture (this is not made super clear), with a different training regime. This is a bit misleading given the title (probably done for citations):)

What are the implications of this work with regard to the mini-project?

A similar technique to this could be applied to any ASR system to find more likely word sequences. Although they train ~~BERT~~ a model similar to BERT on the Librespeech corpus this may not be necessary, and simply using a pre-trained BERT model would likely be sufficient to see improvements. Utilizing large attention based LMs such as BERT should be advantageous for our project, where deciphering the speaker's utterances often requires disambiguation through examining the context.

Masked Language Model Scoring (Salazar et al., 2020)

- Similar to the previously mentioned paper (Shin et al., 2019), this work looks at utilizing Masked Language Models (MLM) i.e BERT (Devlin et al., 2018) and their Bi-directional capabilities, to score candidate ASR outputs.
- **The authors release their library** for ASR MLM scoring[3] !
- As with prior work the authors use pseudo-perplexity score sentences. These are obtained from iteratively masking each word, taking the log, and adding to the total at each step.
- **Unlike** previous work with MLM scoring (Shin et al., 2019) pre-trained models are used, rather than training an LM from scratch
- Model fusion, where the encoder and decoder (LM) are coupled in some fashion (see Toshniwal et al. (2018)), requires both models to be auto-regressive and share the same tokenization. With re-scoring methods any LM that can provide a probability for a given sequence may be used.

Model	**dev**		**test**	
	clean	other	clean	other
baseline (100-best)	7.17	19.79	7.26	20.37
GPT-2 (117M, cased)	5.39	16.81	5.64	17.60
BERT (base, cased)	5.17	16.44	5.41	17.41
RoBERTa (base, cased)	**5.03**	**16.16**	**5.25**	**17.18**
GPT-2 (345M, cased)	5.15	16.48	5.30	17.26
BERT (large, cased)	4.96	16.26	5.25	16.97
RoBERTa (large, cased)	**4.75**	**15.81**	**5.05**	**16.79**
oracle (100-best)	*2.85*	*12.21*	*2.81*	*12.85*

Figure 4: WERs on LibriSpeech after rescoring. Baseline lists and oracle scores are from Shin et al. (2019)

- Figure 4 displays the WER when using different out-of-the-box pre-trained models. The baseline represents choosing the best beam (beam-width is 100) according to the seq-to-seq ASR model. *Oracle* shows the WER when the beam with the lowest WER is selected every time (the best re-scoring possible).
- Results demonstrate that standard pre-trained models can be used effectively for ASR beam re-scoring. Additionally, GPT-2 shows the worst improvement over baseline, suggesting that the bi-directional pseudo-perplexities of BERT and RoBERTa are preferable.

[3]https://github.com/awslabs/mlm-scoring

- RoBERTa (Liu et al., 2019) shows the best re-scoring capabilities (marginally). The RoBERTa architecture and parameter count is identical to BERT (RoBERTa utilizes an improved pre-training regime).
- "Out-of-the-box rescoring may be hindered by how closely our models match the downstream text".

Model	**dev**		**test**	
	clean	other	clean	other
baseline (100-best)	7.17	19.79	7.26	20.37
uni-SANLM	6.08	17.32	6.11	18.13
bi-SANLM	5.52	16.61	5.65	17.44
BERT (base, Libri. only)	**4.63**	**15.56**	**4.79**	**16.50**
BERT (base, cased)	5.17	16.44	5.41	17.41
BERT (base, uncased)	5.02	16.07	5.14	16.97
+ adaptation, 380k steps	**4.37**	**15.17**	**4.58**	**15.96**
oracle (100-best)	*2.85*	*12.21*	*2.81*	*12.85*

Figure 5: Comparison of out-of-the-box pre-trained models with BERT model trained from scratch, and BERT model fine-tuned (training is performed on LibreSpeech text corpus)

- The effects of domain-mismatch are demonstrated in figure 5. Interestingly, BERT models trained on cased text perform worse. The LibreSpeech 4GB text corpus contains no punctuation, which hurts performance in the BERT model trained on cased text.
- Performance on the BERT-base-uncased model in comparative to that of the large cased RoBERTa model. There is not an uncased version of RoBERTa. Unfortunately, the authors do not present results for BERT-large-uncased, presumably this would offer further reduced WER.
- Fine-tuning BERT on libre-speech improves performance. Training from scratch does not offer any advantages over fine-tuning, however it does outperform out-of-the-box models.

