RGBSticks : A New Deep Learning Based Framework for Stock Market Analysis and Prediction

Eren Unlu
Datategy SAS
Paris, France
eren.unlu@datategy.net

Abstract—We present a novel intuitive graphical representation for daily stock prices, which we refer as RGBSticks, a variation of classical candle sticks. This representation allows the usage of complex deep learning based techniques, such as deep convolutional autoencoders and deep convolutional generative adversarial networks to produce insightful visualizations for market’s past and future states.

Legal and Ethical Notice : This work is conducted completely for scientific purposes. It shall not be considered as an investment strategy or financial advice.

I. INTRODUCTION

Candlestick charts have been by far the most widespread way of visualizing the daily stock market and exchangeable currency, derivative or commodity values for a long time. According to many sources, they can be traced back to Japan’s Meiji period in 18th century, developed for rice trading. Some academics particularly credit it to Munehisa Homma, a prominent rice broker of the era [1] [2].

The very fact that, they have been used uninterruptedly and unchanged throughout the history of modern trading hints us about the effectiveness of the method. Even with today’s complex cutting edge digital infrastructure and automated trading algorithms; they are still among the most valuable elements of a broker’s toolbox; and this is likely to last for years. Possibly, it’s effectiveness comes from its simplicity, where the overall image of many consecutive days (or any other intended time period) can abstract many hidden and latent factors of market dynamics inside a broker’s mind. The human brain’s preference for visual data to rapidly process complex tasks subconsciously is a well studied phenomenon [3] [4].

Based on this observation, we propose to process the candlestick charts as an image, in contrast to many deep learning and data science based stock price analysis and forecasting techniques, which approaches the issue as raw tabular or temporal feature extraction [5] [6] [7] [8].

The proposed method’s advantage appears to be twofold. Firstly, it may allow the deep learning algorithms to capture the complex patterns that raw data processing cannot provide; an insight stemming from the success of candlesticks that driving human traders to make correct decisions in a highly chaotic environment throughout three centuries. Second, as the results provided by the deep learning methods are also candlestick like presentations; the human traders can gain insights on the past and the future market and conclude with interpretable artificial intelligence support.

After we have come up with the idea, firstly we have reviewed the literature to check whether a similar approach was followed before. To the best of our knowledge, [9] is the only one to mention the encoding of stock price movements as candlestick images, however it still does not propose a pure visual representation like the method explained in this paper. The authors use Gramian Angular Fields to encode the stock market timeseries and apply deep Convolutional Neural Networks (CNN).

Candlestick approach is highly straightforward : The value of the traded entity (currency, commodity, contract etc.) in a time period (a day, a business week, 5 minutes etc.) are represented with four values. Open and Close are the first and last are the prices at the beginning and end of the period, respectively. If the close value is lower than the open value, it is considered as a bearish period and the inner candle of the candlestick is colored red. The inverse case corresponds to a bullish period and represented with a green candle. As the name suggests, the high and low are the highest and lowest prices in the period, and drawn as thin, stick like vertical lines on top and bottom of the inner candle. The large gap between these values indicates a high volatility in that particular period.

In this paper, we present RGBSticks, a novel framework to transform candlestick market prices to a structure, where it is still both readable for human agents and also in the form which benefits the deep learning methods. The name comes from the encoding of open, close, low and high prices on red, green and blue channels of a digital image. After explaining this visual representation, we analyze the daily stock price of a company by using a dense deep neural network based autoencoder. By using the same architecture the next day is tried to be predicted. Finally, a Deep Convolutional Generative Adversarial Network (DC-GAN) architecture is applied to simulate artificial market periods.
out of these ranges its values are clipped. The image size of
In case, for the relatively rarer cases where the next days is
place RGBSticks in the vertical axis for a specific timeblock.

of the minimum value of previous 16 days. Thanks to this
larger than the 110% of the maximum and smaller than 90%
maximum and minimum of all 4 indicators. Without loss of
representation a day. For each 16 days, we first calculate the

want our machine learning algorithms to forecast the next day
and look after values are determined as 16 and 1, meaning we
instances in future we want to predict. In this paper, look back

data point length is referred as look after, the number of
look back

parameter, that being
Temporal deep Learning algorithms, whether they process
image (e.g. video frame prediction) or not require the supervi-
sion of the sequential data in to a tabular structure, where
the lagged instances are grouped together. This procedure is
sometimes referred as timeseries supervision or lagged/sliding
window feature generation. The maximum length of each
lagging window is called the look back parameter, that being
also the expression used in this paper. Similarly, the output
data point length is referred as look after, the number of
instances in future we want to predict. In this paper, look back
and look after values are determined as 16 and 1, meaning we
want our machine learning algorithms to forecast the next day
based on previous 16 days.

