Project Proposal

University of Rochester
CSC 240/440 Data Mining Fall 2016
Daily News Sequence Classification
for Stock Market Prediction
with LSTM Recurrent Neural Network
Team Member: Yue Wang, Hizkia Febianto

Problem Statement

The Efficient Market Hypothesis (EMH) states that stock market prices are largely driven by new information. In this project, we are looking to find out the impact of daily news on stock market behavior. Positive news, such as good earnings reports, increased corporate governance, new products and acquisitions, as well as positive overall economic and political indicators, usually translates into buying pressure and an increase in stock price. On the other hand, negative news, such as economic and political uncertainty, and unexpected, unfortunate occurrences, may result in stock price to fluctuate negatively. The possible impacts are predefined as three major classes: negative, positive and neutral fluctuation. One of our project goal is to classify the potential market price reaction based on daily news.

There are three variants of the EMH hypothesis: "weak", "semi-strong", and "strong" form. The weak form of the EMH claims that prices on traded assets already reflect all past publicly available information. The semi-strong form of the EMH claims both that prices reflect all publicly available information and that prices instantly change to reflect new public information. The strong form of the EMH additionally claims that prices instantly reflect even hidden "insider" information. We are going to approach this problem based upon above hypotheses: whether news report at current time stamp has an impact on future stock prices, or news is instantly reflected to the change of current stock prices or current stock price has finished its adjustment before news go public.

Proposed Solution

Word embeddings is going to be used to convert words into multi-dimensional vectors, a numerical representations of word features. It turns text into a form that deep neural nets can understand. On top of multidimensional word features, we are fitting a LSTM recurrent neural net to take the sequence of each word feature into account. The output of our neural net are the classes indicating the range and direction of stock prices movement. In addition, we are going to

shift the time frame of our model input forward and backward to test on different hypothesis, and compare their prediction accuracy.

Dropout may be applied to the input and recurrent connections of the memory units of our model to avoid overfitting problem. We may also limit the total number of words that we are interested in modeling to the most frequent words(eg. top 5000), and zero out the rest. Besides, the sequence length (number of words) in each daily news headline varies, so we will constrain each day's news to a limited number of words, truncating long news headlines and pad the shorter news with zero values. So that our model input will have the same dimensionality, the number of word features times the number of words used.

Another possible application of LSTM in our project is to predict stock price using the Window Method. The LSTM network has memory which is capable of remembering across long sequences. We are approaching the problem as the time series prediction problem. To predict future stock prices based on historical prices.

DataSet

In this project, we used two main datasets:

1. Dow Jones Industrial Average (DJIA) values from June 2008 to July 2016. The data was obtained using Yahoo! Finance and includes the open, close, high and low values for a given day.
2. Historical News Headlines from Reddit WorldNews Channel from June 2008 to July 2016. They are ranked by reddit users' votes, and only the top 25 headlines are considered for a single date.

Algorithms

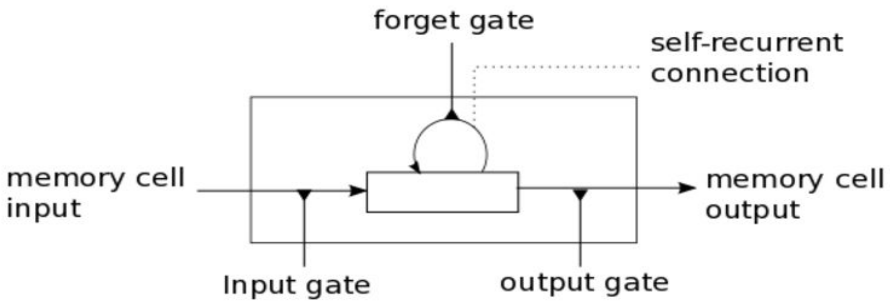
LSTM

In a traditional recurrent neural network, during the gradient back-propagation phase, the gradient signal can end up being multiplied a large number of times (as many as the number of timesteps) by the weight matrix associated with the connections between the neurons of the recurrent hidden layer. This means that, the magnitude of weights in the transition matrix can have a strong impact on the learning process.

If the weights in this matrix are small (or, more formally, if the leading eigenvalue of the weight matrix is smaller than 1.0), it can lead to a situation called vanishing gradients where the gradient signal gets so small that learning either becomes very slow or stops working altogether. It can also make more difficult the task of learning long-term dependencies in the data.

Conversely, if the weights in this matrix are large (or, again, more formally, if the leading eigenvalue of the weight matrix is larger than 1.0), it can lead to a situation where the gradient signal is so large that it can cause learning to diverge. This is often referred to as exploding gradients.

These issues are the main motivation behind the LSTM model which introduces a new structure called a memory cell. A memory cell is composed of four main elements: an input gate, a neuron with a self-recurrent connection (a connection to itself), a forget gate and an output gate. The self-recurrent connection has a weight of 1.0 and ensures that, barring any outside interference, the state of a memory cell can remain constant from one timestep to another. The gates serve to modulate the interactions between the memory cell itself and its environment. The input gate can allow incoming signal to alter the state of the memory cell or block it. On the other hand, the output gate can allow the state of the memory cell to have an effect on other neurons or prevent it. Finally, the forget gate can modulate the memory cell's self-recurrent connection, allowing the cell to remember or forget its previous state, as needed.



Word Embedding

Vector space models (VSMs) represent (embed) words in a continuous vector space where semantically similar words are mapped to nearby points ('are embedded nearby each other'). VSMs have a long, rich history in NLP, but all methods depend in some way or another on the Distributional Hypothesis, which states that words that appear in the same contexts share semantic meaning. The different approaches that leverage this principle can be divided into two categories: count-based methods (e.g. Latent Semantic Analysis), and predictive methods (e.g. neural probabilistic language models).

This distinction is elaborated in much more detail by Baroni et al., but in a nutshell: Count-based methods compute the statistics of how often some word co-occurs with its neighbor words in a large text corpus, and then map these count-statistics down to a small, dense vector for each word. Predictive models directly try to predict a word from its neighbors in terms of learned small, dense embedding vectors (considered parameters of the model).

Reference

[1] Anshul Mittal, Arpit Goel, Stock Prediction Using Twitter Sentiment Analysis. IEEE Computer, 44(10):91–94.

[2] Zhiang Hu, Jian Jiao, Jialu Zhu Using Tweets to Predict the Stock Market.

[3] Bollen, J., Mao, H. and Zeng, X.-J. 2010. Twitter mood predicts the stock market. Journal of Computational Science 2(1):1–8.

[4] Andrej Karpathy blog The Unreasonable Effectiveness of Recurrent Neural Networks.