You’re your own best teacher: A Self-Supervised Learning Approach For Expressive Representations

Johan Vik Mathisen

May 5, 2024

0.1 Stage 1

As mentioned in the section on representation learning, one needs to determine a set of tasks one wish to evaluate on, in order to say anything about the quality of the representations. We evaluate the representations based on two tasks

0.1.1 Evaluation metrics

- **Reconstruction**: We evaluate the models ability to reconstruct the original data from latent representation. Success indicating perservation of information.
- **Downstream classification**: We evaluate the latent representations on its ability linear classification.
- **Training time**
- **Number of parameters**
-

0.1.2 Reconstruction

0.1.3 Classification

0.1.4 Codebook investigations

In the two tokenization models, how does the codebooks differ? Look at codebook utlization. Histograms across dimensions?

0.2 Stage 2

0.2.1 Evaluation metrics

- **IS**:
- **FID**:
- **Visual inspection**:
- **Token usage**:
- **Generating distribution**:

0.3 Ablation studies

0.3.1 Augmentation Reconstruction Weight

Here are the results of the ablation on the effect of "Augmentation Reconstruction Weight" on Stage 1. "Augmentation Reconstruction Weight" is the weight given to the reconstruction loss on the augmented branch. Tested weights 0.05, 0.1, 0.15 and 0.2. Augmentations [Window Warp, Amplitude Resize] and [Slice and Shuffle]. The weight has little effect on linear probe accuracy across the four datasets tested, and the two sets of augmentations. The effect on Validation reconstruction loss is small for all except FordA. It seems, not very surprisingly, that the choice of augmentations are of (much) greater importance.

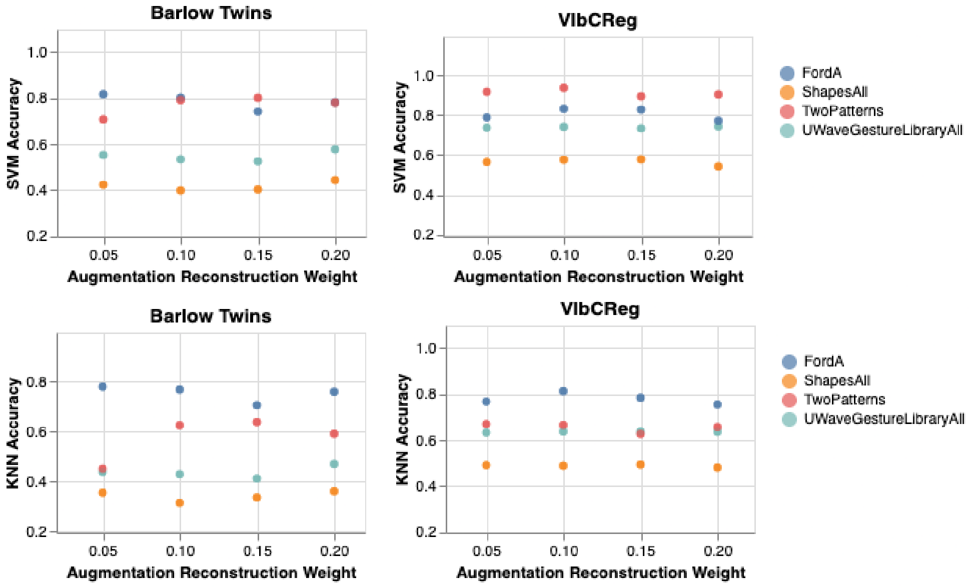

Figure 1: Augmentations: Window Warp and amplitude resize. Averaged across 2 runs. Trained for 250 epochs

0.3.2 Augmentation robustness

TODO: Download the Wandb data.

Plot for each dataset and each augmentation: Mean KNN / SVM / ReconsLoss against augReconsWeight. Color code according to SSL-model.

