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

Why Stock Prediction?

- **Which one to buy?**

All NASDAQ Securities

Displaying 1-50 of 3504 results Download this list

Name	Symbol	Market Cap	Country	IPO Year	Subsector
111, Inc.	YI	$47.89M	China	2018	Medical/Nursing Services
⋮	⋮	⋮	⋮	⋮	⋮

Exchange: NYSE

Displaying 1-50 of 3129 results Download this list

Name	Symbol	Market Cap	Country	IPO Year	Subsector
3D Systems Corporation	DDD	$1.07B	United States	n/a	Computer Software: Prepackaged Software
⋮	⋮	⋮	⋮	⋮	⋮

- **When to sell?**

  You may not have time, e.g., paper submission deadline.

Machine Learning for Stock Prediction

Standard Classification

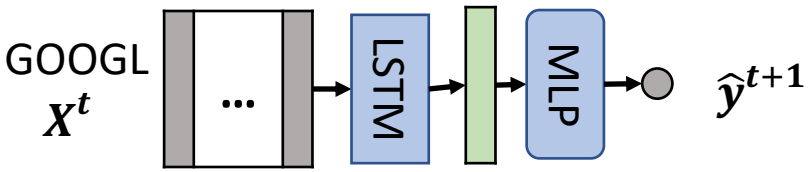
For each stock, on trading day t:

Historical prices → Stock features

Open
10AM
Close

$$X^t = \begin{bmatrix} t-w & & & t-1 & t \\ 1197.5 & 1172.6 & \cdots & 1120.2 & 1105.6 \\ 1199.3 & 1179.4 & & 1126.8 & 1113.4 \\ \vdots & \vdots & \ddots & \vdots & \\ 1173.3 & 1166.5 & \cdots & 1121.4 & 1106.5 \end{bmatrix}$$

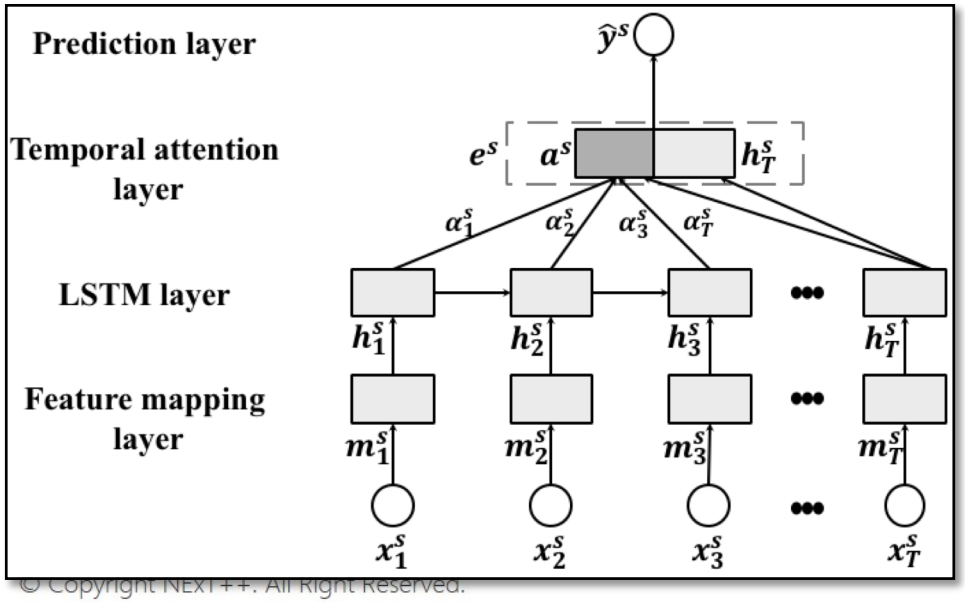
DNNs, especially RNN [Zhang, KDD'17][Xu, ACL'18] are used to solve asset price prediction as a standard classification problem.



