Enhancing Stock Movement Prediction with Adversarial Training

Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun & Tat-Seng Chua
fulifeng93@gmail.com
Stock Prediction

Alphabet Inc. (GOOGL)

Open 1,066.93
High 1,067.00
Low 1,027.03
Close 1,038.74
Volume 4.84M

% Change -11.82%

1,250.00
1,200.00
1,150.00
1,100.00
1,050.00
1,038.74
1,027.03
1,067.00
1,066.93
1,228.00
1,100.00
1,000.00

Apr 7 14 May 7 14 21

6/3/2019

© Co
Why Stock Prediction?

- Which one to buy?
  - All NASDAQ Securities
  - Exchange: NYSE

- When to sell?
  You may not have time, e.g., paper submission deadline.
DNNs, especially RNN [Zhang, KDD’17][Xu, ACL’18] are used to solve asset price prediction as a standard classification problem.

For each stock, on trading day $t$:

Historical prices $\rightarrow$ Stock features

$$
\begin{bmatrix}
1197.5 & 1172.6 & \ldots & 1120.2 & 1105.6 \\
1199.3 & 1179.4 & \ldots & 1126.8 & 1113.4 \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
1173.3 & 1166.5 & \ldots & 1121.4 & 1106.5
\end{bmatrix}
$$

$X^t$

$$
\hat{y}^{t+1}
$$
• Basic Model: *Attentive LSTM*

- LSTM layer captures sequential dependency and projects sequential inputs into hidden representations.
- Temporal attention layer adaptively aggregates hidden representations at different time-steps into \( e^s \).

On a benchmark dataset

Fit the training data 20 months since 2015 Jan.

Two months after the training period.
Stock Prediction with NN
Stochasticity of Stock Price Feature

Historical prices $\rightarrow$ Stock features

\[
\begin{bmatrix}
1197.5 & 1172.6 \\
1199.3 & 1179.4 \\
\vdots & \vdots \\
1173.3 & 1166.5
\end{bmatrix}
\]

\[X^t \rightarrow +\]

Observing the price at a slightly different time (10:00 $\rightarrow$ 10:01).

--- A new feature matrix
--- Different prediction (might wrong)

NN is sensitive to slight feature changes $\rightarrow$ poor generalization ability
• **Basic Model: Attentive LSTM**

- LSTM layer captures sequential dependency and projects sequential inputs into hidden representations.

- Temporal attention layer adaptively aggregates hidden representations at different time-steps into $e^s$.
Stock Prediction with NN
Handling Stochasticity with Adversarial Training

- **Standard training**
  Updates model parameters to fit training data (*clean examples*)

- **Adversarial training**
  Additionally constructs *adversarial examples* via adding small *perturbations* to the input of clean examples, and encourages the model to correctly classify the adversarial examples.

- **Standard training (ideally)**
  Updates model parameters to fit *clean example* as well as all the other points.

- Features observed at a slightly different time.

Adversarial example, the point within the range that is hardest to be predicted as +
• Basic Model: *Attentive LSTM*
  - LSTM layer captures sequential dependency and projects sequential inputs into hidden representations.
  - Temporal attention layer adaptively aggregates hidden representations at different time-steps into $e^s$.

• Adversarial Training
  - Constructs *adversarial examples* via adding *perturbation* to latent representation $e^s$.
  - With an additional loss to encourage correct predictions for the adv. examples.
Stock Prediction with NN

Experiments

**Experiment dataset:** ACL18, a public dataset with 88 high-trade-volume-stocks in NASDAQ and NYSE [Xu, ACL’18].

**Performance comparison:** ALSTM is the basic model; LSTM is ALSTM removing attention; Adv-ALSTM is ALSTM with adv. training; StockNet is the SOTA using VAE.
- Significant improvements.

**Table 1:** Statistics of the dataset.

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Validation</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Examples (+)</td>
<td>10,305</td>
<td>1,139</td>
<td>1,908</td>
</tr>
<tr>
<td>#Examples (-)</td>
<td>10,010</td>
<td>1,416</td>
<td>1,812</td>
</tr>
</tbody>
</table>

**Table 2:** Performance of the compared methods. RI denotes the relative improvement of Adv-ALSTM compared to the associated baseline.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Acc</th>
<th>RI</th>
<th>MCC</th>
<th>RI</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAND</td>
<td>50.89±</td>
<td>12.40%</td>
<td>-0.0023±</td>
<td>—</td>
</tr>
<tr>
<td>LSTM</td>
<td>53.18±5e-1</td>
<td>7.56%</td>
<td>0.0674±5e-3</td>
<td>120.03%</td>
</tr>
<tr>
<td>ALSTM</td>
<td>54.90±7e-1</td>
<td>4.02%</td>
<td>0.1043±7e-3</td>
<td>42.19%</td>
</tr>
<tr>
<td>StockNet</td>
<td>54.96±</td>
<td>4.08%</td>
<td>0.0165±</td>
<td>798.79%</td>
</tr>
<tr>
<td>Adv-ALSTM</td>
<td>57.20±</td>
<td>—</td>
<td>0.1483±</td>
<td>—</td>
</tr>
</tbody>
</table>

**Distributions of classification confidences** assigned by ALSTM and Adv-ALSTM for clean examples in validation and testing.
- Enforce margin.
Stock Prediction with NN
Conclusion and Future Work

Structured data
- Historical prices
- Domain knowledge

Unstructured data
- News reports
- Analyst reports

Stochasticity of historical price features should be considered.

Incorporating knowledge into the data driven learning model.

Fusion of traditional financial data and unstructured alternative data.

https://github.com/hennande/Adv-ALSTM
Thank You

Our research is supported by the National Research Foundation, Prime Minister’s Office of Singapore under its IRC@SG Funding Initiative.

For more info, please visit nextcenter.org
Stock Prediction with NN

Adversarial Training VS VAE

- **Bayesian Deep Learning**
  - Modeling historical prices as **stochastic variables** rather than static values.
  - [Xu, ACL’18] encodes historical prices with *Variational Autoencoder*.

- **Adversarial Training**
  - **Common training** updates model parameters to fit training data (*clean examples*), *i.e.*, make correct classifications.
  - **Adversarial training** additionally constructs *adversarial examples* via adding small *perturbations* to the input of clean examples, and encourages the model to correctly classify the *adversarial examples*.