Author: Cécile Macaire
Date: 03/31/2021

Experimental setup and evaluation of the results from fine-tuning XLSR-53 with the Na corpus

From previous works, ESPnet reported a Character Error Rate (CER) on the same dataset (i.e. Na spontaneous narratives) of 14.5% [1]. The following work reports a CER of 5%.

A. Experimental setup

The experiment is based on https://huggingface.co/blog/fine-tune-xlsr-wav2vec2 which fine-tuned the XLSR-53 model on the CommonVoice dataset. The following code was used: https://github.com/macairececile/internship_lacito_2021/blob/main/Fine-Tune%20XLSR-Wav2Vec2%20-%20Na.ipynb

➢ Corpus

Na corpus	**Training**	**Val**	**Test**
Number of files	1738 pairs of <wav,trans>	372 pairs of <wav,trans>	335 pairs of <wav,trans>
Size	≈ 3 hours	≈ 1 hour	≈ 55 minutes
Speaker type	1 female speaker		

Preprocessing from na.py script provided by Oliver Adams (remove punctuations, bad na symbols, space char, …). The audios are resampled in 16kHz in mono.

➢ Vocabulary

JSON file with the following information (each character is mapped to a unique integer).

{"ɟ": 0, "e": 1, "…": 2, "w": 3, "ɬ": 4, "ɻ": 5, "ˌ": 6, "õ": 7, "ʐ": 8, "ʋ": 9, "ɨ": 10, "ɖ": 11, "ˈ": 12, "m": 13, "z": 14, "ʂ": 15, "j": 16, "f": 17, "ʈ": 18, "ɤ": 19, "̃": 20, "t": 21, "ĩ": 22, "d": 23, "ɣ": 24, "ẽ": 25, "ʑ": 26, "i": 27, "ʰ": 28, "g": 29, "ə": 30, "h": 31, "p": 32, "ʁ": 33, "ɯ": 34, "ɭ": 35, "|": 36, "[": 37, "o": 38, "ɛ": 39, "ɑ": 40, "n": 41, "u": 42, "b": 43, "k": 44, "æ": 45, "ŋ": 46, "s": 47, "l": 48, "ɲ": 49, "ɔ": 50, "ɻ": 51, "q": 52, "ɳ": 53, "ɛ": 54, "c": 55, "[UNK]": 56, "[PAD]": 57}

➢ Training

model	facebook/ wav2vec2-large-xlsr-53
attention_dropout	0.1
hidden_dropout	0.1
feat_proj_dropout	0.0
mask_time_prob	0.075
layerdrop	0.1
gradient_checkpointing	True
ctc_loss_reduction	mean

Hyperparameters

gradient_accumulation_steps	2
evaluation_strategy	steps
num_train_epochs	50
fp16	True
save_steps	1000
eval_steps	50
learning_rate	3e-4
warmup_steps	500

Training arguments

Author: Cécile Macaire
Date: 03/31/2021

➢ Information

The training time with 1 GPU provided by Google Collab was 7.30 hours.
To compute the results (i.e. the prediction of the model on the test set), it took around 1 hour. They are stored in a CSV file with two columns: *Reference* and *Prediction*.

B. Evaluation

The evaluation was conducted thanks to the evaluation script available here:
https://github.com/macairececile/internship_lacito_2021/blob/main/evaluation.py

From the evaluation.py script:

- by calling *lev_dist* - it computes the levenshtein distance between each reference and its corresponding prediction (the sentence itself, and the list of words from each sentence). The levenshtein average distance of the list of levenshtein distance per words is also computed.
- by calling lev_dist_notones - it does the same computation as before but tones are not taken into account (i.e. {"˩", "˥", "˧", "˧˥", "˩˧", "˥˩", "˧˩"}.
- by calling *eval_char* - it computes the levenshtein distance, the F-score, Precision, Recall and confusion matrix between the list of characters from the reference and the corresponding prediction
  Example
  Ref: ['m', 'm', 'm', '…', 'ʁ', 'o', 'ɬ', 'q', 'ʰ', 'w', 'ɤ', 'ɻ', '<SP>', 'l', 'e', 'ɬ', 'p', 'v', 'ˌ', 'ɻ', 'k', 'ʰ', 'ɯ', 'ɻ']
  Pred: ['m', 'm', '*', '…', 'ʁ', 'o', 'ɬ', 'q', 'ʰ', 'w', 'ɤ', 'ɻ', '<SP>', 'l', 'e', 'ɬ', 'p', 'v', 'ˌ', 'ɻ', 'k', 'ʰ', 'ɯ', 'ɻ']
  where '*' refers to an Insertion or a Deletion.

A pdf is available with the references and the predictions to better see the Insertion, Deletion and Substitution (in red): https://github.com/macairececile/internship_lacito_2021/blob/main/results/out_XLSR53.pdf
And the results in CSV files are in https://github.com/macairececile/internship_lacito_2021/tree/main/results

➢ Results

	Sentences		**Words**		**Characters**
	Tones	No tones	Tones	No tones	
Average Levenshtein distance	0.929	0.955	0.68	0.66	0.95
Average F-score					0.863
Average Precision					0.901
Average Recall					0.904

The highest, the better the results.

Bibliography

[1] Adams, O., Galliot, B., Wisniewski, G., Lambourne, N., Foley, B., Sanders-Dwyer, R., ... & Hill, N. (2020). User-friendly automatic transcription of low-resource languages: Plugging ESPnet into Elpis. *arXiv preprint arXiv:2101.03027*.