- Although not really relevant to our project; the authors train BERT to compute sentence probabilities in one forward pass, removing the need for iterative masking. This offers massive inference speed improvements. The BERT model is trained using a student-teacher setup, where the student model is trained to predict the sentence probability given by the teacher model using iterative masking.

What are the implications of this work with regard to the mini-project?

This work shows that using out-of-the-box pre-trained models to re-score candidate sentences can improve accuracy. This means a benchmark for our target domain can be easily obtained that integrates this approach e.g Audio $\longrightarrow$ Wav2vec2 $\longrightarrow$ n-gram LM $\longrightarrow$ BERT-rescoring $\longrightarrow$ transcript $\longrightarrow$ WER.

MLM re-scoring is pipeline independent and as long as there is a set of candidate sentences for a given input produced by the ASR system, this technique can be used to re-rank them and likely improve WER.

The authors demonstrate that fine-tuning BERT to better match the target domain improves WER; however, the domain that BERT is fine-tuned on is of the same characteristics (books, librespeech text corpus) to the target domain of LibreSpeech. It may be worth comparing performance between BERT large- and base- uncased on our target domain. Using BERT large uncased without finetuning may be sufficient, and there may be little relative (time+resources vs advantage) improvement from finetuning on librespeech. If it is possible to obtain a large text corpus that has similar characteristics to our oral history data (not likely??), then it would be worth fine-tuning BERT.

A paper by Futami et al. (2021) presents a comparable approach, however this is more complex and therefore less suited to our project.

Efficiently Fusing Pretrained Acoustic and Linguistic Encoders for Low-resource Speech Recognition (Yi et al., 2021)

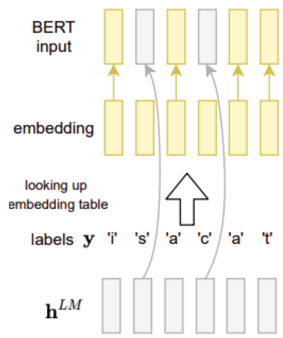

Figure 6: Depiction of acoustic representations being integrated into a BERT encoder. Acoustic representations (h^{AC}) are passed through a fully connected layer, mapping $h^{AC} \longrightarrow h^{LM}$. These representations (h^{LM}) are randomly mixed into the BERT encoder during training

- This work looks at fusing acoustic encoders (Wav2vec2) with linguistic encoders (Bert), for speech domains where there exists little labeled data
- The authors categorize ASR approaches into two sets: Pipeline methods, that use separated acoustic and linguistic models; and, end-to-end models that integrate all components into one, this generally involves a seq-to-seq model with an encoder-decoder framework that intakes speech and outputs text.
- These encoder-decoder frameworks have the capacity to outperform pipeline methods on most public datasets. However, these networks require large amounts (100s of hours) of transcribed speech to perform effectively on the target domain, which is problematic for low-resource ASR.
  - Transfer learning (where knowledge learned from other tasks is applied to the target one) is one method of adapting these models to low-resource domains, this *generally* (See SpeechStew (Chan et al., 2021)) requires "domain-similar" labelled data to pre-train the model on, which may be hard to find.

- Another method (Jiang et al., 2019) is to train the acoustic encoder in an unsupervised fashion (masked predictive coding) on unlabeled data, then add the decoder to the model while fine-tuning on labeled data. However, the decoder cannot be pre-trained separately as it relies on the acoustic representations from the encoder. Consequently, the decoder is only able to learn a model of language during labelled fine-tuning.

- Pipeline methods (including wav2vec2) require much less labelled data due to the effectiveness of self-supervised pre-training techniques, however pipeline components (i.e the language model) are often combined through some fixed weighting, which is "inflexible"

- Because of these factors the authors choose to explore better methods of combining linguistic encoders (BERT) with acoustic models. "The fused model has been separately exposed to adequate speech and text data, so that it only needs to learn the transfer from speech to language during fine-tuning with limited labeled data."