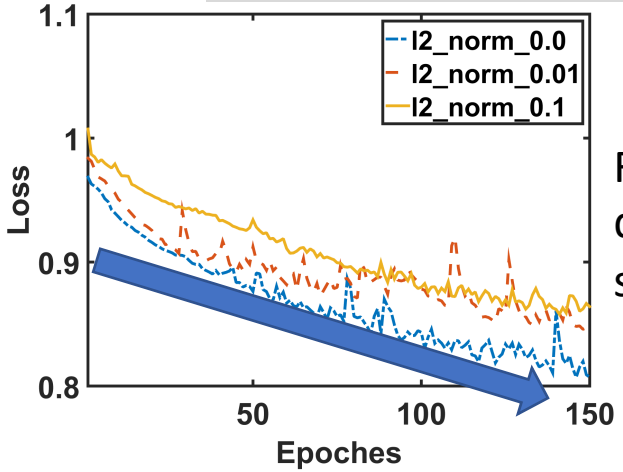
Machine Learning for Stock Prediction

Training Neural Network (NN)

- 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 $\boldsymbol{e^s}$.

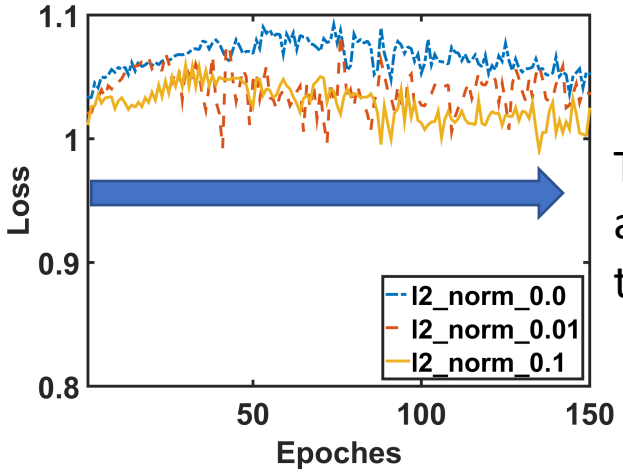


On a benchmark dataset



Training data

Fit the training data 20 months since 2015 Jan.



Testing data

Two months after the training period.

Stock Prediction with NN

Stocasticity of Stock Price Feature

Historical prices → Stock features

	$t-w$			$t-1$	t
Open	1197.5	1172.6	…	1120.2	1105.6
10AM	1199.3	1179.4		1126.8	1113.4
	⋮		⋱	⋮	
Close	1173.3	1166.5	…	1121.4	1106.5

X^t ⟶ +

Historical prices → Different stock features

	$t-w$			$t-1$	t
Open	1197.5	1172.6	…	1120.2	1105.6
10:01	1198.8	1179.4		1125.7	1113.4
	⋮		⋱	⋮	
Close	1173.3	1166.5	…	1121.4	1106.5

X^t ⟶ -

Observing the price at a slightly different time (10:00→10:01).

--- A new feature matrix

--- Different prediction (might wrong)

