Text

Description automatically generated

**DREAM Challenge 2022  
Predicting gene expression using millions of random promoter sequences  
by  
BioNML**

***Abstract***

*The relationship between DNA sequences and gene expressions is vital in understanding the complex nature of how regulatory activities are encoded in all living organisms. Investigations of this relationship are particularly challenging due to limited availability of (sequence) samples and measurements. An alternative approach of using random promoter sequences and their associated gene expressions thus provides opportunities for having sufficient sequence variability along with measured expression datapoints and therefore providing a unique platform for understanding this problem. With data generated via this platform, approaches like neural networks (convolutional and/or transformer-based) have been reported to be useful in modeling gene expression from DNA sequences and many are with astonishingly good performances. In this Dream Challenge competition, we approach the same problem via building upon nowadays a popular transformer architecture called vision transformer (ViT). Specifically, we exploit the tokenized internal representations of the ViT models as alternative feature maps that are used to make final predictions, which deviates from many published practices of using direct transformer encoded outputs. In addition, we explore and adopt several innovative modifications from deep learning literature that are generally helpful in scaling up and increasing modeling capabilities for the problem.*

**1. Description of data usage**

All training sequences are one-hot encoded and padded or trimmed to 110 bp w.r.t the center of the non-constant oligo sequence part. During training, the entire data set is first randomly stratified (via binning the expression values to their closest integer) into 65%/10.5%/24.5% brackets for training/validation/testing. The training and validation sets are used to warm-up the model in the initial stage of the training, followed by a fine-tuning stage where a re-sampled validation set (~16.25% of the entire dataset) from the original training bracket is used and we set all the remaining data points to be used for training in the fine-tuning stage. We check for overfitting via implementing early stopping rules in both stages and by observing a weighted metric formed with the square root of positive coefficient of determination (COD) and Pearson’s correlation coefficient.

**2. Description of the model**

The main foundation of this proposed model is based on nowadays the popular vision transformer (ViT) model. We supplement the one-hot encoded input sequences with transcription factor motif patterns (to be learned) and k-mer patterns (freeze initially, later allow to learn during fine-tuning). Both work like DNA related information encoders here and are having the sequence information encoded at the very beginning of the networks and then allow the encoded feature stream to be further compressed, positionally embedded, and padded with extra learnable representation token for downstream transformer layers. Deviated from models like DeepTransformer, only the representation tokens are used for further information encoding and for final predictions.

On top of utilizing the DNA related information encoders and ViT-based transformer architecture, the proposed model is supplemented with several additional modeling features to help boosting the performance. They are:

* Swish activation and SwiGLU layer (in transformer’s feedforward part).
* Inception-like module for exploring different intermediate signal passing through routes and calculations.
* Suppressed cross-attention query signal.

**3. Training procedure**

The model is trained in two stages.

In the first stage, the model is warmed up using train/valid splits (~75.5% of the entire dataset). We freeze weights for modeling k-mers at this stage and train the network by monitoring a 40/60 weighted metric formed of the square root of positive coefficient of determination (COD) and Pearson’s correlation coefficient and wait till early stopping is hit (patient is set to 3 epochs) or out of training epochs. A mean squared error is used as loss function at this stage and with a relatively large learning rate of 0.001.

In the second stage of the training, the best weight from warm-up period is loaded, and we let k-mer weights to be learnable. We re-sample 16.25% of the entire dataset as validation set from previous training bracket and set all the remaining data points into training bracket. We allow learning schedule to be implemented in this stage where the learning rate decreases piecewisely. A Huber loss function is used to replace the MSE loss function in this stage with a delta value of 0.85 so to put relatively small focus on outliers. We let the training process to hit early stopping (patient is set to 10 epochs) or completing 50 epochs to stop the entire fine-tuning process.

In both stages, we set the optimizer to be Adamax, and we have some l2 regularizations at the penultimate layer and along with dropout layers at several places of the network. For the entire training process, we allow samples to be weighted differently w.r.t their frequency but with a weight limit of 2.0 (starting from 1.0) to adjust slightly for unbalanced expression bins.

**4. Other important features**

**5. Contributions and Acknowledgement  
5.1 Contributions**

|  |  |  |
| --- | --- | --- |
| **Name** | **Affiliation** | **Email** |
| Xiaoting Chen | CCHMC | Xiaoting.Chen@cchmc.org |
|  |  |  |

**5.2 Acknowledgement**We thank Dr. Matthew Weirauch from CCHMC for extensive discussion over the scope of this project and supporting of the team for participating this Dream Challenges competition.

**6. References**

Dosovitskiy, A., Beyer, L., Kolesnikov, A., *et al*. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. https://arxiv.org/abs/2010.11929

Vaishnav, E.D., de Boer, C.G., Molinet, J. *et al.* The evolution, evolvability and engineering of gene regulatory DNA. *Nature* **603,**455–463 (2022). https://doi.org/10.1038/s41586-022-04506-6

**7. Feedback (optional)**