Applied Artificial Intelligence
CS-514

Project 5 – Predict Future Sales Challenge (Kaggle Dataset)

Machine Learning

Sherlock

TOPIC	PAGE
Project Abstract	3
Software Requirement	3
Setup Instructions	3
Models Used	3
Results	4
References	5

Project Abstract:

Given the historical dataset, the task in hand is to try forecasting the total amount of products sold in a set of shops. The dataset is provided by a Russian Software Firm - 1C Company. Thus, applying machine learning techniques, we try to predict the sales of the product shop-wise by training our machine learning classifiers with the data provided and test them for accuracy.

Software Requirement:

- Python (version 3.5 or higher)
- Other Dependencies – Numpy, pandas, tensorflow, sklearn and lightgbm etc.

Setup Instructions:

- Download the dataset from https://www.kaggle.com/c/competitive-data-science-predict-future-sales/data
- Extract all the files in the same folder as the python files.
- Make sure all the dependencies are resolved. (If exceptions are thrown when trying to install dependencies try running as administrator)
- Run the Sherlock1.py file and wait for few seconds for the data to process to observe the test file being generated and the metrics being displayed for two models.
- Run the Sherlock2.py file and wait until the all the 10 epochs are done and observe the metrics.

Models Used:

LINEAR REGRESSION MODEL – To understand the simple supervised approach we use a basic classification technique – a linear regression model to train the classifier which assumes that there is a linear relationship between the input attributes and the class.

Light GBM MODEL – This is a grading boosting network that employs tree-based algorithms for classification and learning. Light GBM is a new algorithm and has gained attention due to it's lightening speed execution and lower memory consumption.

LSTM - Long short-term memory networks are a specialized type of recurrent neural network (RNN)—a neural network architecture generally used for the modeling sequential data which come in handy with our scenario. These tend retain information for long periods of time, allowing for important information learned early in the sequence to have a larger impact on model decisions made at the end of the sequence which reflects on the better accuracy on the model.

Results:

Basic data cleaning methods were adopted before training the classifier. Tried filling out the missing values, removing duplicate values, concatenating the data to produce meaningful structure to process the data.

Choosing Mean Squared Error as the metric to understand the performance of the models,

Linear regression Classifier

RMSE - 0.8928421754192852   MSE: 0.7971671502074418

Light GBM Classifier

RMSE: 0.7995511699065118   MSE: 0.6392820732988717

LSTM Classifier

Epoch 1/10

214200/214200 [==============================] - 64s 298us/step - loss: 1.3718 - MSE: 1.3718

Epoch 2/10

214200/214200 [==============================] - 60s 278us/step - loss: 1.3650 - MSE: 1.3650

Epoch 3/10

214200/214200 [==============================] - 59s 274us/step - loss: 1.3615 - MSE: 1.3615

Epoch 4/10

214200/214200 [==============================] - 62s 288us/step - loss: 1.3582 - MSE: 1.3582

Epoch 5/10

214200/214200 [==============================] - 72s 335us/step - loss: 1.3518 - MSE: 1.3518

Epoch 6/10

214200/214200 [==============================] - 60s 280us/step - loss: 1.3537 - MSE: 1.3537

Epoch 7/10

214200/214200 [==============================] - 60s 282us/step - loss: 1.3391 - MSE: 1.3391

Epoch 8/10

214200/214200 [==============================] - 56s 259us/step - loss: 1.3205 - MSE: 1.3205

Epoch 9/10

214200/214200 [==============================] - 60s 280us/step - loss: 1.3213 - MSE: 1.3213

Epoch 10/10

214200/214200 [==============================] - 60s 280us/step - loss: 1.3056 - MSE: 1.3056

References:

https://medium.com/simple-ai/linear-regression-intro-to-machine-learning-6-6e320dbdaf06

https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc

https://www.oreilly.com/ideas/introduction-to-lstms-with-tensorflow

https://www.kaggle.com/mjbahmani/tutorial-on-ensemble-learning/

https://medium.com/coinmonks/text-classifier-with-keras-tensorflow-using-recurrent-neural-networks-ad63dd5fc316