NN is sensitive to slight feature changes → poor generalization ability

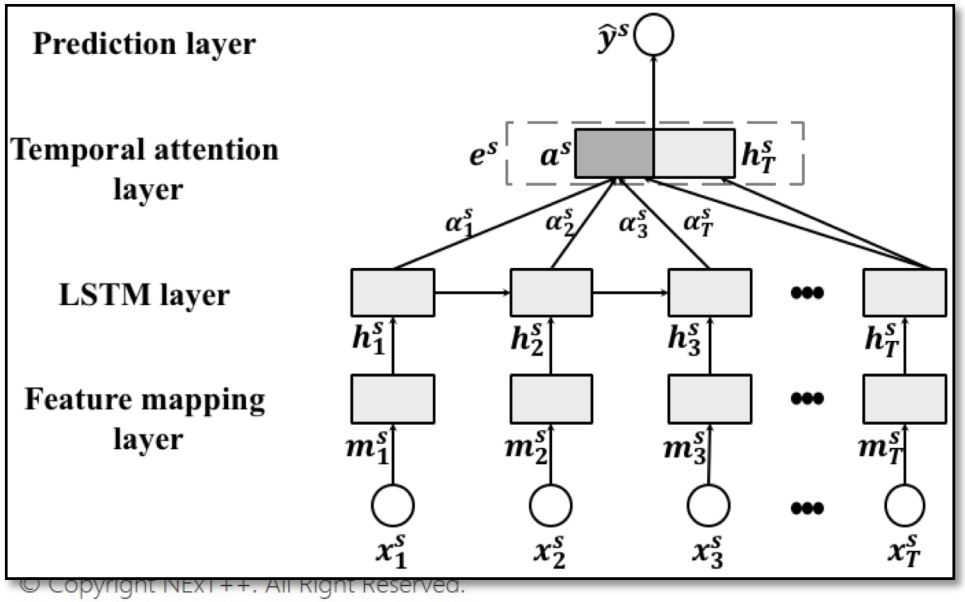
Stock Prediction with NN

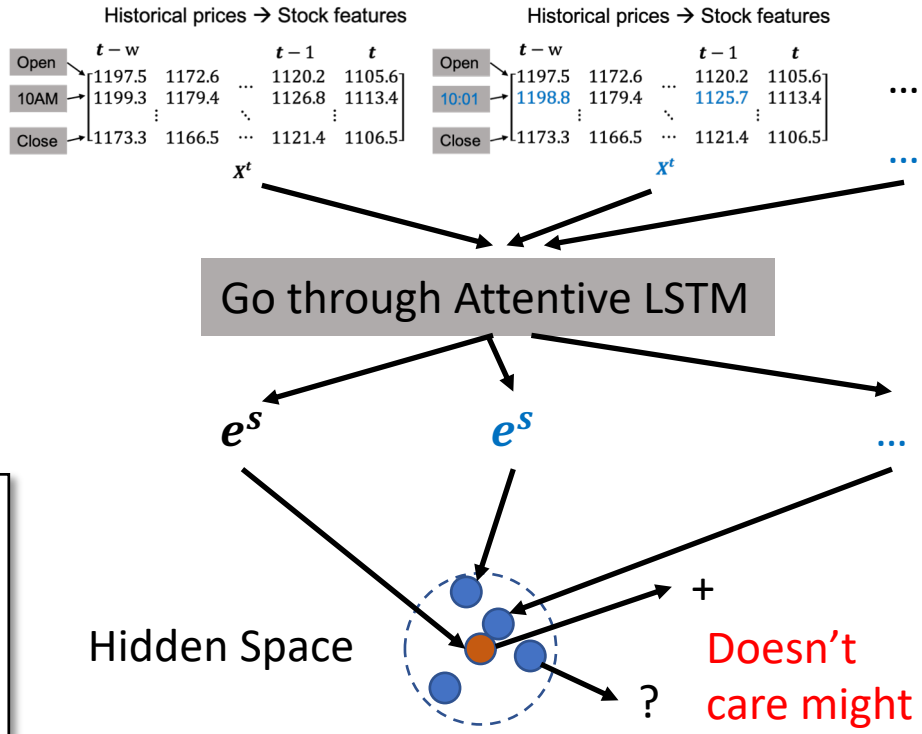
Stochasticity of Stock Price Feature

- 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.



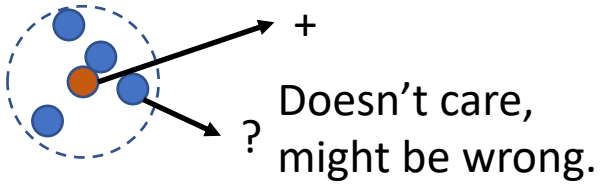


Stock Prediction with NN

Handling Stochasticity with Adversarial Training

- Standard training

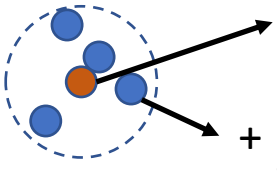
Updates model parameters to fit training data (*clean examples*)



Doesn't care, might be wrong.

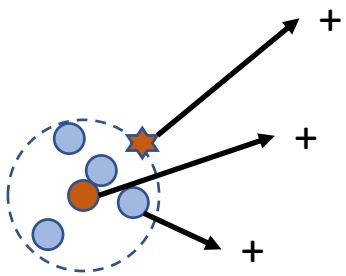
- Standard training (ideally)

Updates model parameters to fit *clean example* as well as all the other points.



Features observed at a slightly different time.

