Forecasting the Stock Market Using LSTMs

Aurélien Géron

April 30th, 2018
AI Conference, New-York
wheelchair basketball: 0.829
basketball: 0.114
streetball: 0.020

Large-scale Video Classification with Convolutional Neural Networks, CVPR 2014
https://youtu.be/qrzQ_AB1DZk
• Lookahead Factor Models
• Introduction to Recurrent Neural Networks
• LSTMs for Forecasting Company Fundamentals
• Handling Cross Validation & Hyperparameter Tuning
A Random Walk?
Random Walk? > 3 fake, 1 true
Random Walk? Ξ 3 fake, 1 true
- High frequency trading
- Order execution
- News, rumors, product launches
- Financial reports
  - Revenue, income, assets, debt, dividends...
• Intrinsic value?
• Compute *value factors* for each stock
  ○ E.g. EBIT/EV, book-to-market
Random Walk? > Value Investing

- Compute *value factors* for each stock
  - E.g. EBIT/EV, book-to-market
- Create a stock portfolio with highest ranking stocks
Improving Factor-Based Quantitative Investing by Forecasting Company Fundamentals

John Alberg
Euclidean Technologies
john.alberg@euclidean.com

Zachary C. Lipton
Amazon AI
Carnegie Mellon University
zlipton@cmu.edu

Abstract

On a periodic basis, publicly traded companies are required to report fundamentals: financial data such as revenue, operating income, debt, among others. These data points provide some insight into the financial health of a company. Academic research has identified some factors, i.e. computed features of the reported data, that are known through retrospective analysis to outperform the market average. Two popular factors are the book value normalized by market capitalization (book-to-market) and the operating income normalized by the enterprise value (EBIT/EV).
Simulated Annualized Return vs Months of Clairvoyance

- **Book / Market**
- **EBIT / EV**
- **Net-Income / EV**
- **Sales / EV**
Forecasting Fundamentals
Forecasting > Time Series

Revenue vs Time

- t-5
- t-4
- t-3
- t-2
- t-1
- t

Forecasting
Revenue

Imputing

... t-5 t-4 t-3 t-2 t-1 t
Forecasting > Time Series

Imputing

Revenue

... t-5 t-4 t-3 t-2 t-1 t

Time
Forecasting > Time Series

Forecasting

Revenue

Time
Forecasting > Time Series

Revenue

Time

Forecasting
Forecasting > Time Series

Revenue

... t-5 t-4 t-3 t-2 t-1 t t+1 t+2

Forecasting
Revenue

Time

... t-5 t-4 t-3 t-2 t-1 t t+2

Naive Forecasting
Forecasting

ARIMA Forecasting
Forecasting > Time Series

Revenue

... t-5 t-4 t-3 t-2 t-1 t t+2

Forecasting
Forecasting > Time Series

Revenue

... t-5 t-4 t-3 t-2 t-1 t t+2

Forecasting
### Forecasting: Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>9.4</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>13.3</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>12.7</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>13.4</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>8.0</td>
</tr>
</tbody>
</table>
### Forecasting: Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>t-1</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>10.0</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>9.4</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>13.3</td>
<td>9.4</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>12.7</td>
<td>13.3</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>13.4</td>
<td>12.7</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>8.0</td>
<td>13.4</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>12.6</td>
<td>8.0</td>
</tr>
</tbody>
</table>
## Forecasting: Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>t-2</th>
<th>t-1</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>9.4</td>
<td>10.0</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>13.3</td>
<td>9.4</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>12.7</td>
<td>13.3</td>
<td>9.4</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>13.4</td>
<td>12.7</td>
<td>13.3</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>8.0</td>
<td>13.4</td>
<td>12.7</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>12.6</td>
<td>8.0</td>
<td>13.4</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>14.1</td>
<td>12.6</td>
<td>8.0</td>
</tr>
</tbody>
</table>
# Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>t-2 Revenue</th>
<th>t-1 Revenue</th>
<th>t Revenue</th>
<th>t+2 Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>9.4</td>
<td>10.0</td>
<td>13.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>13.3</td>
<td>9.4</td>
<td>10.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>12.7</td>
<td>13.3</td>
<td>9.4</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>13.4</td>
<td>12.7</td>
<td>13.3</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>8.0</td>
<td>13.4</td>
<td>12.7</td>
<td>9.4</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>12.6</td>
<td>8.0</td>
<td>13.4</td>
<td>13.3</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>14.1</td>
<td>12.6</td>
<td>8.0</td>
<td>12.7</td>
</tr>
</tbody>
</table>
## Forecasting > Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>t-2</th>
<th>t-1</th>
<th>t</th>
<th>t+2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>9.4</td>
<td>10.0</td>
<td>13.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>13.3</td>
<td>9.4</td>
<td>10.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>12.7</td>
<td>13.3</td>
<td>9.4</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>13.4</td>
<td>12.7</td>
<td>13.3</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>8.0</td>
<td>13.4</td>
<td>12.7</td>
<td>9.4</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>12.6</td>
<td>8.0</td>
<td>13.4</td>
<td>13.3</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>14.1</td>
<td>12.6</td>
<td>8.0</td>
<td>12.7</td>
</tr>
</tbody>
</table>
## Forecasting: Series to Supervised Learning