- During fine-tuning a random selection of wav2vec2's Acoustic representations are randomly mapped into BERT's input layers. These two networks are connected through a CIF (continuous-integrate-and-fire) (Dong and Xu, 2019) mechanism, which is a varient of Attention, designed to combat its monotonic alignment issues.

- Figure 6 depicts how the encoders are fused during fine-tuning. Initially BERT will not know how to interpret these representations (h^{LM}), and will "view" them as masked input, mainly relying on the transcription labels to make predictions. Through fine-tuning BERT is able to "learn" the meaning of the h^{LM} representations, and utilize them along with prior linguistic knowledge to make predictions.

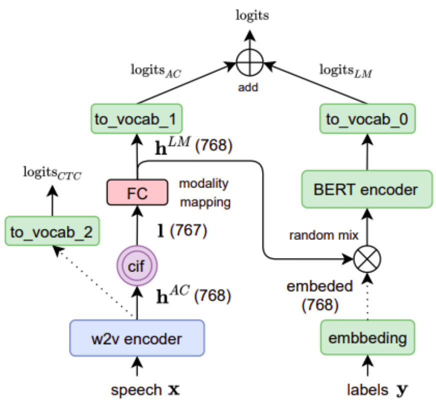

Figure 7: Depictions of the entire fusion network. Dotted lines represent modules that are ignored during inference

- Figure 7 presents the entire network. Logits from both encoders are combined and the entire network is trained through cross-entropy loss. The Wav2vec2 encoder receives additional supervision from its regular CTC task.

What are the implications of this work with regard to the mini-project?

This approach shows reasonable improvements on some low-resource data-sets compared to a selection of previous work. The authors provide good justification for their approach of acoustic and LM encoder fusion for low-resource settings. More recent work (Zheng et al., 2021) provides a direct comparison with the previously mentioned BERT re-scoring (Shin et al., 2019) system on a (different) range of low-resource languages. Results showed that Yi et al. (2021)'s fusion approach performed worse than BERT re-scoring method in all settings. Catastrophic forgetting (where knowledge learnt during pre-training is lost during fine-tuning) is likely the reason for this models comparatively poor performance. Fusing the Wav2vec2 directly with BERT's input may cause linguistic knowledge learnt via pre-training to be forgotten (Zheng et al., 2021).

Selecting a simpler and more effective approach such as the BERT re-scoring method would be best for our project. The Wav-BERT (Zheng et al., 2021) system fuses both models (wav2vec2 + BERT) through a "representation aggregation module" and a Gated Attention operation (Xue et al., 2019; Zhang et al., 2018), that helps overcome the catastrophic forgetting issue. Wav-BERT shows improved results over previous models on a selection of low-resource languages. The improvements seen from Wav-BERT are still only marginal when compared against the much simpler re-scoring (Shin et al., 2019) methods; additionally, the implementation Wav-BERT is likely outside the scope of our project.

Conformer: Convolution-augmented Transformer for Speech Recognition (Gulati et al., 2020)

- Transformers have enjoyed success on a range of sequence modelling tasks due their ability capture long range dependencies. However these networks are less capable of extracting "fine-grained local feature patterns".
- Convolutional networks excel at capturing local information, hence their uptake in computer vision, however they require many layers and parameters to capture a sup-optimal global context.
- The authors: "hypothesize that both global and local interactions are important for being parameter efficient. To achieve this, we propose a novel combination of self-attention and convolution will achieve the best of both worlds"
- The paper introduces the Conformer network that utilizes convolutions and attention to produce their encoder, and an LSTM based decoder
- The Conformer improves on previous work including LAS (Chan et al., 2015) and Transformer-Transducers (Zhang et al., 2020a) networks on the Librespeech dataset
- This architecture utlizes SpecAugment (Park et al., 2019), which is method used for data augmentation in ASR which has led to improved performance on models such as LAS (Chan et al., 2015).

What are the implications of this work with regard to the mini-project?

This version of the Conformer network has been super-seeded by more recent models including Wav2vec2. Additionally, the approach used in this paper only utilizes labeled data, which may make it harder to adapt to low-resource settings. The author's investigate different parameter sizes with this model and see reasonable results with model sizes as low as 10M parameters, training a low parameter model should require less data and resources.