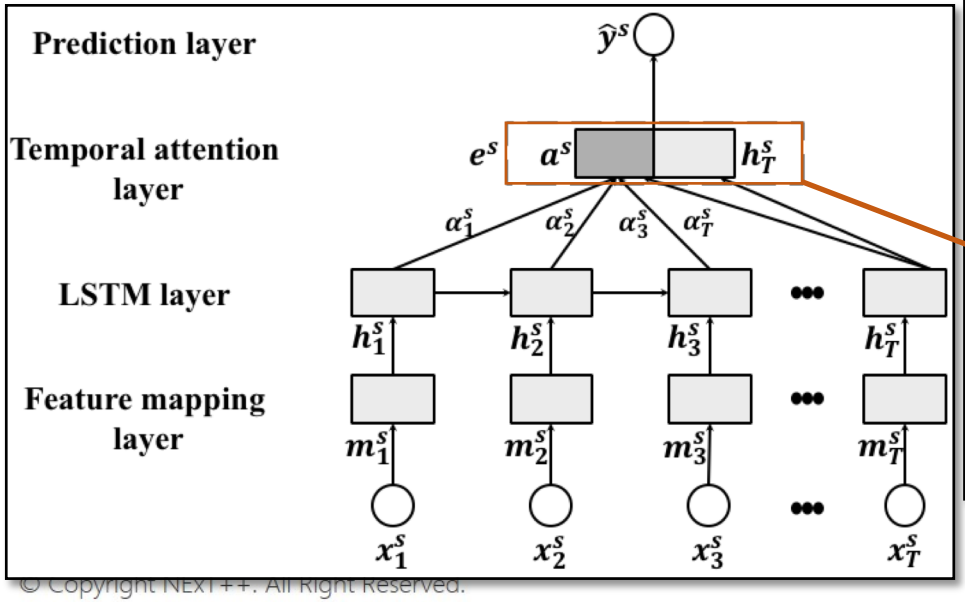
- Adversarial training
  - Additionally constructs *adversarial examples* via adding small *perturbations* to the input of clean examples, and encourages the model to correctly classify the adversarial examples.



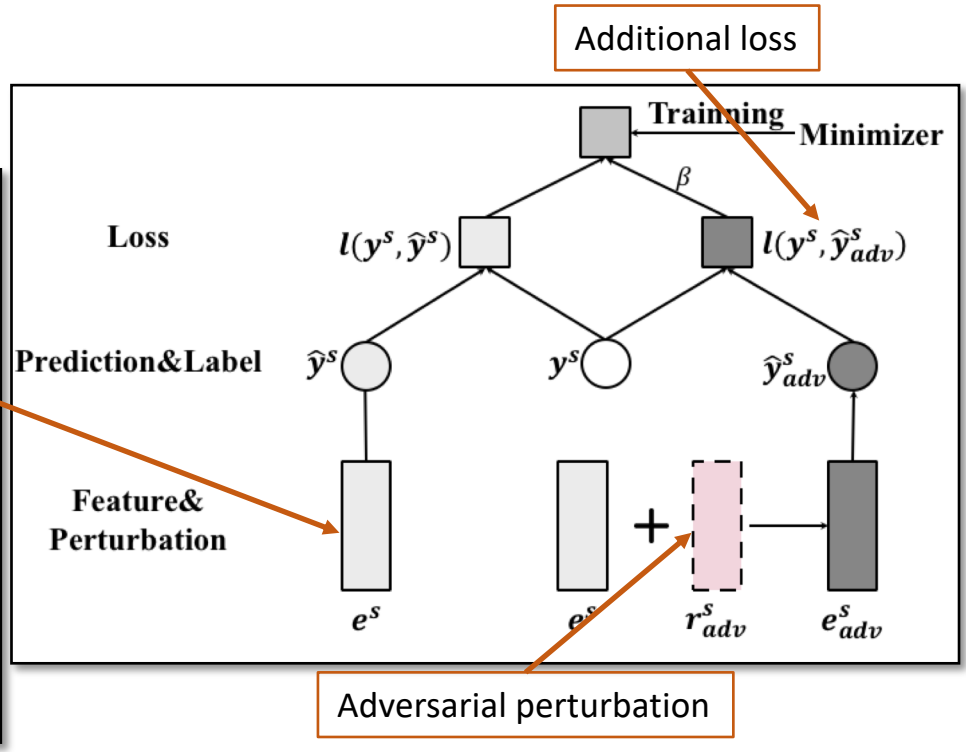
Adversarial example, the point within the range that is hardest to be predicted as +

Stock Prediction with NN
Adversarial Training

- 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 $\boldsymbol{e^s}$.



- Adversarial Training
  - Constructs *adversarial examples* via adding *perturbation* to latent representation $\boldsymbol{e^s}$.
  - 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].

Table 1: Statistics of the dataset.

	Training	Validation	Testing
Duration	Jan-01-2014 Jul-31-2015	Aug-01-2015 Sep-30-2015	Oct-01-2015 Dec-31-2015
#Examples (+)	10,305	1,139	1,908
#Examples (-)	10,010	1,416	1,812

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.

