Segmentation Transformer Model

# Main Body Description

Transformers are a machine learning architecture known primarily for their success in the field of natural language processing. At a high level, they achieve this success in part because of their ability to model long-range relationships between features within a sequence, like how the words in a sentence depend not just on the words in the same sentence, but on those in the rest of a body of text as well. Transformers have also been useful in segmentation, particularly in the image domain. We hypothesize that the task of EPG probe segmentation would also benefit from being able to model long-range dependencies as probe signals can be long and distinction between waveform types often requires comparison to other types within the same probe across large distances. Therefore, here we adapt the progressive upsampling variant of the Segmentation Transformer designed for image segmentation as described in Zheng et. al 2020 for use in our time-series domain. The primary difference in our implementation is that we replace their two-dimensional patches which are a given fraction of the total image size with one-dimensional, fixed-width windows and our use of a one-dimensional positional embedding as opposed to two-dimensional. We then follow the same process of encoding using a transformer and decoding using progressive upscaling. For more details on our model architecture, please see the supplementary methods.

# Model

## Architecture

This model begins by splitting each probe into windows of length *w*. Next, we apply a linear layer with GELU activation to each window with an output dimension of *embed\_dim* and add a sinusoidal positional embedding. Our embeddings are then encoded using pytorch’s TransformerEncoder with *t\_layer* layers and *t\_head* heads. Next we apply a linear layer with GELU activation to each of the encoded windows to bring them back to the same dimension as our number of classes. Finally, we use progressive upsampling in which we apply a convolution with width 2 to the series, instance norm, repeat the convolution, and then do a linear upsample to double the length of the series until we reach the length of our original probe.

In our experiments, we found *w* = , *embed\_dim* = , *t\_layer* = , and *t\_head* = to be the best parameters for maximizing the F1 score.

## Training

The model was trained with Adam with CrossEntropy loss, using an experimentally determined learning rate of for epochs.

# References

Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, Li Zhang: “Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers”, 2020; [http://arxiv.org/abs/2012.15840 arXiv:2012.15840].