<table>
<thead>
<tr>
<th>Date</th>
<th>t-2</th>
<th>t-1</th>
<th>t</th>
<th>t+2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-31</td>
<td>9.4</td>
<td>10.0</td>
<td>13.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-12-31</td>
<td>13.3</td>
<td>9.4</td>
<td>10.0</td>
<td>NaN</td>
</tr>
<tr>
<td>2017-09-30</td>
<td>12.7</td>
<td>13.3</td>
<td>9.4</td>
<td>13.0</td>
</tr>
<tr>
<td>2017-06-30</td>
<td>13.4</td>
<td>12.7</td>
<td>13.3</td>
<td>10.0</td>
</tr>
<tr>
<td>2017-03-31</td>
<td>8.0</td>
<td>13.4</td>
<td>12.7</td>
<td>9.4</td>
</tr>
<tr>
<td>2016-12-31</td>
<td>12.6</td>
<td>8.0</td>
<td>13.4</td>
<td>13.3</td>
</tr>
<tr>
<td>2016-09-30</td>
<td>14.1</td>
<td>12.6</td>
<td>8.0</td>
<td>12.7</td>
</tr>
</tbody>
</table>
Feedforward Neural Network

- **Input layer**: \( x_1 \), \( x_1 \), \( x_3 \)
- **Hidden layer** (e.g., ReLU): \( \Sigma \), \( \Sigma \), \( \Sigma \), \( \Sigma \)
- **Output layer**
Feedforward Neural Network

- Input layer
  - \( x_1 \), \( x_1 \), \( x_3 \)

- Hidden layer (e.g., ReLU)
  - \( \Sigma \), \( \Sigma \), \( \Sigma \), \( \Sigma \)

- Output layer

2017-03-31

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0</td>
<td>13.4</td>
<td>12.7</td>
</tr>
</tbody>
</table>
Feedforward Neural Network

- **Input layer**: $x_1$, $x_1$, $x_3$
- **Hidden layer** (e.g., ReLU)
- **Output layer**: $7.8$

**Forward pass**

- **Date**: 2017-03-31
- **Values**: 8.0, 13.4, 12.7, X
Feedforward Neural Network

Forward pass

Input layer

Hidden layer (e.g., ReLU)

Output layer

\[ \sum \]
\[ \sum \]
\[ \sum \]
\[ \sum \]

Input layer: \( x_1, x_1, x_3 \)

Output layer: \( y \)

\[ 7.8 \]
\[ 9.4 \]
Feedforward Neural Network

- **Input layer**
  - \( x_1 \), \( x_1 \), \( x_3 \)

- **Hidden layer**
  - (e.g., ReLU)

- **Output layer**
  - \( y \) = \( \sum \)

- **Reverse pass**

**Example Input:**

- **2017-03-31**
  - 8.0, 13.4, 12.7
Feedforward Neural Network

- **RNN Basics**
- **Feedforward Neural Network**

![Diagram of a feedforward neural network](image)

- **Input layer**
  - $x_1$, $x_1$, $x_3$

- **Hidden layer**
  - (e.g., ReLU)

- **Output layer**
  - $y = 7.8 + 9.4$

- **Reverse pass**

**Table**

| 2017-03-31 | 8.0 | 13.4 | 12.7 | X |
Feedforward Neural Network

Forward pass

Input layer

Hidden layer (e.g., ReLU)

Output layer

2017-03-31 | 8.0 | 13.4 | 12.7 | X
Recurrent Neural Networks
RNN Basics > Recurrent Neuron

\[ h(t) \]

\[ h(t-1) \]

\[ x(t) \]
$h_{(t-2)} = 6.5$

$h_{(t-3)} = 0.0$
Recurrent Neuron

\[ h_{(t-1)} = 7.8 \]

\[ h_{(t-2)} = 6.5 \]
RNN Basics > Recurrent Neuron

\[ h(t) = 9.3 \]

\[ h(t-1) = 7.8 \]

\[ x(t) \]
RNN Basics > Recurrent Neuron > Unrolled Through Time

\[ h(t-2) = 6.5 \]

\[ h(t-1) \]

\[ h(t) \]

\[ x(t-2) = 7.8 \]

\[ x(t-1) \]

\[ x(t) \]