Fig. 1. A traditional candlestick representation and an RGBStick representa-
tion for a bullish and bearish period. A candlestick transforms the open,
close, low, high values into 5 semantic visual features: high, low, maximum
and minimum of the open and close price and whether the close value was
higher than the open or not, where the last is represented with red or green
color.

II. RGBStick

An RGBStick is visually almost identical to a conventional
candlestick as it can be seen Fig. 1, with the exception that
the outer sticks have the same width with the inner candle
and encoded in blue channel. This allows the deep learning based
computer vision algorithms to process and output human read-
able three dimensional (width, height indices and 3 channel
color) images. Thanks to this representation, we can proliferate
the performance of deep learning algorithms which uses this
kind of input shape, forcing the synaptic and concolutional
weights to be tuned to plausible and distinctive ranges by
assigning the information to highest values of a single channel.

Temporal deep Learning algorithms, whether they process

image (e.g. video frame prediction) or not require the supervi-
sion of the sequential data in to a tabular structure, where
the lagged instances are grouped together. This procedure is
sometimes referred as timeseries supervision or lagged/sliding
window feature generation. The maximum length of each
lagging window is called the look back parameter, that being
also the expression used in this paper. Similarly, the output
data point length is referred as look after, the number of
instances in future we want to predict. In this paper, look back
and look after values are determined as 16 and 1, meaning we
want our machine learning algorithms to forecast the next day
based on previous 16 days.

So, each input data point in our system is a square image, as
in Fig. 2.). And the output data point is a rectangular image
representing a day. For each 16 days, we first calculate the
maximum and minimum of all 4 indicators. Without loss of
generality, we assume that the values of next day will not be
larger than the 110% of the maximum and smaller than 90%
of the minimum value of previous 16 days. Thanks to this
defined global maximum and minimum, we can normalize and
place RGBSticks in the vertical axis for a specific timeblock.
In case, for the relatively rarer cases where the next days is
out of these ranges its values are clipped. The image size of
64x64 pixels are determined for this paper, thus the width and
height of a single RGBStick is normalized according to.

As you can see, the candlesticks are placed on a total
black image plane, where all the 3 channel values has 0
intensity. As the information on being bullish or bearish and
the volatility (the outer limits of a RGBStick) are encoded as
the highest intensity of all independent 3 channels; this permits
the machine learning algorithms to capture and exploit the
most important features of the data by enforcing optimization
process to iterate through the extrema.

III. Deep Neural Network Forecasting of Future
Market

Based on the explained framework, as a first attempt, we
have built a regular dense deep neural network to predict
the next days based on the previous 16 days. An hourglass
shaped, autoencoder like architecture is preferred to reduce
dimensionality on latent feature space to mitigate the highly
volatile nature of stock market data. The details of the used
neural network can be inspected in Fig. 3.

We present the results of our work for a single company’s
daily stock market prices for a calendar year. The first 90% of
the days are used as training, whilst the remaining last days
of the years are evaluated for testing purpose.

As it can be seen from the figures, our RGBStick backed
deep learning framework not only provides accurate results,
it also permits the human trader to develop a market strategy
by evaluating the results based on the color shades, similar to
typical candlestick charts, where he/she already is familiar.

IV. Deep Autoencoders for Understanding
Dynamics of Market

We have used the above mentioned architecture as a deep
autoencoder, with the intentions to extract information out of
the history of the stock prices. As the architecture was chosen
Fig. 3. The hourglass shaped deep dense neural network used for predicting the next days’ RGBsticks based on previous 16 days. The relu layers are used in all layers, except the last layer, where logistic regression probability is preferred for matching the output.

