Latest Developments in Deep Learning in Finance

8th November 2019

Artificial Intelligence Finance Institute

NYU Courant

Artificial Intelligence Finance Institute

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

The Artificial Intelligence Finance Institute's (AIFI) mission is to be the world's leading educator in the application of artificial intelligence to investment management, capital markets and risk. We offer one of the industry's most comprehensive and in-depth educational programs, geared towards investment professionals seeking to understand and implement cutting edge AI techniques.

Taught by a diverse staff of world leading academics and practitioners, the AIFI courses teach both the theory and practical implementation of artificial intelligence and machine learning tools in investment management. As part of the program, students will learn the mathematical and statistical theories behind modern quantitative artificial intelligence modeling. Our goal is to train investment professionals in how to use the new wave of computer driven tools and techniques that are rapidly transforming investment management, risk management and capital markets.

Deep Learning in Finance

Machine Learning in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Supervised Learning

Predictive or Descriptive

Regression

Learn Regression Function

$f: \mathbb{R}^n \to \mathbb{R}$

Given: Inputs and outputs

(X_i, Y_i)

Classification

Learn Class Function

$f: \mathbb{R}^n \to \{1, \dots, k\}$

Given: Inputs and outputs

(X_i, C_i)

Unsupervised Learning

Descriptive

Clustering

Learn Class Function

$f: \mathbb{R}^n \to \{1, \dots, k\}$

Given: Inputs

(X_i)

Representation Learning

Learn Representer function

$f: \mathbb{R}^n \to \mathbb{R}^k$

Given : Inputs

(X_i)

Reinforcement Learning

Prescriptive

Learn Policy

Learn Policy Function

$f: \mathbb{R}^n \to \mathbb{R}^k$

Given : Tuples

(X_i, a_i, X_{i+1}, r_i)

Inverse Reinforcement Learning

Learn Reward Function

$f: \mathbb{R}^n \to \mathbb{R}$

Given : Tuples

(X_i, a_i, X_{i+1})

Machine Learning in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Supervised Learning

Regression

- Earnings Prediction
- Returns Prediction
- Algorithmic Trading
- Credit Losses

Classification

- Credit Ratings
- Sustainable Development Goals Scores
- Stock Picking
- Fraud
- AML

Unsupervised Learning

Clustering

- Customer Segmentation
- Stock classification

Representation Learning

- Factor Modeling Estimation
- Regime Changes

Reinforcement Learning

Learn Policy

- Trading Strategies
- Option Replication
- Marketing Strategies

Inverse Reinforcement Learning

- Reverse engineering of consumer
- Trading

Machine Learning in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

UNSUPERVISED CLUSTERING

- k-Means, FuzzyC-Means
- Hierarchical
- Neural Networks
- Gaussian Mixture
- Hidden Markov Models

SUPERVISED

CLASSIFICATION	REGRESSION
Support Vector Machines	Neural Networks
Discriminant Analysis	Decision Trees
Naïve Bayes	Ensemble Methods
Nearest Neighbors	Non-linearReg. (GLM, Logistic)
CART	Linear Regression

Deep Learning

- Multilayer Perceptron
- Convolutional Neural Networks
- Long Short Term Memory
- Restricted Boltzman Machine
- Auto Encoders