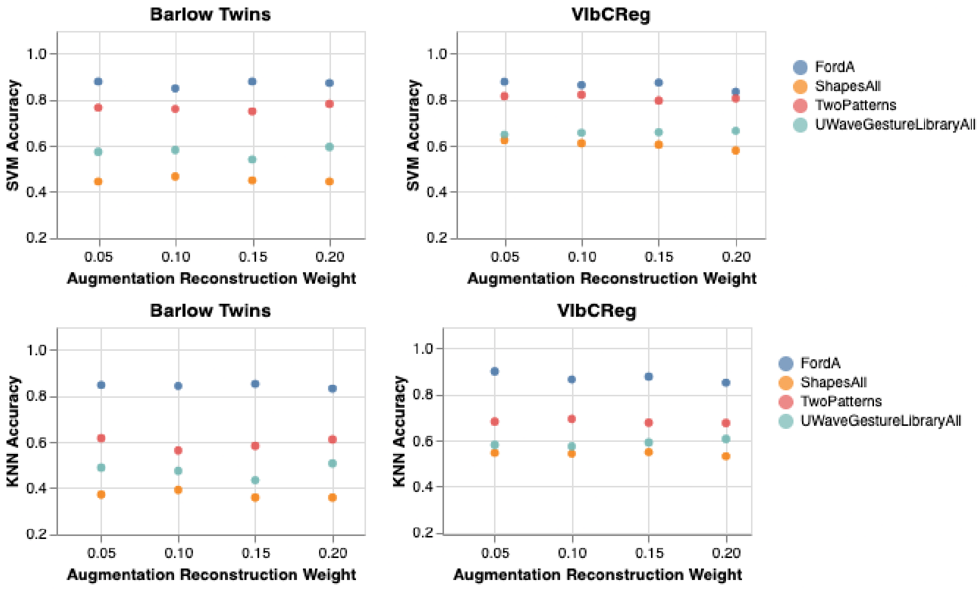

Figure 2: Augmentation: Slice and shuffle. Averaged across 2 runs. Trained for 250 epochs

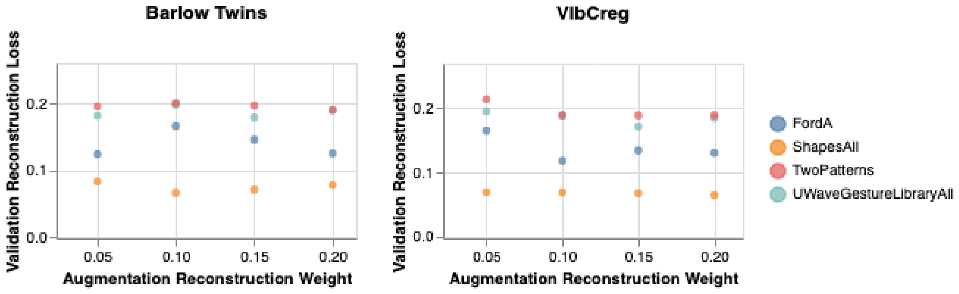

Figure 3: Augmentation: Window Warp and amplitude resize. Averaged across 2 runs. Trained for 250 epochs

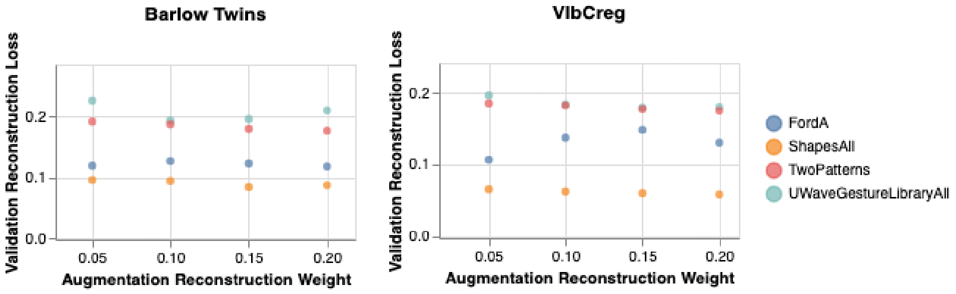

Figure 4: Augmentation: Slice and shuffle. Averaged across 2 runs. Trained for 250 epochs