This is methodology presents good results, and it is worth being aware of this architecture and paper as later literature builds upon this. A Wav2vec2 based approach should still be more suited to our project, however approaches using this architecture may be worth considering. Wav2vec2 offers the clear advantage of unsupervised pre-training but requires additional methods to incorporate linguistic models for optimal performance.

SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition (Park et al., 2019)

- Data Augmentation (DA) has been applied in state-of-the-art systems for computer vision tasks, and has also seen success in the speech domain. DA can improve the robustness of models, and can be easily implemented for domains that use continuous data.
- SpecAugment can be directly applied to the feature inputs of a model. The authors select deformations for DA that help ASR networks learn usefulel features, and improve robustness to: temporal deformations, partial loss of frequency information, and partial loss of speech segments. The following mentions each of the deformations performed, see the paper for more details:
  - Time Warping
  - Frequecy Masking
  - Time Masking
- Results from this work show that the SpecAugment DA method can improve the WER of a network such as LAS Chan et al. (2015). Their approach shows sizeable improvements to the original networks performance, outperforming **prior** work.
- The authors not that Time Warping, although helpful, is not a major factor in the performance improvements, and this deformation can be dropped given any budgetary/hardware limitations.
- This DA method can cause networks that previously overfitted on a given dataset, to underfit. This means deeper networks can be trained on smaller datasets using this approach.

What are the implications of this work with regard to the mini-project?

DA, and particularly the SpecAugment method, may be helpful for our project. Such techniques may increase the robustness of our model, and increase/augment the amount of data we have. SpecAugment is integrated into the current SOTA on librespeech (Zhang et al., 2020b). These techniques would be trivial method to implement as there exists an implementation of this method on GitHub[4] which allows for integration with Pytorch and Tensorflow based models.

Pseudo-Labeling

This section will overview a selection of papers that explore the use of pseudo-labelling for the task of semi-supervised ASR.

[4]https://github.com/DemisEom/SpecAugment

SELF-TRAINING FOR END-TO-END SPEECH RECOGNITION (Kahn et al., 2019)

Pseudo labelling is a technique that utilizes a trained model to provide labels for unlabeled data, this process works as follows:

1. We have a labelled dataset D, unlabelled audio X, and unpaired text data Y
2. Train an acoustic model on paired labelled D
3. Train a language model on unpaired text data Y
4. Combine the acoustic model and the language model and generate psuedo-labels for unlabelled audio X, creating new dataset $\bar{D}$
5. Train new acoustic model on a combination of D and $\bar{D}$

The authors employ a heuristic based filtering method, and a log likelihood based confidence score, to remove undesirable examples from $\bar{D}$.

Model ensembles can be used to decrease the impact of sample noise (inaccurate psuedo-labels). This can be performed using the process described in the above list, using N randomly initialized models, producing a collection: $\{\bar{D}_1, ...\bar{D}_N\}$. When training the final acoustic model, psuedo-labels are uniformly sampled from one of the N models each epoch.

The authors filter out text from unpaired dataset Y, that shares an overlap with data from D used to train the initial acoustic model. This is done to make the task more realistic, where the available unpaired audio and text are not of similar domains. Training a LM on near-domain unpaired text data (if available) is desirable for this approach.

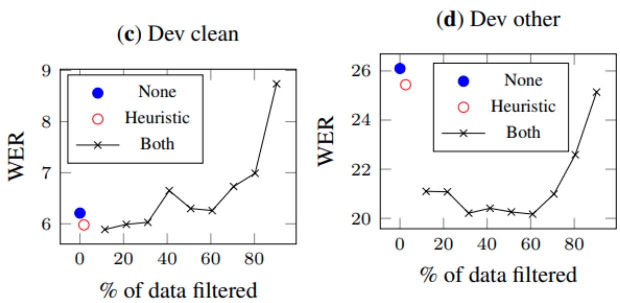

Figure 8: The effect of filtering noisy psuedo-lables on the WER

Figure 8 demonstrates the benefits of filtering the generated psuedo-labels when working with “unclean” audio. Filtering 10% of the psuedo-lables via the heuristic and confidence based measures lead to a sizeable drop in WER when working with noisy audio. The optimal amount of filtering for the noisy data (d) was 60%, although this will of course vary given a different set of data.

