Optimizing Architectures For Continuous Emotion Recognition in Speech

Who:
cmarti88 (Christian Johann Martinez) lcsha (Lucas Sha), yliu622 (Yihong Liu), ptutsi (Pir Servan Tutsi)

Introduction:

Paper Overview, Objectives:

The basis for our project is the paper *Personalized Adaptation With Pre-Trained Speech Encoders.*

In this paper, the researchers propose novel architectural and technical modifications to speech encoder models to improve their performance at speech recognition tasks, surpassing prior benchmarks on the MSP-Podcast dataset.

The two techniques they propose are:

(1) Personalized Adaptive Pre-training:
- A method of making models 'personalize' and learn robust speech-speaker relationships during pre-training.
- Basically, vector/feature-ized speech utterances are fed through a pre-trained speech encoder [the authors use Meta's HuBERT], and the speaker features for these audio are combined with a learnable embedding layer.
- The results of this robust, personalized speech representation step are then used for fine-tuning for emotion recognition.

(2) Personalized Label Distribution Calibration:
- A method come test-time for making model predictions more robust.
- The authors identify label-shift—where the probability distribution of emotional indicators predicted for a test speaker differs substantially from those seen in the test-data—as a substantial problem models face in evaluating unseen speakers.
- So what PLDC does is the following: during evaluation, the model will also search back into saved training data-files to find speakers with the most similar embeddings as the currently evaluated speaker, and then use these to shift the mean of its predictions in order to make them more robust.

Why This Paper?

We chose this paper because we find the sort of HCI work it deals with very interesting and, of course, more and more relevant as AI seeps into our everyday lives. The paper was published by USC's Institute for Creative Technologies as part of their work in the field of Affective Computing.

The authors' description of the work's relevance resonated with us in particular—they noted that speech emotion recognition is not an obvious and easily generalizable task due to the variance in how people of different backgrounds and cultures express themselves. AI taking this into account is important to make sure AI tools and developments can be accessible to all.

What Type of Project Is It?

I think it's not super clear that it's one type of project or another! But, of course there are elements of basic regression supervised learning (using CCC loss which is basically MSE Loss on steroids for valence regression during the fine-tuning process, for example) but also self-supervised learning (which is what happens when you continue the pre-training process of the HuBERT model through PAPT) at play.

The harder part of the project, PAPT, will be self-supervised learning for sure.

Project Objectives:

The paper is a good baseline and gives a lot of food for thought, but we hope to extend the original scope of the paper in this project, and in particular experiment with different architectures that might be able to produce similar results but with *less computational power* required.

The authors' experimentation with different/modified architectures in their paper was relatively lean; ablation studies were primarily confined to where the embeddings for the speakers were added during the processing phases.

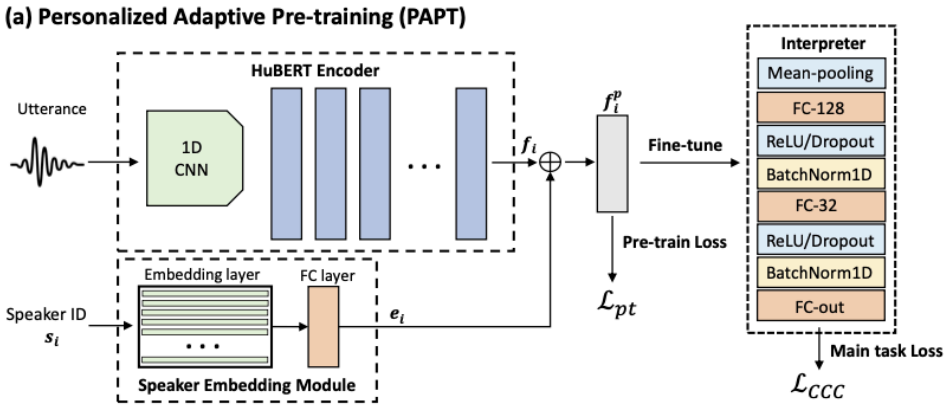

Figure 1: Computational graph of PAPT technique taken from the paper.

That is, the authors experimented with adding e_i, the speaker embedding, to the output of the 1D CNN, concatenating it with that output, and adding it to the output of the speech encoder—they found the latter worked best so that's why it is in their graph!

But we believe some further experimentation with both the speaker-representation as well as the fine-tuning architecture could reveal potentially untapped improvements in models' abilities to perform ER tasks.