Reinforcement Learning

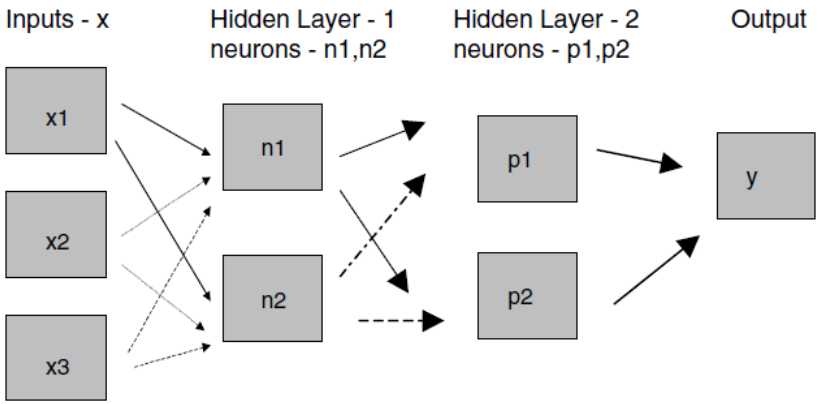
Deep Neural Networks

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Neural Networks

How it Works
Inspired by the human brain, a neural network consists of highly connected networks of neurons that relate the inputs to the desired outputs. The network is trained by iteratively modifying the strengths of the connections so that given inputs map to the correct response.

Best Used…
- For modeling highly nonlinear systems
- When data is available incrementally and you wish to constantly update the model
- When there could be unexpected changes in your input data
- When model interpretability is not a key concern



$$n_{k,t} = w_{k,0} + \sum_{i=1}^{i^*} w_{k,i}\, x_{i,t}$$

$$N_{k,t} = \frac{1}{1 + e^{-n_{k,t}}}$$

$$p_{l,t} = \rho_{l,0} + \sum_{k=1}^{k^*} w_{l,k}\, N_{k,t}$$

$$P_{l,t} = \frac{1}{1 + e^{-p_{l,t}}}$$

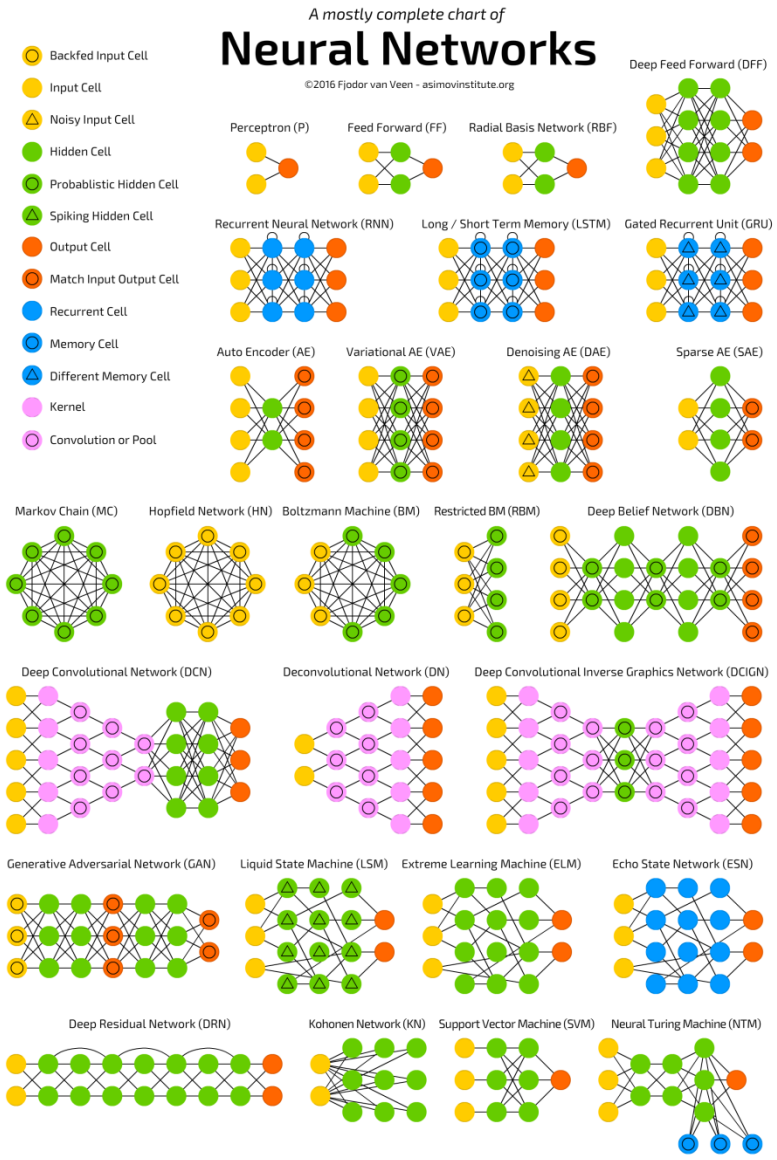
$$y_t = \gamma_0 + \sum_{l=1}^{l^*} \gamma_l\, P_{l,t}$$

