馬拉松運動博覽會參訪動線類別預測

Pattern Recognition Final Project

Group: Team18 Team member: 林蔭澤, 吳書磊, 羅宇辰, 涂皓鈞

Introduction

Analyzing the traffic flow of the people helps us understand the preferences and stay of visitors, thus making good use of the limited space for those people with specific demands. The dataset for the issue was collected from 2018 Marathon Expo, and the visitors had been organized into five types. In this project, we implement several models to perform the classification tasks, including Transformer + LSTM, Fully-Connected Neural Network (FC-NN), and XGBoost. The model architectures of the first two will be introduced in the following sections.

Framework

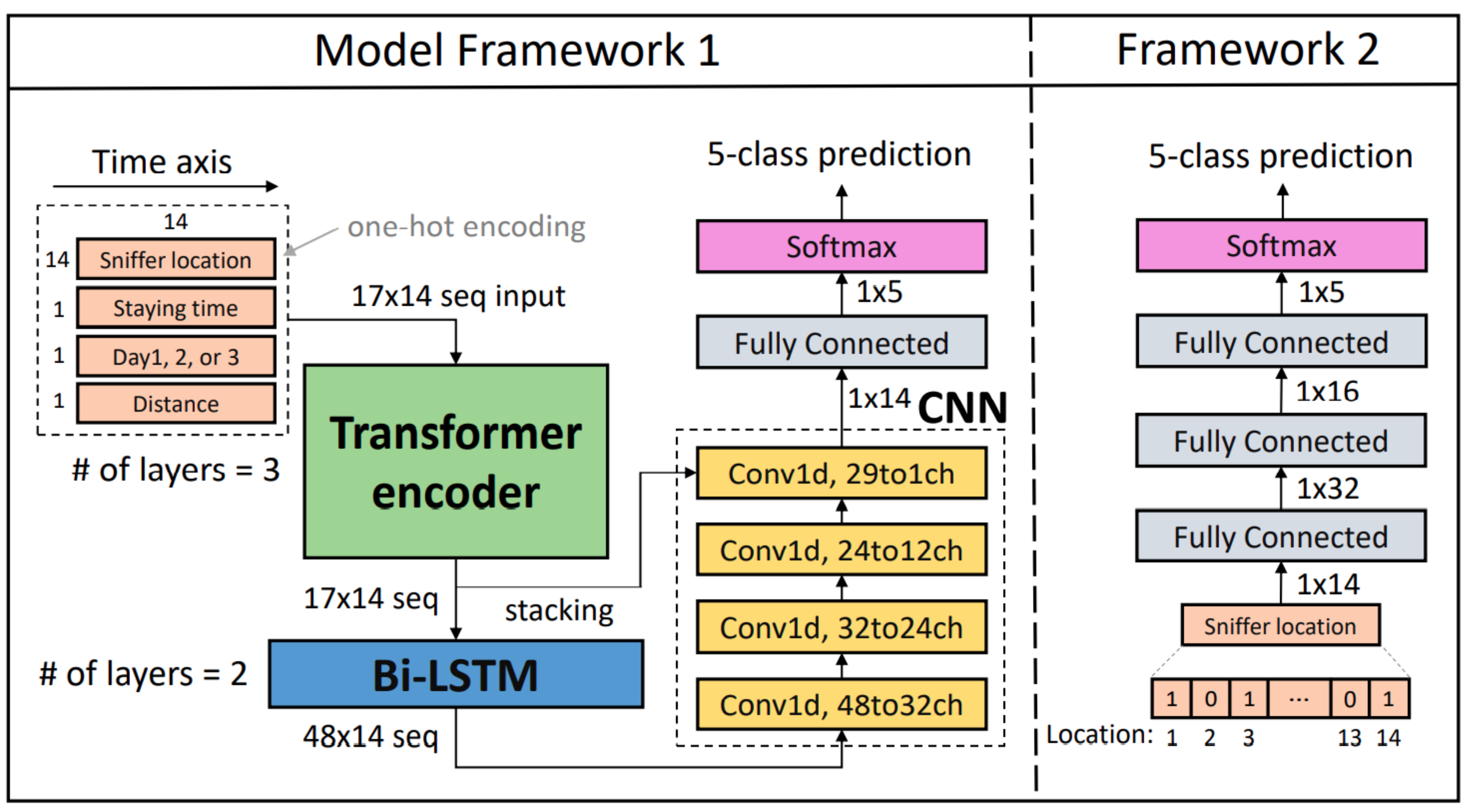

Fig. 1: The proposed model frameworks.
(Left: Transformer + LSTM; Right: Fully-Connected Neural Network)

- Framework 1 (time-dependent model)

1. Input Features:

Input features include sniffer locations, staying time and distance between each location and which day the visitors participated the activity. The above features are in chronological order.

2. Model Architecture:

Fig. 1 on the left shows the model architecture of the proposed Transformer + LSTM. Transformer Encoder enables the model learn the range of the receptive field automatically according to the different length of the input. The bidirectional LSTM can further help us find the relationship on the time axis among the sequence. CNN is mainly used to extract more high level feature and reduce dimension.