Fig. 4. Real RGBStick and the predicted one with our dense deep neural network for a bearish and bullish day in the test dataset. As it can be noted, the algorithm performs quite efficiently to predict the tendency and volatility. A human interpreter can have an insight on the volatility and the tendency of the forecast by evaluating the difference hue variations. Intensity of the blue on the outer edges signal the limits and confidence of the volatility. Similarly, stronger reddish or greenish hue give an idea on the upper and lower limits of close and open values. For instance yellowish hue would mean the mixture of green and blue; where open/close values are close to each other for a bullish day. In contrast, violet hue pixels can be interpreted for the same effect for a bearish day. Brownish pixels can be interpreted as the confidence on bearish/bullish decision is lower as mixture of red and green.

Fig. 5. The predicted RGBSticks days of the test part. This kind of concatenated view permits the human trader to evaluate the forecasts and the short to mid term tendency in the market in a global fashion with the parameters such as confidence, volatility and extrema are encoded in color shades.

Fig. 6. The days in the past of the stock prices with real and the predicted RGBSticks using deep autoencoder. The autoencoder allows us to interpret that on these specific two days open/close values would actually meant to be more closer to each other based on the overall dynamics of the environment.

V. MARKET ANALYSIS AND SIMULATION WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS

Generative Adversarial Networks (GAN) are relatively new architectures in deep learning research, being started to be developed few years ago [10] [11] [12]. They are one of the most innovative breakthroughs in the artificial intelligence. With this state-of-the-art models we are now capable of generating highly sophisticated artificial data, such as deepfakes or artificial human faces or transfer styles between batch of images. The central idea is to have two separate neural networks connected to each other called a generative network and a discriminator network. In the case for convolutional networks for images, the generator network takes randomized inputs for its latent space and map it to the images of the intended size through deconvolutional layers. As for the variational autoencoders, GAN also learns the distribution via latent space rather than the direct processing of input data. The discriminator’s task is to be fed with real training input images and the artificial outputs of the generative network, and classify it as a real one or fake one. Throughout training, generative network learns gradually to produce more real like artificial outputs, whilst discriminative network gets better to discriminate the fake ones. Thanks to this adversarial concept, at then end we are able to produce realistic fake images. Even though it is an adversarial setting, the nature is cooperative, where at the end generator network is able to generate realistic fake images. The DC-GAN architecture is highly efficient...
The deep generator has grasped the sharp upside and generated by the DC-GAN architecture in Fig. 8. Note the stock price history of the company and 24 fake timeblocks images. The details of the used architecture can be seen in Fig. network with the daily stock prices represented as RGBStick images to produce plausible results [12].

When you consider the fact that the latent vector is fed with random inputs, adversarial setting and deep convolutional and dense layers is capable of being trained in short time to produce plausible results [12].

We have trained a deep convolutional generative adversarial network with the daily stock prices represented as RGBStick images. The details of the used architecture can be seen in Fig. 7. We provide 32 arbitrary real RGBStick timeblocks of the stock price history of the company and 24 fake timeblocks generated by the DC-GAN architecture in Fig. 8. Note the fact that, the deep generator has grasped the sharp upside and downside trends and high volatility in the data.

VI. CONCLUSION

Candlestick charts have been used extensively for a long time for the analysis of trading markets. The success of this kind of a chart comes from its effective visual abstraction for human traders, who can have a wide angle view of the history. This has encouraged us to theorize that there might be a powerful latent information encoded visually. Thus, we propose to represent open, close, high and low prices as candlestick like graphical representation, which we refer as RGBSticks. The outer sticks defining the difference between the high and low values, i.e. the limits of the volatility in the time period is encoded as full intensity in the blue channel (thus, zero intensity in the other 2 color channels). The bearish and bullish inner candles are represented as full intensity in the green and red channels. This encoding of important information on the extrema of color channels helps deep learning algorithms to reach more optimal weights for intended tasks. We have tested a dense neural network to predict the RGBStick of the next day based on previous 16 days. The same architecture is used also to evaluate the history of the stock prices by autoencoding. Finally, a DC-GAN architecture is applied to simulate and understand the stock prices of a company.

We believe RGBStick representation has great potential to integrate human decision process and deep learning for stock market analysis and forecasting. The traders who are highly familiar with candlesticks are able to evaluate the results generated by deep learning algorithms by inspecting the varying color shades in a compact, instinctual and rapid fashion.

REFERENCES

Fig. 8. 32 arbitrary real RGBStick timeblocks of the stock price history of the company and 24 fake timeblocks generated by the DC-GAN architecture.