Figure 9 shows the benefits obtained from utilizing an ensemble of models for psuedo-labelling.

- Using more models is beneficial for both sets in reducing WER, especially when only using the heuristic based filtering method.
- When only working with heuristic filtering on the noisy data-set many pseudo-labels will inaccurate.

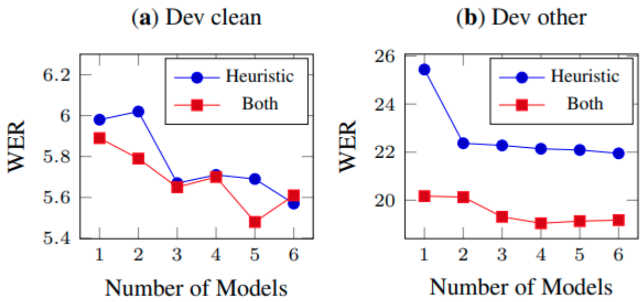

Figure 9: WER with respect to number of models in ensemble

- Results from figure 9 demonstrate that model ensembles for psuedo-labelling decreases the impact that inaccurate labels have on model performance.
- The authors hypothesise that the sample ensembling method is effective because the model is exposed to a different set of labels each epoch, which helps prevent the model from over-fitting to any inaccurate labels

This paper presented an applicable method of easily leveraging unpaired audio and text data to perform semi-supervised training. Experiments involving the model ensembles demonstrate that labels sampled from multiple models is desirable, especially for noisy data; however, training multiple acoustic models may be resource/time intensive depending on the parameter size and amount of data.

Although it is not mentioned in this work, the approach of Monte-Carlo (MC) Dropout (Gal and Ghahramani, 2016), could be an reasonable alternative to the model ensembling method. MC Dropout involves applying dropout at inference time and performing N forward passes through the model. It is only necessary to apply dropout to the last layer of a network for this approach, which saves on computation. Due to model uncertainty, labels of poor quality will vary between each forward pass because of the variation introduced via dropout. Hence, this may have similar benefits to the ensembling approach with regard to preventing over-fitting to inaccurate pseudo-labels. Because MC Dropout can be used as an approximation of model uncertainty, this technique could additionally be applied to help to identify, and remove, inaccurate pseudo-labels.

Iterative Pseudo-Labeling for Speech Recognition (Xu et al., 2020)

This paper builds on the pseudo-labelling methodology presented in the work mentioned above (Kahn et al., 2019). For a explanation of psuedo-labelling see the enumerated list given in the previous section.

- A new, more accurate, acoustic model can be obtained via training on a combination of labelled dataset D and data-set produced using pseudo-labels $\bar{D}$
- As the title of this paper suggests, this process of psuedo-labelling can be continually repeated using the acoustic model obtained from the previous iteration of semi-supervised training.
- The authors refer to this process as iterative psuedo-labelling (IPL). IPL can boost performance in standard and low-resource settings, and can scale to large amounts of unlabeled data.

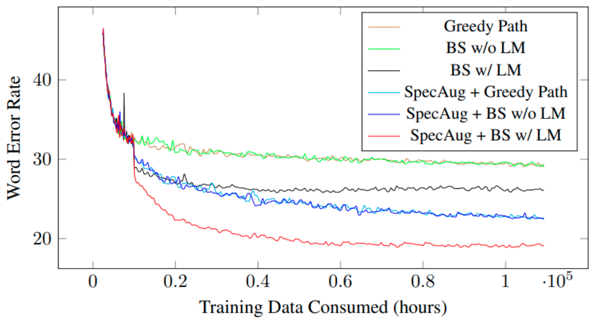

Figure 10: WER on dev-other with different IPL training strategies. Beam-search (BS) decoding is performed with an 4-gram LM.

