Predict missing single cell markers using the transformer encoder based neural network

Subchallenge 1 writeup

Team: Ostar
Xueer Chen (team leader), Department of Biomedical Informatics, University of Pittsburgh
Jiwei Liu, AI infrastructure, NVIDIA

Will you be able to make your submission public as part of the challenge archive? Yes

Summary

We designed a transformer encoder neural network that jointly learns the sequential pattern of all the missing single cell markers by utilizing the self-attention mechanism.

Introduction

In subchallenge 1, our goal is to predict the missing markers in specific time points after perturbation. We make two key observations:

1. The levels of the missing markers correlate with the levels of other measured known markers in the same experiment as defined by treatment, cell line, cell ID and file ID. Consequently, known markers could be strong features.
2. There exists a clear sequential pattern of the levels of markers along the time axis for the same experiment. In general, the difference between the levels of makers at two time points is small if they are close in time, and large otherwise. There also exists a long term trend where the level of the marker could go up first, then drop and become flat.

Based on these two observations, we believe a neural network that learns sequential pattern could be a promising approach. Specifically, the transformer[1] is the state of the art model that outperforms traditional methods such as recurrent neural networks. It leverages the self attention mechanism to learn 1) the weights of known markers with respect to the missing marker and the weights of all time points with respect to the current time point.

Our final solution is a single model of such a design without any ensemble. We believe there is great potential to further improve the prediction accuracy using this approach.

Methods

The task is formulated into a multi-target regression problem and our model jointly predict the missing missing markers of all six cell lines in one shot. Our approach includes the following steps:

1. Design a validation strategy. A robust validation strategy is a crucial step for our approach since it lays the foundation for iterative experimentation of ideas. It is quite

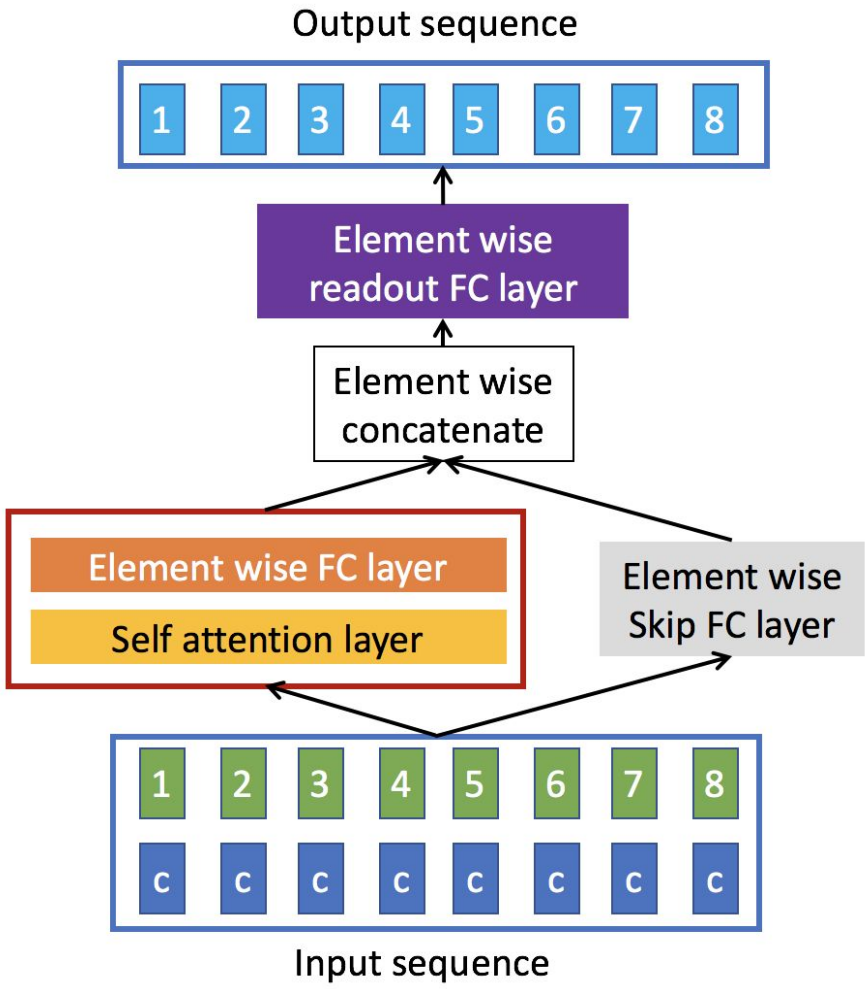
challenging to find a proper validation dataset since the missing markers are not provided at all for the six cell lines at any time points. Our key idea is to find similar cell lines to the six test cell lines and similar markers to the missing markers. The detailed split up of validation data and test data is shown in the table below. We selected 6 cell lines similar to the test cell lines and 3 markers similar to the missing markers, which are shaded pink. The blue shaded cells represent the combinations of cell lines and markers for the validation data. The green shaded cells represent the combinations of cell lines and markers for test data as given. All the rest cell lines and markers are used as training data.