Time
RNN Basics > Recurrent Neuron > Unrolled Through Time

\[ h_{(t-2)} = 6.5 \quad h_{(t-1)} = 7.8 \quad h_{(t)} \]

\[ x_{(t-2)} = 7.8 \quad x_{(t-1)} = 13.4 \quad x_{(t)} \]

Time
RNN Basics > Recurrent Neuron > Unrolled Through Time

\[ h_{(t-2)} = 6.5 \]
\[ h_{(t-1)} = 7.8 \]
\[ h_{(t)} = 9.3 \]

\[ x_{(t-2)} = 7.8 \]
\[ x_{(t-1)} = 13.4 \]
\[ x_{(t)} = 12.7 \]
RNN Basics > Recurrent Layer > Unrolled Through Time

\[ h^{(0)} \quad h^{(1)} \quad h^{(2)} \]

\[ x^{(0)} \quad x^{(1)} \quad x^{(2)} \]

Time
RNN Basics > Memory Cell

[Diagram of a memory cell with input x, output y, and hidden state h]
RNN Basics > Deep RNN > Unrolled Through Time
RNN Basics > Seq2Seq, Seq2Vec
Encoder-Decoder (Seq2Seq)
Forecasting a Single Time Step

$X_{(6)}$

$W, b$

$X_{(0)}$ $X_{(1)}$ $X_{(2)}$ $X_{(3)}$ $X_{(4)}$
RNN Basics > Forecasting the Full Shifted Series

\[ X^{(2)} \xrightarrow{W,b} X^{(3)} \xrightarrow{W,b} X^{(4)} \xrightarrow{W,b} X^{(5)} \xrightarrow{W,b} X^{(6)} \]

\[ X^{(0)} \xrightarrow{W,b} X^{(1)} \xrightarrow{W,b} X^{(2)} \xrightarrow{W,b} X^{(3)} \xrightarrow{W,b} X^{(4)} \]
RNN Basics > Forecasting the Full Shifted Series

\[ X^{(2)} \]
\[ X^{(3)} \]
\[ X^{(4)} \]
\[ X^{(5)} \]
\[ X^{(6)} \]
Training a Stateless RNN
Fundamentals

Batch 1

Time
Fundamentals

Batch 2

Time
Fundamentals

Batch 3

Time
Fundamentals

Initial state = 0

Batch 3

Time
Fundamentals

Initial state = 0
Stateless RNNs > Problems

Fundamentals

Initial state = 0

Output state

Batch 3

Time
Fundamentals

Initial state = 0
Output state

Batch 3
Time
Training a Stateful RNN
Fundamentals
Fundamentals

Batch 1

Time
Stateful RNNs

Single Time Series

Batch Size = 1

Fundamentals

Batch 2

Time

state
Stateful RNNs > Single Time Series > Batch Size = 1

Fundamentals

Batch 2

Time
Fundamentals

Batch 3

state

Time
Fundamentals

Batch 3

Time
Fundamentals

Batch 3

Time

state

state
Fundamentals

Batch 1, Batch 2, Batch 3, Batch 4, Batch 5, Batch 6, Batch 7, Batch 8

Time

Reset state
Ignored

state
Stateful RNNs > Padding

Fundamentals

Batch 1  Batch 2  Batch 3  Batch 4  Batch 5  Batch 6  Batch 7  Batch 8

Time

Reset state

Ignored

0
Stateful RNNs > Reset State at Each Epoch

Fundamentals

Batch 1  Batch 2  Batch 3  Batch 4  Batch 5  Batch 6  Batch 7  Batch 8
Batch 9  Batch 10 Batch 11 Batch 12 Batch 13 Batch 14 Batch 15 Batch 16

Time

Ignored

Reset state
Stateful RNNs > Instances are Not IID

Fundamentals

Batch 1  Batch 2  Batch 3  Batch 4  Batch 5  Batch 6  Batch 7  Batch 8
Batch 9  Batch 10 Batch 11 Batch 12 Batch 13 Batch 14 Batch 15 Batch 16

Time

Ignored

Reset state
Fundamentals
Fundamentals

Batch 1

Time
Fundamentals

Batch 3
Stateful RNNs > Reset All States at Each Epoch

Fundamentals

Time

Batch 1  Batch 2  Batch 3  Batch 4  Batch 5  Batch 6  Batch 7  Batch 8

Reset states
Stateful RNNs

Reset All States at Each Epoch

Fundamentals

Time

Batch 1 Batch 2 Batch 3 Batch 4 Batch 5 Batch 6 Batch 7 Batch 8
Batch 9 Batch 10 Batch 11 Batch 12 Batch 13 Batch 14 Batch 15 Batch 16