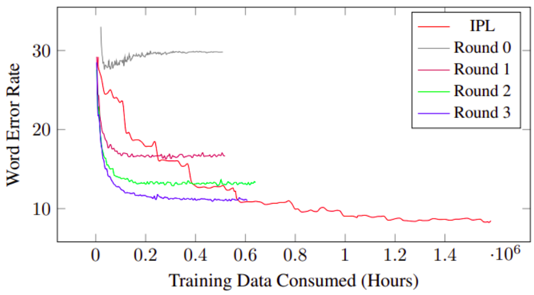

Figure 11: It is not optimal to train an acoustic model from scratch each psuedo-labelling iteration. The figure presents training from scratch at different rounds of pseudo-labelling vs iteratively fine-tuning the acoustic model (IPL)

- As depicted in figure 10 the authors experiment with different training and decoding strategies for IPL, and plot WER as a function of training data.
- Figure 10 demonstrates that beam search with an 4-gram LM and SpecAugment for DA (Park et al., 2019) (mentioned previously) is more effective than other configurations trialled by the author for IPL.
- The authors repeat the pseudo-labelling process every 10 epochs
- If the labelled data-set is sufficiently large, it may be expensive/time-consuming to psuedo-label the entire set. The authors conduct ablation experiments that conclude that it may only be necessary to psuedo-label %20-%40 of the unlabeled data each psuedo-labelling iteration. IPL using a randomly sampled portion of %20-%40 of the unlabeled data each iteration will converge to the same WER in a similar time-frame as using 100%. This may or may not be relevant depending on the amount of unlabeled data which we work with
- It is **not** necessary to train an acoustic model from scratch each pseudo-labelling iteration, in fact continually fine-tuning the existing acoustic model on the new labels converges to the optimal WER (see figure 11)

In this work the LM is trained on corpora with similar characteristics as the unlabeled audio. Using an LM trained on a different style of corpora to our target domain during IPL may undesirably bias the acoustic model towards the LM's training distribution (Likhomanenko et al., 2020; Xu et al., 2020). If *near* target domain text data is not available it **might** be worth performing IPL training without a LM, using SpecAugment and greedy CTC decoding (see figure 10).

Results from this paper showed that there may be no need to continually re-train the acoustic model; however, this may be contingent on the size of the training corpora, the authors use an extremely large amount of training data to get good results. Re-training the acoustic model may help prevent over-fitting if the corpora size used for IPL based training is fairly small.

Unsupervised Domain Adaptation For Speech Recognition via Uncertainty Driven Self-Training (Khurana et al., 2020)

- Similar procedure as general pseudo-labelling approaches
- The authors use the MC Dropout uncertainty estimation (Gal and Ghahramani, 2016) discussed earlier to filter potentially innacurate samples
- Variation between model outputs for each forward pass (with dropout enabled) is used to quantitate uncertainty, and uncertain samples are removed

The level of uncertainty that is acceptable for a pseudo-label would have to be trained on some sort of test set as this would likely vary between models.

What are the implications of this work with regard to the mini-project?

The pseudo-labelling strategies discussed all present useful, and applicable methods for utilizing our unlabeled data. Other work worth considering that functions on a similar premise is as follows: Liu et al. (2018); Likhomanenko et al. (2020); Park et al. (2020); Hsu et al. (2020); Higuchi et al. (2021)

A COMPARISON OF TECHNIQUES FOR LANGUAGE MODEL INTEGRATION IN ENCODER-DECODER SPEECH RECOGNITION (Toshniwal et al., 2018)

- This paper looks at the various techniques for integrating LMs into acoustic models (AM). Direct comparisons of performance for each approach are given, and various terminology is introduced and explained.
- The integration methods discussed are as follows:

Shallow Fusion

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} \log p(\mathbf{y}|\mathbf{x}) + \lambda \log p_{LM}(\mathbf{y}) \tag{1}$$

Shallow fusion integration is where the LM is integrated **only** at inference time. Shallow fusion is depicted in the above equation, log likelihoods from the acoustic and language models are combined, with value λ representing some weighting that is attributed to the LM. In this equation $\mathbf{y}$ denotes text and $\mathbf{x}$ denotes audio.

Shallow fusion can be advantageous over other approaches as it does not require training LMs and AMs jointly on transcribed audio. This can make it more computationally efficient. Additionally this allows for the AM to use self-supervised pre-training techniques such as contrastive predictive coding.

Shallow fusion is currently being used in our baseline Wav2vec2 system with a kenlm model trained on the tedium corpus being used for decoding. This baseline uses a λ value of 1 (equal weighting to LM and AM), however these parameters are usually optimized on a training set.