Deep Learning

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Deep Learning

- Multilayer Perceptron
- Convolutional Neural Networks
- Long Short Term Memory
- Restricted Boltzman Machine



Deep Architectures in Finance - Pros and cons

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Pros
  - State of the art results in factor models, time series, classification
  - Deep Reinforcement Learning
  - XGBoost as a competing model
- Cons
  - Non Stationarity
  - Interpretability
  - Overfitting

Deep Learning in Finance

Modeling Aspects

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Classic Theorems on Compression and Model Selection

  - Minimum Description Length principle - The fundamental idea in MDL is to view learning as data compression. By compressing the data, we need to discover regularity or patterns in the data with the high potentiality to generalize to unseen samples. Information bottleneck theory believes that a deep neural network is trained first to represent the data by minimizing the generalization error and then learn to compress this representation by trimming noise.

  - Kolmogorov Complexity – Kolmogorov Complexity relies on the concept of modern computers to define the algorithmic (descriptive) complexity of an object: It is the length of the shortest binary computer program that describes the object. Following MDL, a computer is essentially the most general form of data decompressor.

  - Solomonoff's Inference Theory - Another mathematical formalization of Occam's Razor is Solomonoff's theory of universal inductive inference (Solomonoff, 1964). The principle is to favor models that correspond to the "shortest program" to produce the training data, based on its Kolmogorov complexity

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- The expressive power of DL models - Deep neural networks have an extremely large number of parameters compared to the traditional statistical models. If we use MDL to measure the complexity of a deep neural network and consider the number of parameters as the model description length, it would look awful. The model description can easily grow out of control. However, having numerous parameters is necessary for a neural network to obtain high expressivity power. Because of its great capability to capture any flexible data representation, deep neural networks have achieved great success in many applications.

    - **Universal Approximation Theorem** - The Universal Approximation Theorem states that a feedforward network with: 1) a linear output layer, 2) at least one hidden layer containing a finite number of neurons and 3) some activation function can approximate any continuous functions on a compact subset of to arbitrary accuracy. The theorem was first proved for sigmoid activation function (Cybenko, 1989). Later it was shown that the universal approximation property is not specific to the choice of activation (Hornik, 1991) but the multilayer feedforward architecture.

