Background

In a market economy, goods and services have prices because resources are limited, and the price of a given good or service depends on demand and supply, in other words, price is the visualization of demand-supply relationship model, these are the two basic principles of economics. The factors that can influence demand and supply are many, ranging from government policies to individual buying and selling behavior. These information can be summarized as an expression of sentiment toward a good or service. Intuitionally, people using the information they can gather to analyze sentiment in order to predict prices. And now, with the rapid growth of the Internet, we can get news and people's opinions from social media. This makes gathering information more quickly. However, we need the help of an algorithm to quickly analysis the sentiment on social media to determine the likely price trend. That's why we chose this topic, which is to help predict cryptocurrency prices by analyzing sentiment in real time.

Introduction

In order to analyze sentiment to predict cryptocurrency prices, we divided the pipeline of the project into four main parts: data streaming, sentiment analysis, prediction model and front-end presentation. Our final goal is to provide an online web page to present our model results, including predicted prices and price trend graphs. Next, my teammates and I will go through the pipeline to present each part separately and demonstrate our project.

Slide 1

In this part, I will introduce the pipeline of the prediction model, including the model we used, optimizations, prediction function and predicted results. After stream processing and sentimental analysis, the pipeline of the project input two processed dataset into here, and the objective of the prediction model is that to find and conclude the potential relationship between price and sentimental value, even it is obscure. For the prediction model, the input is pre-processed cryptocurrency price data and sentimental values, and the output is the predicted price and price trend graph.

Slide 2

I will introduce how we construct the prediction model. The right side graph shows the pipeline of the function that construct the model. First, we need to do some data cleaning for the two input dataset. We need to merge them into one dataset and normalize to ensure it has fixed change range. After that, split the dataset into training set with 80% and test set with 20%. Then, we introduced Grid search, which test all given hyperparameters to calculate the Root Mean Square Error value iteratively, and provide the hyperparameter set which has the best performance. Finally we input the hyperparameter set and training data into the model that we will construct.

We use LSTM-GRU hyper model to predict the price. The model is mainly contain a 3 LSTM layers and 1 GRU layer. The reason that we choose LSTM as the model is that the crypto prices represent time dependencies, meaning that past prices influence future prices. LSTM are naturally suited for handling time series data. LSTM memory cell has input, forget, and output gates that control the flow of information in, retention, and out of the cell, which makes the model can capture the relationship between past information and current input. Also, crypto prices can be affected by long-term trends and sudden events. LSTMs can capture these long-term dependencies, improving the accuracy of the predictions.

Slide 3

In training model, the optimizations are essential for the model accuracy and training time. Here I list several ways in which we optimize our model, we significantly increase the generalize ability and robustness of the model to quickly capture the features of the dataset and make predictions.

Grid Search:

First is grid search. We train several hyperparameters in the grid model and find the best ones. The grid search is able to determine the accuracy of the model by determining the Root Mean Square Error for each hyperparameter combination.

AMSGrad:

Second is AMSGrad, it is a optimizer in neural network. AMSGrad retains the maximum of all past second moment estimates to prevent sudden drops in learning rate, resulting in better optimization performance. Also, it uses a correction factor to ensure the gradient average is not underestimated during the iteration, it in order to get the better convergence performance. In general, AMSGrad has the best performance and robustness in the result from grid search.

Dropout Layer:

Third is dropout layer. By adding dropout layers after each LSTM layer of the model to randomly discard the output of a certain percentage of neurons to ensure that the model does not rely excessively on certain neurons. In this model, we set each LSTM layer to have 30% of neurons result will discarded. We found that the addition of the dropout layer significantly increases the generalization ability of the model and prevents overfitting.

GRU Layer:

The last one is GRU model. GRU has simpler structure than LSTM, which GRU has only two gates, the update and reset. The different structure makes GRU has faster training time and can capture different features than LSTM.

Slide 4

After trained model, we need to use the model to predict the future price of the crypto currency. The prediction function is a separate function from the training model, which can directly use trained model to predict. In the actual user experience, the user usually expect a quick response from the website, so we let prediction function to join the project pipeline but not the full function containing the training model. The figure on the right hand side is the pipeline of the project which zoom-in on the prediction function. The prediction function load model from the local storage and fetch data set from the previous stage of the pipeline, then output the predicted price and price trend graph to the front-end.

Slide 5

The two graph shows the test and prediction result of our model on recent trade days. The left side graph is the test set result, and the right side is the prediction for the future price.

Project Deployment

We successfully deployed the project on website. We create a crontab to run the training model function daily, including fetch the crypto currency price and sentimental analysis. When user clicks the currency button, the prediction function will run to predict the price and trend graph then plot them on the front end. Next, we will demonstrate our project on the local machine.

Conclusion

In general, we create a full pipeline of the project. We use spark and optimizations to streaming the data of crypto currency and tweet data. We use lexicon on sentimental analysis to get the highest accuracy. We use LSTM-GRU hyper model to discover the features of the relationship between two datasets, and predict the future price and price trend. Finally, we plot the prediction result on the front-end, and deploy the full project on-line and provide a full-automatically process of fetch data, analysis, train model and predict the price.