TIME SERIES CLASSIFICATION WITH CONVOLUTIONAL NEURAL NETWORK

ANTHONY VAN DER WAL | UWA DATA ANALYTICS

PROJECT 3 | 27 FEB 2021

MOTIVATION

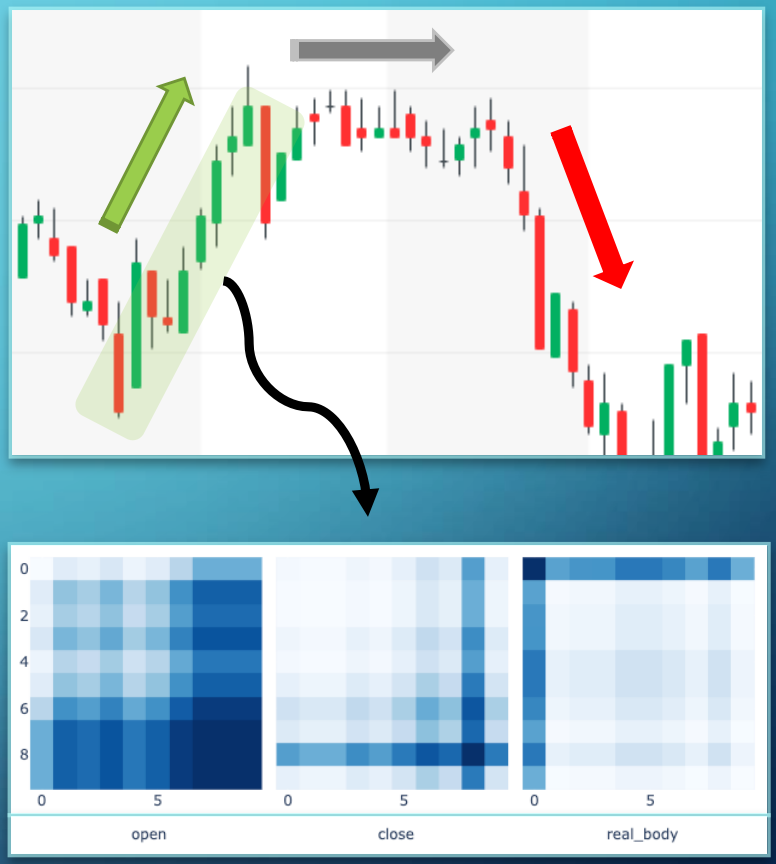
"… there seems to be something non-algorithmic about our conscious thinking. In particular, … that, at least in mathematics, conscious contemplation can sometimes enable one to ascertain the truth of a statement in a way that no algorithm could."

— **Roger Penrose, The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics**

- In the early 1990's I attended a lecture by Roger Penrose. He argued that human intelligence cannot be replicated by a Turing Machine. I want to improve my understanding of what currently qualifies as 'artificial intelligence'.
- Gain a functional understanding of image and time series classification tools.

OUTLINE

- Objective: classify simple patterns in time series data.
- Method:
  - Time series from currency exchange prices.
  - 1D time series labelled (pos., neg., neut.)
  - Transform into 2D images.
  - Train labelled images with Convolutional Neural Network (CNN).
- Result: model is used to classify (predict) unlabelled time series.



SOURCES

- EUR-USD hourly exchange rates.
- Gramian angular summation field:
  - Wang & Oates (2015) -- https://arxiv.org/abs/1506.00327
- Training:
  - Kaggle -- https://www.kaggle.com/imetomi/eur-usd-forex-pair-historical-data-2002-2019
- Prediction
  - Yahoo Finance -- https://pypi.org/project/yfinance/

FUNCTIONAL UNDERSTANDING

- Split data consecutively – images are created from groups of 10 consecutive datapoints.
- GASF – 2D images that retain the angular relationships of a 1D series.
- CNN – computationally efficient. Accurate. Modelled after mammal eyesight.
- Conv2D – extract features from images. Filters are trained on coarse shapes at first – subsequent convolutional layers will look for finer features. Each filter identifies a feature. Output 'convolved' feature maps. Feature maps are smaller than the input image.
- MaxPooling – partition convolved feature maps. Retain only the maximum value of each partition. Pooled feature maps are smaller than the convolved feature maps. The max pooling retains only the prominent features.
- Flatten – transform the 2D images into a 1D series easily consumed by a neural network.
- Dense (fully connected) – create additional attributes of the feature maps. Run them forward/backward to adjust the weights.
- Dropout – randomly set feature values to 0. Helps reduce overfitting by preventing adjacent nodes being 'influenced' by each-other.
- https://www.cs.ryerson.ca/~aharley/vis/conv/flat.html
- https://towardsdatascience.com/wtf-is-image-classification-8e78a8235acb

TRAIN & TEST

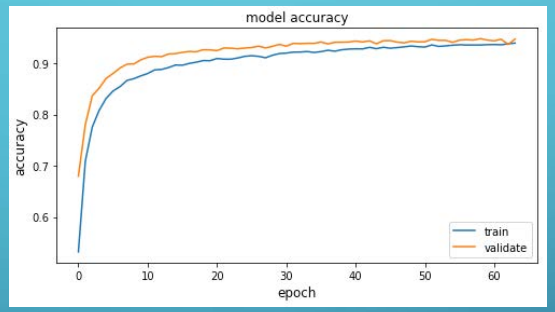
y_train

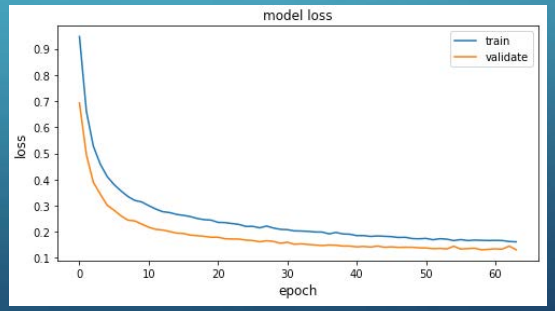
```
['negative', 9868]
['neutral', 17786]
['positive', 9846]
```