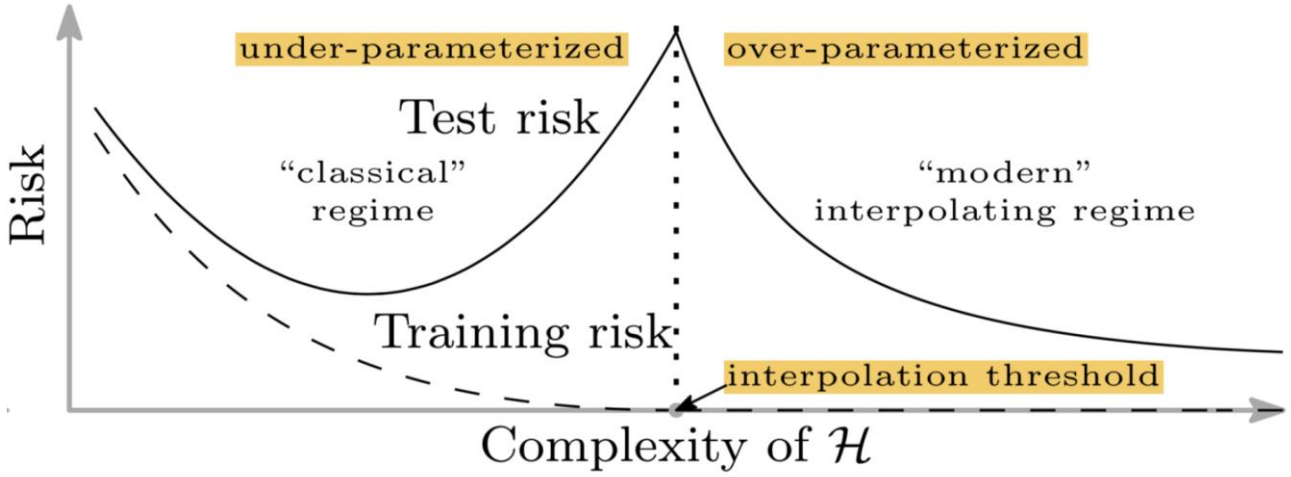
    - **Stochastic processes**

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Deep Learning and Overfitting (1)
  - **Modern risk curve for Deep Learning**
  - **Regularization and Generalization error -** Regularization is a common way to control overfitting and improve model generalization performance. Interestingly some research (Zhang, et al. 2017) has shown that explicit regularization (i.e. data augmentation, weight decay and dropout) is neither necessary or sufficient for reducing generalization error.
  - **Intrinsic Dimension** (Li et al, 2018). Intrinsic dimension is intuitive, easy to measure, while still revealing many interesting properties of models of different sizes. One intuition behind the measurement of intrinsic dimension is that, since the parameter space has such high dimensionality, it is probably not necessary to exploit all the dimensions to learn efficiently. If we only travel through a slice of objective landscape and still can learn a good solution, the complexity of the resulting model is likely lower than what it appears to be by parameter-counting. This is essentially what intrinsic dimension tries to assess.

Deep Architectures in Finance - Model Risk - W shape Bias-Variance ?



In a recent paper by Belkin et al. (2018) they reconciled the traditional bias-variance trade-offs and proposed a new double-U-shaped risk curve for deep neural networks. Once the number of network parameters is high enough, the risk curve enters another regime.

The paper claims that it is likely due to two reasons:

- The number of parameters is not a good measure of inductive bias, defined as the set of assumptions of a learning algorithm used to predict for unknown samples
- Equipped with a larger model, we might be able to discover larger function classes and further find interpolating functions that have smaller norm and are thus "simpler".

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Deep Learning and Overfitting (2)

  - **Heterogenous layer robustness** - Zhang et al. (2019) investigated the role of parameters in different layers. The fundamental question raised by the paper is: “are all layers created equal?” The short answer is: No. The model is more sensitive to changes in some layers but not others. Layers can be categorized into two categories with the help of these two operations:
    - Robust Layers: The network has no or only negligible performance degradation after **re-initializing or re-randomizing the layer.**
    - Critical Layers: Otherwise.

  - **Lottery ticket hypothesis** - The lottery ticket hypothesis (Frankle & Carbin, 2019) is another intriguing and inspiring discovery, supporting that only a subset of network parameters have impact on the model performance and thus the network is not overfitted. The lottery ticket hypothesis states that a randomly initialized, dense, feed-forward network contains a pool of subnetworks and among them only a subset are “winning tickets” which can achieve the optimal performance when trained in isolation.

Deep Learning in Finance

Time Series

Long Short Term Memory Networks

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

Deep Learning in Finance: Prediction of Stock Returns with Long Short Term Memory Networks

Miquel Noguer Alonso *[1], Gilberto Batres-Estrada †[2], and Aymeric Moulin ‡[3]

[1] *Department of Industrial Engineering & Operations Research, Columbia University, New York, USA. Department of Economics, Finance and Accounting, ESADE, Barcelona, Spain. Institute d'Estudis Financers, IEF, Barcelona, Spain.* [2] *Webstep, Bergen, Norway.* [3] *Department of Industrial Engineering & Operations Research, Columbia University, New York, USA.*