	p.GSK3b	p.MAPKAPK2	p.BTK	p.Akt.Ser473	p.ERK	p.HER2	p.PLCg2	p.S6
MPE600								
BT474								
HCC2185								
MCF7								
184A1								
BT549								
AU565								
EFM19								
HCC2218								
LY2								
MACLS2								
MDAMB436								

2. Data Preprocessing and Feature engineering. The features of our model include two groups:
    a. Context features: context features are the common features for the entire sequence shared by all time points. We used two context features: treatment and cell line. They are label encoded to map to consecutive integers and later on represented by embeddings in the neural network model.
    b. Sequential features: sequential features are specific to each time point. We used 30 sequential features most of which are known maker levels.

    We transform the dataframe so that each row is a sorted sequence of experiments to facilitate batch generation for training the neural network.

3. Model design and training. The general model architecture and training mechanism are shown in the figure below. The input sequential and context features are represented in green and dark blue blocks, assuming 8 timestamps. The context features are copied into each time point. The concatenated input sequence is then fed into two modules in parallel. On the left branch, a classic transformer encoder module with the self attention layer computes a representation of the sequence considering the sequential nature of input. It utilizes an attention mechanism to relate all time points of the input sequence with respect to the current time point. On the right branch, a simple element wise fully connected layer is employed to prevent overfitting of the weak or noisy sequential pattern for some target markers. The outputs of two branches are concatenated again for read out. The final output sequence is simply all the missing markers at each time point.



To reproduce the results, please run the notebooks in the following order:

1. subchallenge_1_prepare_miao.ipynb

2. subchallenge_1_prepare_rnn_miao.ipynb
3. subchallenge_1_fastai_transformer.ipynb

Conclusion

In this challenge, we present a novel model design using transformer encoder based neural network to predict missing markers jointly. The model is capable of learning sequential nature of the input patterns. We make the following observation regarding the design of models:

1. Neural network provides great flexibility jointly learn multiple targets. It not only reduces training time but also improves prediction accuracy by learning the latent interaction of multiple targets.
2. The transformer encoder architecture is much faster than recurrent neural network and it is also much easier to overfit. Therefore, the introduction of the skip fully connected neural network is necessary to prevent such overfitting. We believe the sequential pattern exists in most cases but it can be weak and noisy for some samples.
3. We also found that the information of patients is not very useful regarding predicting the missing markers.
4. We found rapids.ai cudf library delivers amazing speedup in preprocessing the data. Pytorch provides convenient APIs for transformer APIs.

References

[1] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." In *Advances in neural information processing systems*, pp. 5998-6008. 2017.

Authors Statement

Xueer Chen leads the team in terms of understanding the datasets, designing the overall approach and interpreting the results. She also developed data driven methods to find similar cell lines and markers.
Jiwe Liu implemented the algorithm using rapids.ai and pytorch.