Deep Fusion

Similar to shallow fusion, deep fusion integration methods utilize already trained AM and LM. Instead of a linear interpolation of the AM and LMs scores, deep fusion connects the hidden states of both models directly using some sort of weight matrix. Further training has to be performed in order to learn appropriate weights for the new module, however all other model parameters can be fixed/frozen, as the model only needs to learn how to transform the acoustic representation into a vector space which is interpretable for the LM. As the majority of the model weights can be frozen, this approach can also be fairly computationally inexpensive.

Cold Fusion

Cold fusion integrates a randomly initialized (untrained) AM with a decoder that utilizes the parameters from a pre-trained LM, this is similar to some approaches in Machine Translation.

Results

This work found shallow fusion to be the most effective approach. Using cold fusion obtained results in a similar range.

What are the implications of this work with regard to the mini-project?

Shallow fusion is a simple method for LM integration, and in many cases will also present the best performance. This work shows that complex LM integration may be unnecessary for our project. Alternatively a fully e2e encoder-decoder approach could be taken. The results from this work could of course vary given a mismatch between LM and AM training domain, available labelled data, and cold fusion model design choices.

Future Reading/Papers To Be Added:

- THE ACCENTED ENGLISH SPEECH RECOGNITION CHALLENGE 2020: OPEN DATASETS, TRACKS, BASELINES, RESULTS AND METHODS (Shi et al., 2021)
- Pushing the Limits of Semi-Supervised Learning for Automatic Speech Recognition (Zhang et al., 2020b)
- END-TO-END ASR: FROM SUPERVISED TO SEMI-SUPERVISED LEARNING WITH MODERN ARCHITECTURES (Synnaeve et al., 2019)
- Two-Pass End-to-End Speech Recognition

- Improved Noisy Student Training for Automatic Speech Recognition (Park et al., 2020)
- Semi-Supervised Speech Recognition via Local Prior Matching (Hsu et al., 2020)
- Adversarial Training of End-to-end Speech Recognition Using a Criticizing Language Model (Liu et al., 2018)
- SlimIPL: Language-Model-Free Iterative Pseudo-Labeling (Likhomanenko et al., 2020)
- Recent Developments on ESPnet Toolkit Boosted by Conformer (Useful toolkit with tips for training conformer based models) (Guo et al., 2020)

References

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *CoRR* abs/2006.11477.

William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals. 2015. Listen, attend and spell. *CoRR* abs/1508.01211.

William Chan, Daniel S. Park, Chris Lee, Yu Zhang, Quoc V. Le, and Mohammad Norouzi. 2021. Speechstew: Simply mix all available speech recognition data to train one large neural network. *CoRR* abs/2104.02133.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR* abs/1810.04805.

Linhao Dong and Bo Xu. 2019. CIF: continuous integrate-and-fire for end-to-end speech recognition. *CoRR* abs/1905.11235.

Hayato Futami, Hirofumi Inaguma, Masato Mimura, Shinsuke Sakai, and Tatsuya Kawahara. 2021. Asr rescoring and confidence estimation with electra.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning.

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*. Association for Computing Machinery, New York, NY, USA, ICML '06, page 369–376.

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.

Pengcheng Guo, Florian Boyer, Xuankai Chang, Tomoki Hayashi, Yosuke Higuchi, Hirofumi Inaguma, Naoyuki Kamo, Chenda Li, Daniel Garcia-Romero, Jiatong Shi, Jing Shi, Shinji Watanabe, Kun Wei, Wangyou Zhang, and Yuekai Zhang. 2020. Recent developments on espnet toolkit boosted by conformer.

Awni Hannun. 2017. Sequence modeling with ctc. *Distill* Https://distill.pub/2017/ctc.

Yosuke Higuchi, Niko Moritz, Jonathan Le Roux, and Takaaki Hori. 2021. Momentum pseudo-labeling for semi-supervised speech recognition.

Takaaki Hori, Shinji Watanabe, Yu Zhang, and William Chan. 2017. Advances in joint ctc-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM. *CoRR* abs/1706.02737.

Wei-Ning Hsu, Ann Lee, Gabriel Synnaeve, and Awni Hannun. 2020. Semi-supervised speech recognition via local prior matching.

Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. 2021. Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training. *CoRR* abs/2104.01027.

Dongwei Jiang, Xiaoning Lei, Wubo Li, Ne Luo, Yuxuan Hu, Wei Zou, and Xiangang Li. 2019. Improving transformer-based speech recognition using unsupervised pre-training. *CoRR* abs/1910.09932.

Jacob Kahn, Ann Lee, and Awni Hannun. 2019. Self-training for end-to-end speech recognition. *CoRR* abs/1909.09116.

Sameer Khurana, Niko Moritz, Takaaki Hori, and Jonathan Le Roux. 2020. Unsupervised domain adaptation for speech recognition via uncertainty driven self-training. *CoRR* abs/2011.13439.

Tatiana Likhomanenko, Qiantong Xu, Jacob Kahn, Gabriel Synnaeve, and Ronan Collobert. 2020. slimipl: Language-model-free iterative pseudo-labeling. *CoRR* abs/2010.11524.

Alexander H. Liu, Hung yi Lee, and Lin shan Lee. 2018. Adversarial training of end-to-end speech recognition using a criticizing language model.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Loren Lugosch. 2020. Sequence-to-sequence learning with transducers.

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. *Interspeech 2019* .

Daniel S. Park, Yu Zhang, Ye Jia, Wei Han, Chung-Cheng Chiu, Bo Li, Yonghui Wu, and Quoc V. Le. 2020. Improved noisy student training for automatic speech recognition. *Interspeech 2020* .

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* .

Xian Shi, Fan Yu, Yizhou Lu, Yuhao Liang, Qiangze Feng, Daliang Wang, Yanmin Qian, and Lei Xie. 2021. The accented english speech recognition challenge 2020: Open datasets, tracks, baselines, results and methods. *CoRR* abs/2102.10233.

Joonbo Shin, Yoonhyung Lee, and Kyomin Jung. 2019. Effective sentence scoring method using bert for speech recognition. In Wee Sun Lee and Taiji Suzuki, editors, *Proceedings of The Eleventh Asian Conference on Machine Learning*. PMLR, volume 101 of *Proceedings of Machine Learning Research*, pages 1081–1093.

Gabriel Synnaeve, Qiantong Xu, Jacob Kahn, Edouard Grave, Tatiana Likhomanenko, Vineel Pratap, Anuroop Sriram, Vitaliy Liptchinsky, and Ronan Collobert. 2019. End-to-end ASR: from supervised to semi-supervised learning with modern architectures. *CoRR* abs/1911.08460.

Shubham Toshniwal, Anjuli Kannan, Chung-Cheng Chiu, Yonghui Wu, Tara N Sainath, and Karen Livescu. 2018. A comparison of techniques for language model integration in encoder-decoder speech recognition.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding.

Qiantong Xu, Tatiana Likhomanenko, Jacob Kahn, Awni Hannun, Gabriel Synnaeve, and Ronan Collobert. 2020. Iterative pseudo-labeling for speech recognition.

Lanqing Xue, Xiaopeng Li, and Nevin L. Zhang. 2019. Not all attention is needed: Gated attention network for sequence data. *CoRR* abs/1912.00349.

Cheng Yi, Shiyu Zhou, and Bo Xu. 2021. Efficiently fusing pretrained acoustic and linguistic encoders for low-resource speech recognition. *CoRR* abs/2101.06699.

Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. 2018. Gaan: Gated attention networks for learning on large and spatiotemporal graphs. *CoRR* abs/1803.07294.

Qian Zhang, Han Lu, Hasim Sak, Anshuman Tripathi, Erik McDermott, Stephen Koo, and Shankar Kumar. 2020a. Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss.

Yu Zhang, James Qin, Daniel S. Park, Wei Han, Chung-Cheng Chiu, Ruoming Pang, Quoc V. Le, and Yonghui Wu. 2020b. Pushing the limits of semi-supervised learning for automatic speech recognition.

Guolin Zheng, Yubei Xiao, Ke Gong, Pan Zhou, Xiaodan Liang, and Liang Lin. 2021. Wav-bert: Cooperative acoustic and linguistic representation learning for low-resource speech recognition.