ABSTRACT

In this paper we investigate the profitability of a quantitative trading strategy based on Deep Learning methods. Specifically we focus on a variant of the Recurrent Neural Network (RNN), the Long Short Term Memory Network (LSTM) and show its predictive power on stock price data. We use LSTM networks for selecting stocks using historical price. The reason why RNNs are good for regression or classification of time series or data where time ordering matters is that RNNs capture the variation through time, thanks to its internal state dynamics. We made two studies, the first focuses on predicting stock returns using one stock at a time. The hit-ratio in this experiment lies in the range 0.47 and 0.60 for the worst respectively best performing stock on unseen "live" data. The second experiment looks at the whole universe of stocks simultaneously. In this experiment our model achieves a hit-ratio between 0.50 and 0.71 on unseen "live" data. From this experiment two portfolios were constructed, a long portfolio and a long-short portfolio with a Sharpe ratio of 8 respectively 10 for each of the portfolios. Our stock universe in both studies is composed of 50 stocks from the S&P 500.

Keywords: Deep Learning, Neural Networks, Recurrent Neural Networks, LSTM, Back-propagation through time, AI, finance, stock prediction, time series.

Long Short Term Memory Networks

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

The Long Short-Term Memory Network (LSTM) is a set of subnets with recurrent connections, known as memory blocks. Each block contains one or more self-connected memory cells and three multiplicative units known as the input, output and forget gates which respectively support read, write, and reset operations for the cells [14]. The gating units control the gradient-flow through the memory cell and when closing them allows the gradient pass without alteration for an indefinite amount of time, making the LSTM suitable for learning long time dependencies. Thus overcoming the vanishing gradient problem that RNNs suffer from. We describe in more detail the inner workings of an LSTM cell. The memory cell is composed of an input node, an input gate, an internal state, a forget gate and an output gate. First let bold letters denote vectors, each defined at time step t, then the components in a memory cell are as follows:

- The input node takes the activation from both the input layer, $\boldsymbol{x}_t$, and the hidden state $\boldsymbol{h}_{t-1}$ at time $t-1$. The input is then fed to an activation function, either a tanh or sigmoid.
- The input gate uses a sigmoidal unit that get its input from the current data $\boldsymbol{x}_t$ and the hidden units at time step $t-1$. The input gate multiplies the value of the input node, and because it is a sigmoid unit with range between zero and one, it can control the flow of the signal it multiplies.
- The internal state has a self-recurrent connection with unit weight also called the constant error carousel in [18], and is given by $\boldsymbol{s}_t = \boldsymbol{g}_t \odot \boldsymbol{i}_t + \boldsymbol{f}_t \odot \boldsymbol{s}_{t-1}$. The Hadamard product $\odot$, denotes element-wise product, and $\boldsymbol{f}_t$ is the forget gate, see below.
- The forget gate, $\boldsymbol{f}_t$, was not part of the original model for the LSTM but was introduced by [11]. The forget gate multiplies the internal state at time step $t-1$ and can in that way get

rid of all the contents in the past as demonstrated by the equation in the list item above.

- The resulting output from a memory cell is produced by multiplying the value of the internal state s_c by the output gate o_c. Often the internal state is run through a tanh activation function.

The equations for the LSTM network can be summarized as follows. As before let $\boldsymbol{g}$ stand for the input to the memory cell, $\boldsymbol{i}$ for the input gate, $\boldsymbol{f}$ for the forget gate, $\boldsymbol{o}$ for the output gate and $\boldsymbol{s}$ for the cell state, then we have, [21]

$$\boldsymbol{g}_t = \phi(W^{gX}\boldsymbol{x}_t + W^{gh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_g) \tag{13}$$
$$\boldsymbol{i}_t = \sigma(W^{iX}\boldsymbol{x}_t + W^{ih}\boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{14}$$
$$\boldsymbol{f}_t = \sigma(W^{fX}\boldsymbol{x}_t + W^{fh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{15}$$
$$\boldsymbol{o}_t = \sigma(W^{oX}\boldsymbol{x}_t + W^{oh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \tag{16}$$
$$\boldsymbol{s}_t = \boldsymbol{g}_t \odot \boldsymbol{i}_t + \boldsymbol{s}_{t-1} \odot \boldsymbol{f}_t \tag{17}$$
$$\boldsymbol{h}_t = \phi(\boldsymbol{s}_t) \odot \boldsymbol{o}_t. \tag{18}$$

