Project Topic - Predicting social media engagement

https://www.kaggle.com/c/mlb-player-digital-engagement-forecasting/data?select=train.csv

1. Introduction

As MLB is heavily data driven, it is a great opportunity to apply machine learning to yield insights into what constitutes digital popularity. More information on this challenge can be found in MLB Player Digital Engagement Forecasting on Kaggle (Google Cloud/ Major League Baseball, 2021). The purpose of this challenge is to generate insights into what factors influence online activity to predict four measures of fan engagement using baseball player digital content.

2. Data Exploration

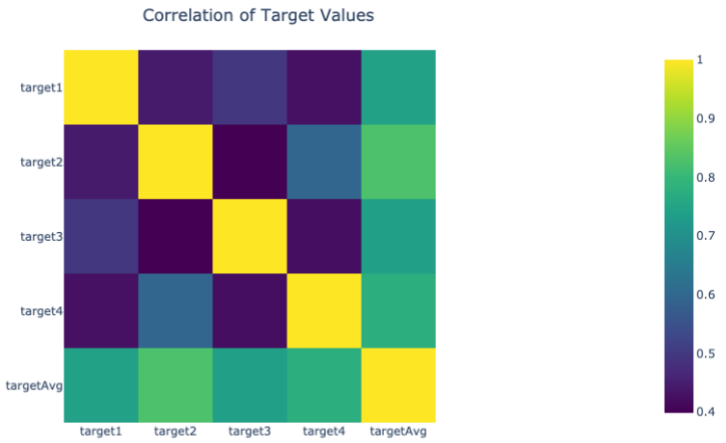
Data Pre-processing

Files	Description
train.csv	A training set containing data on MLB players active at some point since 2018 in nested JSON fields. Predictions are only scored for those players active in 2021, but previous seasons' players are included to provide more data exploration and modelling purposes. The nested JSON fields are: nextDayPlayerEngagement, games, rosters, playerBoxScores, teamBoxScores, transactions, standings, awards, events, playerTwitterFollowers, teamTwitterFollowers
awards.csv	A collection of awards given out prior to 01/01/2018 (the first date in train.csv)
players.csv	A library containing high level information about all MLB players in this dataset
seasons.csv	Information about start and end dates of all seasons in this dataset
teams.csv	A library containing high level information about all MLB teams

(Descriptions taken from MLB Player Digital Engagement Forecasting from Kaggle)

Prior to any data exploration, I unpacked the nested JSON files in train.csv and merged all the fields into a master table. A noteworthy field is *nextDayPlayerEngagement* as it contains *engagementMetricsDate, playerId, target1, target2, target3* and *target4*. The aim is to predict the target predictions for each *playerId* on given engagement metric dates. The dataset has a total of 2,506,176 observations and 72 features.

Targets



As the four targets of engagement are not explicitly specified, I shall first explore how each target variable varies from one another. Figure 1.a in the appendix shows significant seasonality, with all target values peaking in March and dipping in October. As MLB is generally scheduled from early April to early October, this observation implies that fan engagement is affected by on and off season periods. By analysing the Pearson correlation in Figure 1, I can see that the target variables have a medium positive correlation with each other.

3. Evaluation Metric

The model shall be evaluated on the mean column-wise mean absolute error (MCMAE).

$$MCMAE = \frac{\sum_{i=1}^{4}{\sum_{j=1}^{n}|y_{ij} - \hat y_{ij}|}}{4n}$$

where $\hat y_ij$ is the prediction for the i^th target and y_ij is the true value and n is the number of observations.

4. Implementation of Model

Auto Regressive Neural Network

The multilayer perceptron neural network is a supervised learning technique where information only travels forward in the network. The inputs to each node are combined using a weighted linear combination and the result is then modified by a nonlinear function before being outputted to another node. These weights are optimized using a backpropagation algorithm to minimize a loss function. An Autoregressive Neural Network (ARNN) builds upon this by-passing lagged values of the time series data into a network and using it to predict future values. I shall be using this model to predict social media engagement scores.

<u>Pre-processing</u>

As the data provided is temporal, it brings into play issues that can skew predictions such as seasonality (non-stationary data). To counter this, the first difference was taken for each target, effectively detrending the data (appendix fig 5.a before, 5.c after). I then proceeded to create 3 data sets to test 3 variations of the (ARNN):

- ARNN with 5 lags of each target as features,
- ARNN with 20 lags as features
- ARNN with 5 lags and additional player box score metrics as features.

To motivate the selection of 5 lags, a plot (see Figure 9.a appendix) was constructed showing the autocorrelation distribution for player target averages across different lag values. Here, lag means are reasonably high until lag 5 but then reduces significantly afterwards. 20 lags were selected as reference of comparison. Player box scores such as *hits, strikeOuts, homeRuns,* and *strikeOutsPitching*, have all shown to be strongly correlated with engagement. This correlation is expected as better player performance will usually result in more commentary and digital engagement surrounding that performance. We shall use the ARNN model to further explore these features' predictive power. After constructing each data set, each was split into train and tests, with the use of time series split to avoid data leakage issues.

Construction of the Neural Network's
All three models consist of an input layer containing the nodes that represent the different features for each model, 3 hidden layers and an output layer representing the 4 targets. Regularization in the form of dropout layers and Batch normalization was used to counter overfitting, with the ReLU activation function and MAE for loss. Each model was implemented using Kares, a high level framework based on tensorflow and was trained using K-fold cross validation.

Hyper parameter tuning – Tuning of parameters is necessary in order to find the optimal model. This was done in a three stage process where we tuned the number of nodes in hidden layers, the learning rate and dropout rate of the network. Each hyperparameter was tuned in a method similar to GridSearchCV, where we optimised for the loss of the network. It was found that the best the combination of hyper parameters were 128 hidden nodes , with a learning rate of __ and a drop out rate of ___

Criticisms
Although the set of ARNN's produced comparable results, one of the biggest problems is that there is no quantitative way to measure one feature as having a greater predictive power then another. A neural network is inherently Blackbox in nature. We therefore cannot know what relationships or parameters led to a conclusion, or why the network exactly came to such a conclusion.

5. Results

Autoregressive Neural network Results:

ARNN K-fold Cross Validation Results			
Model	**Lags**	**Folds**	**MCMAE Score**
Autoregressive Neural network	5	10	1.6043152928352356
Autoregressive Neural network	20	10	1.6995096802711487
Autoregressive Neural network with Player box features	5	10	3.3845490169525147

To obtain the results, we used k-fold cross-validation to train the models and found the final MCMAE score through predicting and calculating the loss on a held out test set.

7. Conclusions

In this project, I did extensive data exploration and analysis to determine a feature set which is meaningful to the prediction of next day digital engagement. Given the richness of the dataset deep learning model (ARNN) performed well. From the findings, factors which are most important to forecasting baseball fan engagement includes the lagged values of the targets, position name, the number of times a player appears in the dataset and the number of player and team Twitter followers.

For future work, I could further explore the effects of encoding, particularly for our other models. An area which I could analyse is the effectiveness of categorising events as in or off season and how that influences our results.

I could also experiment with a Long Short Term Memory Recurrent Neural Network (RNN). The main advantage of RNN over artificial neutral networks is that RNN can model sequence of data, like time series, so that each sample can be assumed to be dependent on previous ones.

Additionally, collecting more data, in particular, relating to popular trends or viral videos from Youtube or Tiktok, the player's influence