Thus, we have the following objectives in mind for this project:

(1) Implement the original architecture of the paper and try to reproduce its results.
(2) Variations on speaker personalization:
    (a) We propose experimenting with more forms of how to represent speakers in the first place (how to personalize speech!)
    (b) Within the data, we could basically average (max-pool or actually taking the mean) out the utterance feature vectors generated by a featurizer for each speaker to show what is "common" to the speaker utterances, thus calculating the speaker features as an alternative form of personalization.
    (c) It is absolutely possible that raw features could be just as good as embeddings, depending on architecture and placement.
(3) Variations on architecture:
    (a) The authors use a fairly simple architecture (basically MLP) for their interpreter. We argue that a better baseline could be established by using a transformer where-in we feed a concatenation of speaker-features/embeddings + their utterances rather than simply stacking dense layers and using them to calculate the sum of speaker-features and embeddings.

Data:

MSP-Podcast is our top choice.

I know we are encouraged to use a different dataset if possible. However, there is a huge data scarcity of SER datasets, in particular **continuous** emotional recognition. E.g: the following paper as evidence: people are resorting to using ML models just to pre-process continuous labels from most of the existing discrete annotations (poor speech recognition researchers).

We have identified some possible datasets (LSSED, MuseTrust, EmoFilm), but these datasets do not seem to be annotated with valence, arousal, dominance scores, or if they are, are too small and the audio files too long so that pre-processing would be basically impossible (e.g this is unfortunately the case with MSP-Conversation corpus).

We can certainly alter our metrics from CCC (a continuous regression metric used by the authors of our paper) to CCE loss instead if needed, but we would be doing different forms of emotion recognition basically.

An additional factor of consideration: we have access to the source code, technically.

But… the source code is incomplete.

The researchers gave me the file as a Google doc, not as a Github Repo. It does not have the code for the PAPT and speech embedding components, basically.

I am not sure if perhaps they are not at liberty to share this part of the code (this was a paper funded by the US Army for its work on autonomous digital agents) for some reason or another or just lost track of it, so this part is missing from whatever I could re-implement anyways.

Related Work:

_____

Related Paper

A related paper that is cited by the authors of the paper we are using is '*Unsupervised Personalization of an Emotion Recognition System: The Unique Properties of the Externalization of Valence in Speech'*.

It focuses on the personalization problem as applied to valence regression tasks. (The authors found that personalization had the best effect on valence—it improved the valence prediction as high as 13.52%, while the improvements on arousal and dominance were less than 1.9% (on MSP-Podcast corpus))

The authors achieve these improvements by (1) finding, for each speaker in the test set, the speakers in the target set that have the most similar acoustic range and (2) implementing three strategies for adaptation, i.e., making the model speaker specific: unique speaker, oversampling, and weighting approaches. [Note the similarity to PLDC method in our paper!]

The unique speaker approach randomly selects from the data obtained from the selected speakers (i.e., the most similar speakers in the train set) without replacement so that a speaker in the train set can be found similar to more than one speaker in the test set.

The oversampling approach repeats the sentences from a selected speaker in the training set in the case where this speaker is found to be similar to more than one speaker in the test set. These similar speakers are then weighted more for better inference.

_____

Public Implementations

None found. As mentioned, the source code is not public and is apparently incomplete, so I doubt we will really find any public implementations out there.

Methodology:

_____

Training, Architecture, Techniques

We will be training our models/proposed architectures basically from scratch, except for loading the pre-trained HuBERT model and a Wav2VecFeaturizer. (The file we were given does not have any checkpoints from which we can load saved versions anyways, nor do we have optimal hyperparameters in the (partial) source code we have access to).

We will, as mentioned in **Introduction/***Project Objectives*, have a few different architectures and methodologies we want to experiment with concerning how to personalize speech.

We propose the following combinations of training, some of which are interested in efficiency and ease of implementation in which we explore how far we can get with less pre-training.

Fixed speaker features/Fixed HuBERT encoder output for speech-embeddings + Transformer Fine-Tuner

- easiest possible implementation requiring minimal compute; here, the only thing we actually train is the Transformer Fine-Tuner—here, we would basically cut out entirely the need for continuing the pre-training process on HuBERT, which is quite expensive

Learnable speaker embeddings/Fixed HuBERT encoder output for speech-embeddings + Transformer Fine-Tuner

- here, we do process speaker features into learned embeddings, but keep HuBERT as basically just a glorified preprocessor like before. Obviously a bit more compute needed since we learn not just Transformer fine-tuner but also the embedder.