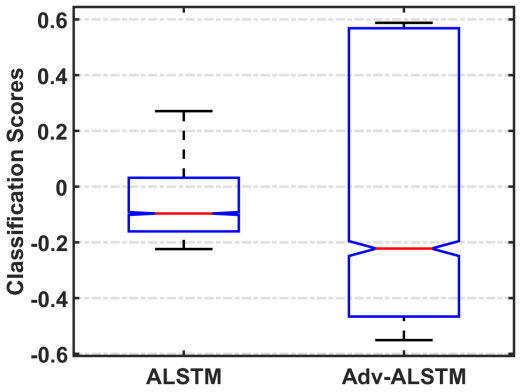
- Significant improvements.

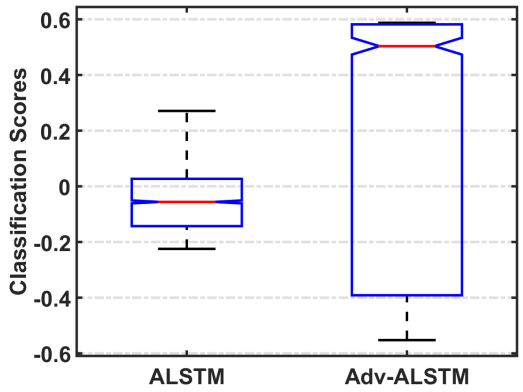
Table 2: Performance of the compared methods. **RI** denotes the relative improvement of **Adv-ALSTM** compared to the associated baseline.

Methods	Acc	RI	MCC	RI
RAND	50.89±——	12.40%	-0.0023±——	—
LSTM	53.18±5e-1	7.56%	0.0674±5e-3	120.03%
ALSTM	54.90±7e-1	4.02%	0.1043±7e-3	42.19%
StockNet	54.96±——	4.08%	0.0165±——	798.79%
Adv-ALSTM	**57.20**±——	—	**0.1483**±——	—

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

Stochasticity of historical price features should be considered.

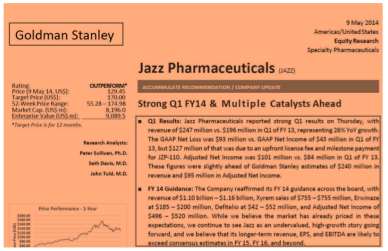
Historical prices

Incorporating knowledge into the data driven learning model.

Domain knowledge

Unstructured data

News reports



Analyst reports

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

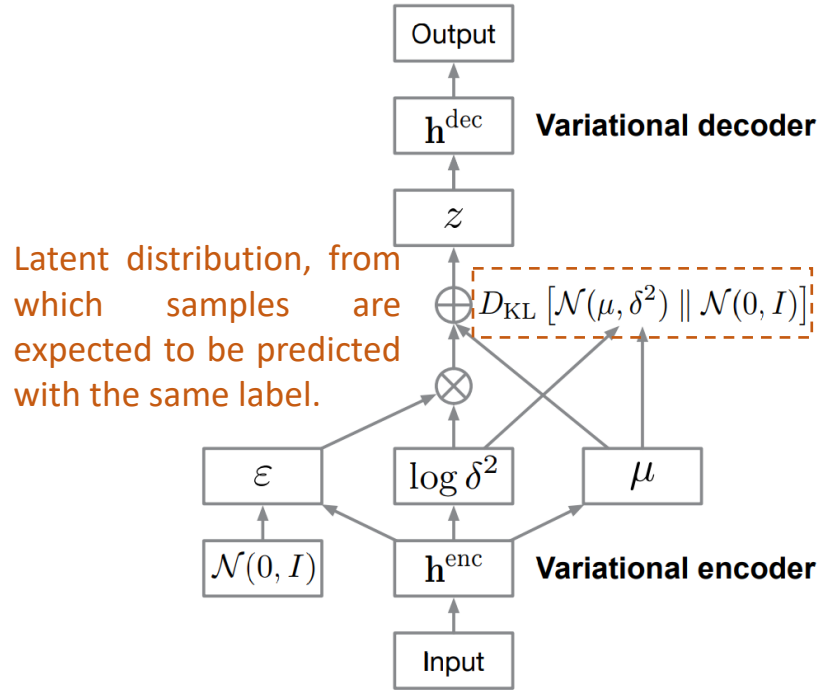
The End

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.