- Framework 2 (time-independent model)

1. Input Feature:

Input feature records if sniffer locations had been visited.

2. Model Architecture:

Fig. 1 on the right shows the model architecture of the proposed FC-NN. Only three fully-connected layers are in the model to prevent overfitting, and the activation function after fully-connected layer is PReLU.

Results

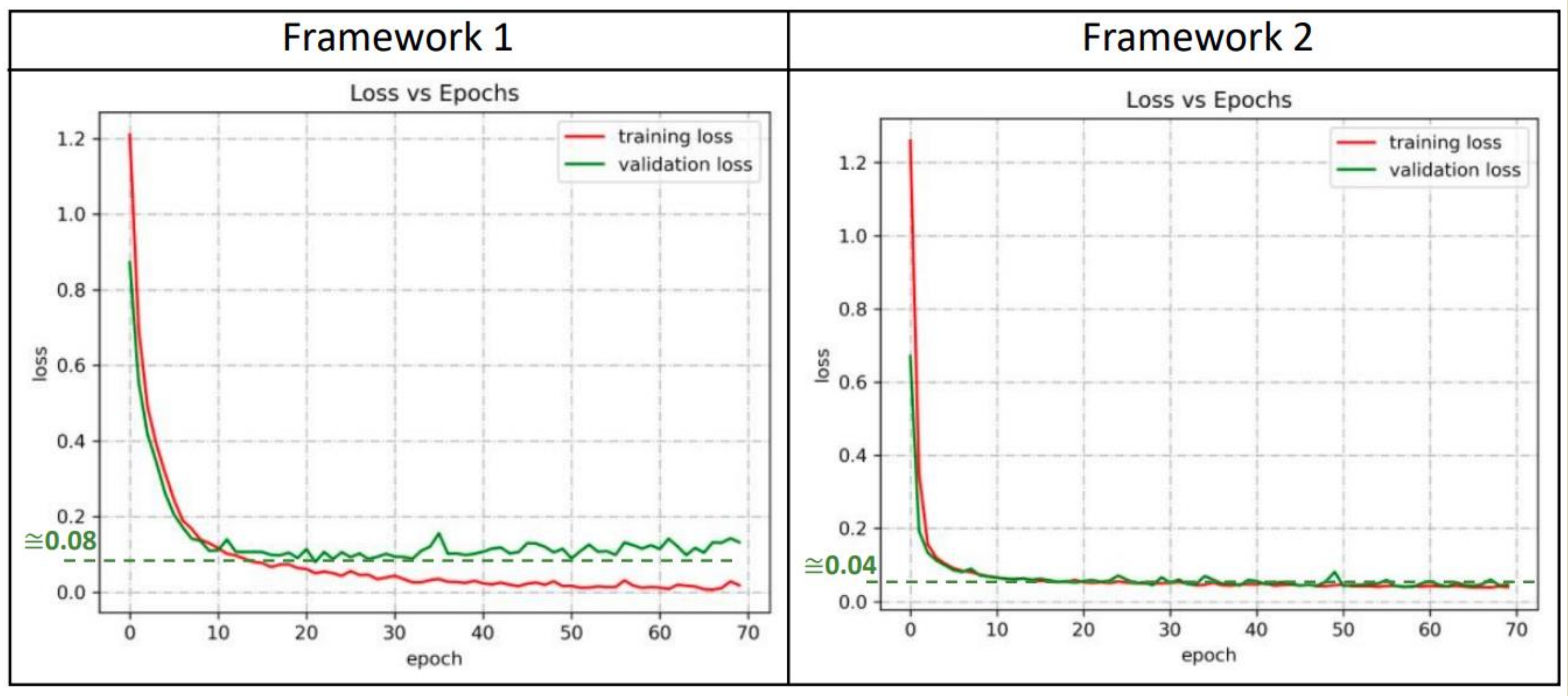

Fig. 2: loss vs. epochs among two different frameworks.
(Left: Transformer + LSTM; Right: Fully-Connected Neural Network)

	Testing Loss
Transformer + LSTM (framework 1: v1)	0.09455
Transformer + GRU (framework 1: v2)	0.07988
FC-NN (framework 2)	**0.04719**
XGBoost	0.08385

Table 1: Testing loss on AIdea.

Fig. 2 shows the training/validation loss among different number of epoch. We use 10-fold cross-validation to evaluate the model performance. We can further utilize the figure to train the model without validation set. Table 1 records our final results on AIdea. We obtain the best testing loss: 0.04719 through FC-NN framework. There are mainly two reasons for this situation. First, the model structure of framework1 might be too complex. Second, the feature in the dataset might be less time-related. The model of FC-NN is time-independent, and it can learn better within the adequate model depth, so its performance is better than the traditional machine learning method – XGBoost.

Summary

In this classification tasks, basically, we consider if the dataset is time-related to the predictive results, so we mainly implement two frameworks: Transformer + LSTM and FC-NN. From the results, we can conclude that deep model could degrade the performance given the limited data, and the feature in the dataset might be less time-related. Finally, we obtain the optimal testing loss from two frameworks: 0.07988 and 0.04719, which framework 2 - FC-NN hit the best performance as our final result.