PAPT re-implementation + Transformer Fine-Tuner/Original Interpreter Fine-tuner

- this is two methods in one, but basically we just try to re-implement the paper though testing out a transformer fine-tuner as well.
- This is obviously a more moderate compute project due to pre-training, and so we are interested in of course doing it but also seeing if smaller compute versions like the above could produce similar results.

Of course, come evaluation time, we will use PLDC on every model since it is just a good technique the authors propose for robust inference.

Other Training Discussion?

We haven't given efficiency too much thought yet. I think it's theoretically possible to just do the less expensive versions on my laptop, but will want to do the big pre-training on a Department Machine for sure!

Difficulties of Implementation

The hardest part to implement will be PAPT less so because of compute limitations and more so because, well, we don't exactly have the source code for that and HuBERT is a really interesting model that doesn't work like the standard things we've been implementing in class.

HuBERT has its own pre-training loss where-in it actually generates pseudo-labels of the data using K-means clustering and predicts this through self-supervised learning.

The authors note that this is equivalent to CE Loss over its masked frames but we're gonna have to figure out how to do this ourselves in our re-implementation so let's hope it works out, basically.

Metrics:

Experiments, Metric of Success Used

I think we mostly covered experimentation in the **Introduction** and **Methodology** sections. The metric of success we will use is CCC loss for Valence Estimation—the authors primarily care about predicting emotional Valence (just a positive negative continuous sliding scale of numbers, basically), so they use CCC loss applied to valence.

CCC loss is a form of regression loss that takes into account not just distance, but also covariance between the predictions and the actual. So we get a way more robust estimate and learning from using this over basic MSE.

Authors' Metrics of Success

The authors simply wanted to establish a better benchmark on MSP-Podcast in terms of *valence regression strength*.

Recall that they discussed the personalization gap as an issue to solve—they believed that bigger datasets would kind of erase the effect of mere fine-tuning, and so they believed PAPT would be a much more robust tool to make models understand personalization better.

Base, Target, Stretch

We think our base goal will be, at minimum, to be able to implement the simpler 2 alternative methodologies (using HuBERT as a glorified preprocessor, and both learnable/fixed embeddings versions) because this doesn't require too much computing power. Nonetheless, it could produce interesting results.

We don't want, of course, to stop there though. Target goal is that we do want to recreate the PAPT architecture proposed by the authors, and probably trying out a Transformer instead of their Interpreter first just to see what happens, since it is a different architecture.

The Stretch could be managing to make both Transformer and Interpreter versions work, since we don't have access to any checkpoints or optimal hyperparameter information and we all know how annoying hyperparameter tuning can be.

Ethics:

Broader Social Issues

Learning algorithms like the one we are implementing will undoubtedly become more prevalent as computers continue to become more integral to the lives of humans and human/computer interactions become more frequent.

On one hand, models that have a better grasp on human emotion could be seen as positive; it will allow the computer that is interacting with the person to do so with more nuance (e.g. it could respond with empathy in the case of an angry human).

On the other hand, If technology such as this continues to get better, we might see computers replacing humans in roles that have been traditionally reserved for humans, such as those that require compassion or customer service roles. This not only continues the trend of machines putting humans out of work, but the prospect of humans turning to support from machines instead of people that can truly understand and resonate with what they are dealing with is, in my eyes, a dark one.

Stakeholders

The paper we are re-implementing has, of course, its unique implementation of an algorithm that personalizes its emotion recognition to specific individuals; this is in fact what attracted us to the paper in the first place.

Attractive though it may be, it is not without its issues. The paper details the "vast variability in how people express their feelings through speech, which can depend on culture [1], gender [2], or age [3], among others."

With this, of course, comes the danger of our personalizations being learned in such a way that it reinforces sexist, racist, or otherwise harmful stereotypes not due to those stereotypes having any truth, but instead due to the feature inherent to the framework of our entire model and data.

It could be a really positive thing for people of many different backgrounds to be included in AI and be better understood by them, but perhaps highlighting the particular worries of stakeholders in vulnerable groups would be important in this project and if it makes mistakes or learns problematic patterns.

Division of Labor:

We will all work together on the PAPT implementation.

For the smaller less compute required alternatives we're interested in exploring, we will have something like Lucas/Johann on one version and Yihong/Servan on the other so that it's balanced out.