balance

```
['negative', 9868]
['neutral', 9868],
['positive', 9844]
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 8, 8, 16)          448
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 4, 4, 16)          0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 2, 2, 32)          4640
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 1, 1, 32)          0
_________________________________________________________________
dropout (Dropout)            (None, 1, 1, 32)          0
_________________________________________________________________
flatten (Flatten)            (None, 32)                0
_________________________________________________________________
dense (Dense)                (None, 64)                2112
_________________________________________________________________
dropout_1 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 32)                2080
_________________________________________________________________
dense_2 (Dense)              (None, 3)                 99
=================================================================
Total params: 9,379
Trainable params: 9,379
Non-trainable params: 0
_________________________________________________________________
```



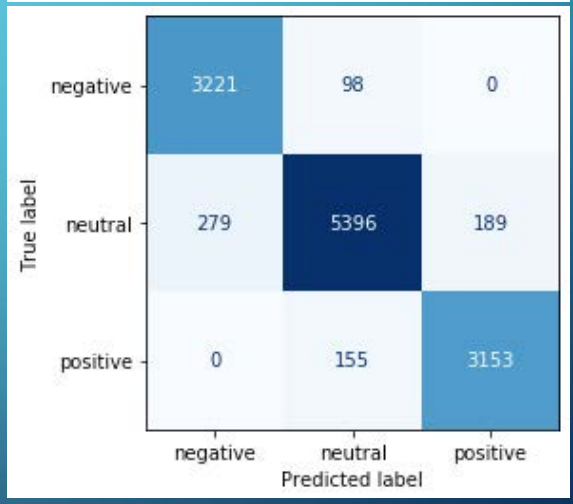


y_test

```
['negative', 3319]
['neutral', 5864],
['positive', 3308]
```

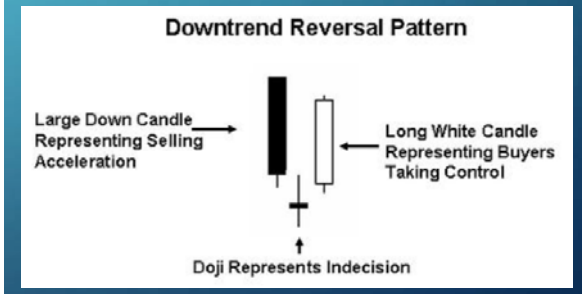
```
Test accuracy 0.94227844
Test loss 0.144216942658093
```

	precision	recall	f1-score	support
negative	0.92	0.97	0.94	3319
neutral	0.96	0.92	0.94	5864
positive	0.94	0.95	0.95	3308
accuracy			0.94	12491
macro avg	0.94	0.95	0.94	12491
weighted avg	0.94	0.94	0.94	12491



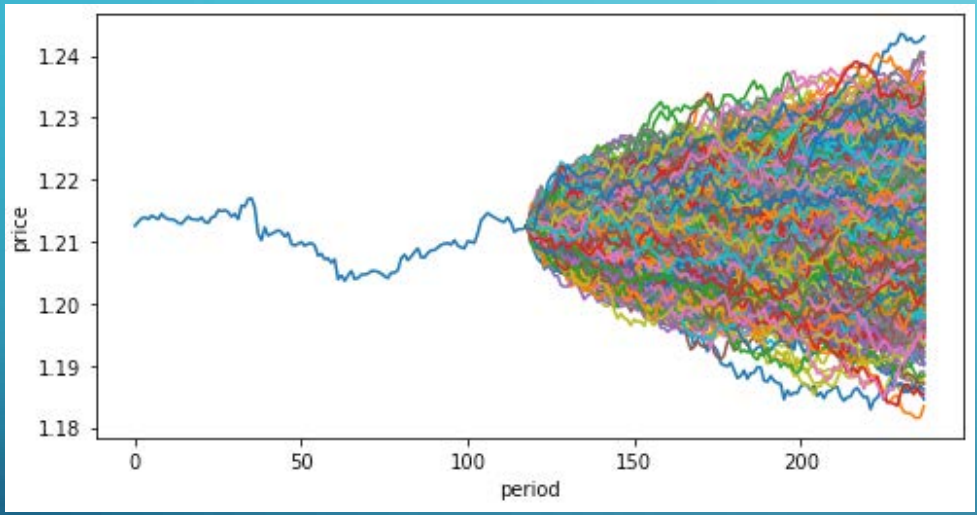
PREDICT

- Positive/negative slopes often occur at inflection points.
- Potential use → as first step in identifying Candlestick patterns.



https://en.wikipedia.org/wiki/Morning_star_(candlestick_pattern)

JUST FOR FUN


price
period
1.24
1.23
1.22
1.21
1.20
1.19
1.18
0
50
100
150
200


FORECAST PRICE BANDS - EURUSD
Price Count
Percentage
Price Band [USD]
1%
3%
8%
17%
23%
24%
15%
7%
3%
1%
1.183-1.189
1.189-1.195
1.195-1.201
1.201-1.207
1.207-1.213
1.213-1.219
1.219-1.225
1.225-1.231
1.231-1.237
1.237-1.243

DISCUSSION

- CNNs for simple tasks are easily created on a small laptop computer.
- Human brains have ~86 billion neurons.
- CNN in this project has ~150 x 3 filters (neurons?).
- In my opinion Penrose is (still) correct.