where the Hadamard product $\odot$ denotes element-wise multiplication. In the equations $\boldsymbol{h}_t$ is the value of the hidden layer at time t, while $\boldsymbol{h}_{t-1}$ is the output by each memory cell in the hidden layer at time $t-1$. The weights $\{W^{gX}, W^{iX}, W^{fX}, W^{oX}\}$ are the connections between the inputs $\boldsymbol{x}_t$ with the input node, the input gate, the forget gate and the output gate respectively. In the same manner $\{W^{gh}, W^{ih}, W^{fh}, W^{oh}\}$ represent the connections between the hidden layer with the input node, the input gate, the forget gate and the output gate respectively. The bias terms for each of the cell's components is given by $\{\boldsymbol{b}_g, \boldsymbol{b}_i, \boldsymbol{b}_f, \boldsymbol{b}_o\}$.

Long Short Term Memory Networks - Results

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Stock	Hidden Units	HR LSTM	HR SVM	HR NN
AAPL	150	0.53	0.52	0.52 (130)
MSFT	100	0.51	0.49	0.49 (150)
FB	100	0.58	0.58	0.56 (90)
AMZN	100	0.55	0.56	0.53 (90)
JNJ	100	0.52	0.47	0.50 (50)
BRK/B	150	0.51	0.51	0.51 (50)
JPM	100	0.52	0.51	0.50 (90)
XOM	100	0.52	0.52	0.49 (50)
GOOGL	100	0.54	0.53	0.53 (70)
GOOG	100	0.55	0.55	0.55 (50)
BAC	100	0.47	0.50	0.59 (50)
PG	100	0.50	0.50	0.50 (110)
T	150	0.52	0.48	0.50 (70)
WFC	150	0.51	0.47	0.50 (70)
GE	100	0.51	0.50	0.50 (110)
CVX	150	0.50	0.53	0.50 (70)
PFE	100	0.49	0.49	0.49 (50)
VZ	150	0.51	0.53	0.50 (50)
CMCSA	150	0.54	0.49	0.50 (110)
UNH	100	0.52	0.48	0.52 (130)
V	100	0.59	0.51	0.55 (70)
C	150	0.52	0.50	0.51 (50)
PM	100	0.56	0.56	0.52 (110)
HD	100	0.53	0.50	0.53 (70)
KO	150	0.51	0.48	0.50 (70)
MRK	200	0.54	0.49	0.50 (110)
PEP	100	0.55	0.52	0.51 (50)
INTC	150	0.53	0.45	0.51 (110)
CSCO	100	0.51	0.48	0.50 (90)
ORCL	150	0.52	0.48	0.50 (130)
DWDP	150	0.51	0.48	0.50 (90)
DIS	150	0.53	0.49	0.52 (130)
BA	100	0.54	0.53	0.51 (50)
AMGN	100	0.51	0.52	0.53 (90)
MCD	150	0.55	0.48	0.52 (130)
MA	100	0.57	0.57	0.55 (130)
IBM	100	0.49	0.49	0.50 (50)
MO	150	0.55	0.47	0.52 (50)
MMM	100	0.53	0.46	0.52 (90)
ABBV	100	0.60	0.38	0.41 (110)
WMT	100	0.52	0.50	0.51 (50)
MDT	150	0.52	0.49	0.50 (50)
GILD	100	0.50	0.52	0.51 (70)
CELG	100	0.51	0.52	0.50 (90)
HON	150	0.55	0.46	0.52 (130)
NVDA	100	0.56	0.55	0.54 (90)
AVGO	100	0.57	0.57	0.51 (130)
BMY	200	0.52	0.49	0.50 (50)
PCLN	200	0.54	0.54	0.53 (70)
ABT	150	0.50	0.47	0.50 (70)

