**Summary of paper**

The paper focuses on stock market prediction using a diverse set of variables using CNN(Convolutional Neural Networks). Deep learning offers a reliable way of prediction since it is a suitable technique for automatic feature extraction. Deep learning is used in various fields like Computer Vision, Natural Language Processing among others. Stock markets show a non-linear behavior and DL seems to offer prediction by extracting the market’s features.

Previously, SVM(Support Vector Machines), Random Forest method, Logistic Regression, etc were used to predict the stock market’s behavior. But they were not as effective as DL techniques, like MultiLayer Perceptron, LSTM etc which were able to extract complex features from the raw data. It is very important that all the features like raw price data, technical indicators obtained from historical data, connection between other markets, etc are aggregated in the prediction algorithm. Therefore, to automatically extract those features and keeping in mind about the complex feature space, CNN poses to be a promising approach for the prediction task.

A CNN model was previously employed but in that model, they took 1D input that took just one feature - closing prices, therefore it failed to perform good prediction since it neglected the correlation of various features. The CNN framework in this paper successfully aggregates various sources of information for prediction. This framework takes a 3D tensor as input to align those vast sets of variables and then trains the network so that it extracts the valuable features for prediction of various stock markets. Feedforward ANN(Artificial Neural Networks), which are used to find optimal initial or final weights are also discussed in the paper. When CNN, LST and MultiLayer perceptron models were applied on the S & P 500 stock market index, CNN outperformed both models.

Various layers of CNN, for e.g. , input layer, convolutional layer, pooling layer, fully connected layer and output layer were discussed in brief. ReLu(Rectified Linear Unit) is used as a non-linear activation function for output of each filter before it passes through the next layer. Also, apart from pooling, another method called dropout is used for the model to restrict the model from learning too much from the training data and thus make the model well-generalized.

In CNNpred, the proposed framework, 2 variations - 2D CNNpred & 3D CNNpred were tried. 2D CNNpred approach is to find a general model for mapping the history of a market to its future fluctuations, a model that is valid for several markets. In 2D CNNpred, basically information from various sources are aggregated and then fed as a 2D tensor as input. 3D CNNpred takes a different approach in the sense that it aggregates information from various sources, feeds a 3D tensor as input, and trains a separate prediction model for each market.

For 2D CNNpred, the size of the matrix depends on the number of variables that represent each day, as well as the number of days back into the history that is

used for making a prediction. Then primary daily features are extracted, decided by the no. of initial variables taken. After that, following layers combine extracted features of different days to construct higher-level features for aggregating the available information in certain durations. For the final prediction, features generated by the last pooling layer are flattened into a final feature vector and sigmoid is chosen as the activation function, to discretize the output to either 0 or 1, representing probability of value going down or up.

For 3D CNNpred, the initial daily variables, the days of the historical record and the markets from which the data is collected, form the three dimensions of the input tensor.For durational feature extraction, the height of filters is selected to be 3 for covering three consecutive time units. Like 2D CNNpred, output of durational feature extraction is flattened to produce the final prediction.

For the CNNpred models, 82 variables were used for representing each day of each index. Variable set included different groups like :

* Primitive variables : closing price and day of week for prediction
* Technical indicators : extracted from historical data of stock prices
* World stock markets : interaction between different stock markets over the globe
* Exchange rates, big companies and commodities

Datasets were obtained from daily direction of the close of S&P 500 index, NASDAQ Composite, Dow Jones Industrial Average, NYSE Composite, and RUSSELL 2000. Each sample has 82 variables and labels have been assigned according to the criterion that if closing price of next day > closing price of preceding day, then value of target is 1, else 0.

Evaluation metric for these models was decided to be Macro-Averaged F-measure which is the mean of F-measures. Keras was used to implement CNN in CNNpred prediction framework. CNNpred is profitable in 4 out of 5 tested indices in presence of transaction costs. 3D CNNpred had better average F-score over other algorithms like PCA. CNNpred was able to improve the performance of prediction in all the five indices over the baseline algorithms by about 3% to 11%, in terms of F-measure.