Report Charity-funding-predictor

By Christian Tourteau

The non-profit Alphabet Soup Foundation seeks to predict the best charity funding. Data containing over 34,000 organizations previously funded by Alphabet Soup and their metadata is preprocessed and fed into a deep learning neural network.

2 strategies were used:

* a first in which the hyperparameters of the neural network are predefined and adjusted to the data (accuracy of 73%) and
* a second optimized in which the hyperparameters were systematically tested used the keras tuner (accuracy of 77%)

Result

Data Processing

* EIN and NAME features were dropped from the model.
* The model's target variable is IS\_SUCCESSFUL. APPLICATION data was analyzed and CLASSIFICATION value was used for grouping
* Each unique value used several data point as a cutoff point to bin categorical variables together in a new value. Categorical variables were encoded by “pd.get\_dummies() ”.

**Compiling, Training, and Evaluating the Model**

Graphical user interface, text, application, email

Description automatically generated

* As can be seen in the table below, 537 parameters were generated and the model accuracy was achieved at 73%

Table

Description automatically generated

Text

Description automatically generated

**Optimization**

* With hyperparameter optimization, the model was able to perform at 77%, which is 2 basis points better than 75%.

Table

Description automatically generated

Text

Description automatically generated

**Summary**

Thanks to the results obtained, the neural model demonstrated decent reliability in predicting the success of charitable organizations. In this sense, Alphabet Soup obtains additional information about charities to fund in the future. You can test other optimizations to be carried out in the future (different pre-processing, additional data, etc.); additional tests (model variance test to check for overfitting) to check the power of the model.

I would suggest applying a PCA and/or tSNE model during the pre-processing step to reduce the number of features**.**