Training period	HR %	Avg Ret %(L)	Avg Ret % (L/S)	Sharpe ratio (L)	Sharpe ratio (L/S)
2000-2003	49.7	-0.05	-0.12	-0.84	-1.60
2001-2004	48.1	0.05	-0.02	2.06	-0.73
2002-2005	52.5	0.11	0.10	6.05	3.21
2003-2006	55.9	0.10	0.16	5.01	5.85
2004-2007	54.0	0.14	0.12	9.07	5.11
2005-2008	61.7	0.26	0.45	7.00	9.14
2006-2009	59.7	0.44	1.06	3.10	6.22
2007-2010	53.8	0.12	0.12	5.25	2.70
2008-2011	56.5	0.20	0.26	6.12	6.81
2009-2012	62.8	0.40	0.68	6.31	9.18
2010-2013	55.4	0.09	0.14	3.57	3.73
2011-2014	58.1	0.16	0.21	5.59	6.22
2012-2015	56.0	0.15	0.21	5.61	5.84

Long Short Term Memory Networks - Conclusions

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

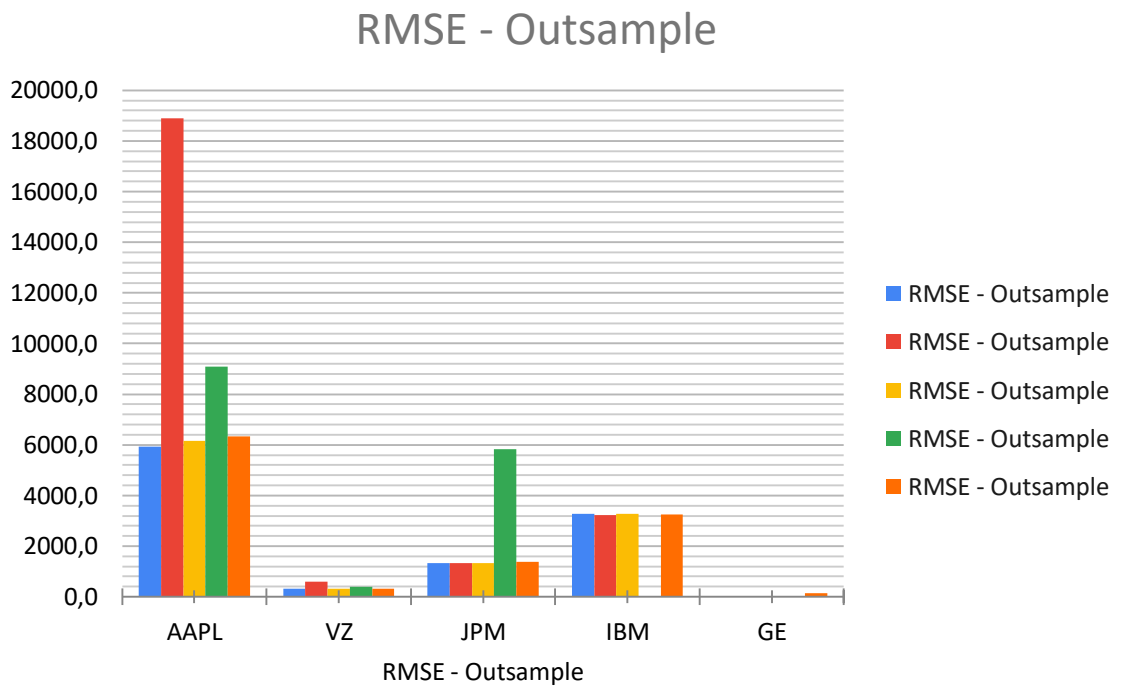
These ratios show a good behavior in-sample and out-of-sample. Sharpe ratios of our portfolio experiments are 8 for the long only portfolio and 10 for the long short version, an equally weighted portfolio would have provided a 2.7 Sharpe ratio using the model from 2014 to 2017. Results show consistency when using the same modeling approach in different market regimes. No trading costs have been considered.