Reset states
Stateful RNNs > Again, Batches Are Not IID

Fundamentals

Batch 1  Batch 2  Batch 3  Batch 4  Batch 5  Batch 6  Batch 7  Batch 8
Batch 9  Batch 10  Batch 11  Batch 12  Batch 13  Batch 14  Batch 15  Batch 16

Time

Reset states
Cross-Validation
Fundamentals

Train

Time
Fundamentals

Train

Validate

Time
Fundamentals

Final Training

Time
Cross-Validation > Split in Time

Fundamentals

Final Training

Test

Time
Fundamentals

Train

Time
Cross-Validation > Split Across Stocks

Fundamentals

Validate

Time
Cross-Validation > Split Across Stocks

Fundamentals

Final Training

Time
Results
Figure 2(b) from the paper
## Results > Figure 2(a) from the paper

<table>
<thead>
<tr>
<th>Strategy</th>
<th>MSE</th>
<th>CAR</th>
<th>Sharpe Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>S&amp;P 500</td>
<td>n/a</td>
<td>4.5%</td>
<td>0.19</td>
</tr>
<tr>
<td>Market Avg.</td>
<td>n/a</td>
<td>7.7%</td>
<td>0.29</td>
</tr>
<tr>
<td>Price-LSTM</td>
<td>n/a</td>
<td>11.3%</td>
<td>0.60</td>
</tr>
<tr>
<td>QFM</td>
<td>0.62</td>
<td>14.4%</td>
<td>0.55</td>
</tr>
<tr>
<td>LFM-Linear</td>
<td>0.53</td>
<td>15.9%</td>
<td>0.63</td>
</tr>
<tr>
<td>LFM-MLP</td>
<td>0.47</td>
<td>17.1%</td>
<td>0.68</td>
</tr>
<tr>
<td>LFM-LSTM</td>
<td>0.47</td>
<td>16.7%</td>
<td>0.67</td>
</tr>
</tbody>
</table>

(a) Out-of-sample performance for the 2000-2016 time period. All factor models use EBIT/EV. QFM uses current EBIT while our proposed LFM-MLPs use predicted EBIT. Price-LSTM is trained to predict relative return.
Thank you!
Bonus Slides
Conv1D Sequence Preprocessing
One-Dimensional Convolutional Layer

\[ \text{ReLU}(w_1 x_1 + w_2 x_2 + w_3 x_3 + b) \]
One-Dimensional Convolutional Layer

\[
\text{ReLU}(w_1 x_2 + w_2 x_3 + w_3 x_4 + b)
\]
Conv1D

ReLU(w₁ x₃ + w₂ x₄ + w₃ x₅ + b)
Conv1D > One-Dimensional Convolutional Layer

ReLU(w_1 x_4 + w_2 x_5 + w_3 x_6 + b)

Conv1D
Conv1D > One-Dimensional Convolutional Layer

$$\text{ReLU}(w_1 x_{13} + w_2 x_{14} + w_3 x_{15} + b)$$
ReLU\((w_1 0 + w_2 x_1 + w_3 x_2 + b)\)
One-Dimensional Convolutional Layer
One-Dimensional Convolutional Layer

Conv1D
One-Dimensional Convolutional Layer

Conv1D
Conv1D > One-Dimensional Convolutional Layer

Time

Conv1D

0

0
One-Dimensional Convolutional Layer
Conv1D > Stride

Conv1D
A Conv1D layer with 3 kernels
Bidirectional RNN
Bidirectional RNNs Architecture

- Concatenate
- Copy
Bidirectional RNNs > Architecture

- Concatenate
- Copy
Bidirectional RNNs > Multiple Layers

Copy

Concatenate
Bidirectional RNNs > Multiple Layers

- Copy
- Concatenate
Phased LSTM
Figure 1 in the paper: https://arxiv.org/abs/1610.09513
Figure 2 (right) in the paper: [https://arxiv.org/abs/1610.09513](https://arxiv.org/abs/1610.09513)
Figure 2 (left) in the paper: https://arxiv.org/abs/1610.09513
Attention
RNNs > Encoder / Decoder for Automatic Translation

Target: Je  bois  du  lait  <eos>
Prediction: Je  bois  le  lait  <eos>

Encoder – Decoder

Softmax

Embedding lookup

0  51  2132  21  431

"<go> Je bois du lait"

Embedding lookup

288  3335  72

"milk drink I"
Figure 1 from the paper: https://arxiv.org/abs/1409.0473
Credits
Many thanks to John Alberg and Zachary C. Lipton for their great paper and for kindly authorizing me to reproduce the figures and results from their paper.

Most images in this presentation are from my book, *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. I also created a few using Matplotlib and Google Slides. The remaining images were released under a creative commons license, or are public domain:

- Clock (slide 8): https://commons.wikimedia.org/wiki/File:Swiss_railway_clock_1.svg