We can conclude that LSTM networks are a promising modeling tool in financial time series especially in the multivariate LSTM networks with exogeneous variables. These networks can enable financial engineers to model time dependencies, non-linearity, feature discovery with a very flexible model that might be able to offset the very challenging estimation and non-stationarity in finance and the potential of overfitting. These issues can never be underestimated in finance even more in models with a high number of parameters, non-linearity and difficulty to interpret like LSTM networks.
We think financial engineers should then incorporate deep learning not only to model unstructured but also to structured data. We have very interesting modeling times ahead of us.

Other Time series Results - Joint work with Sonam Srivastava

- Model out of sample results – 3-2016 – 9/2019

Abs Error Returns	ARIMA	SVR	DeepReg	CNN	LSTM
VZ	5.076315	5.217527	5.074014	5.719326	5.687398
JPM	5.568193	5.977256	5.560975	5.903769	6.180781
IBM	5.300373	5.52681	5.347594	6.468016	5.557617
GE	7.632332	7.852825	7.675091	8.514779	8.788238
AAPL	6.987491	6.762491	6.897164	7.663094	7.33361

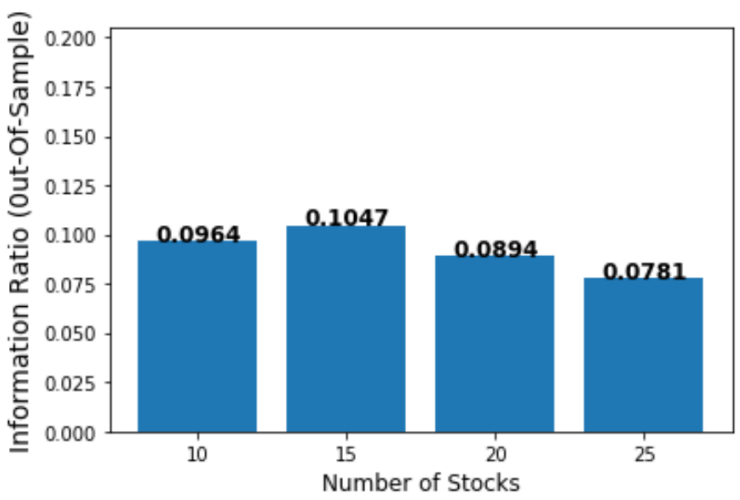


Deep Learning in Finance

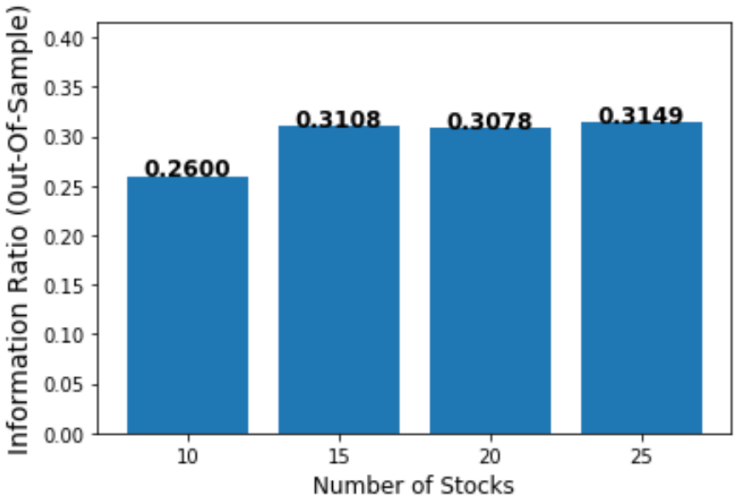
Factor Models

Factor Model Results

- 218 SP 500 Stocks - Selection of the x top performing stocks from the universe
- each out-of-sample period.
- Bloomberg Factors



Linear Regression



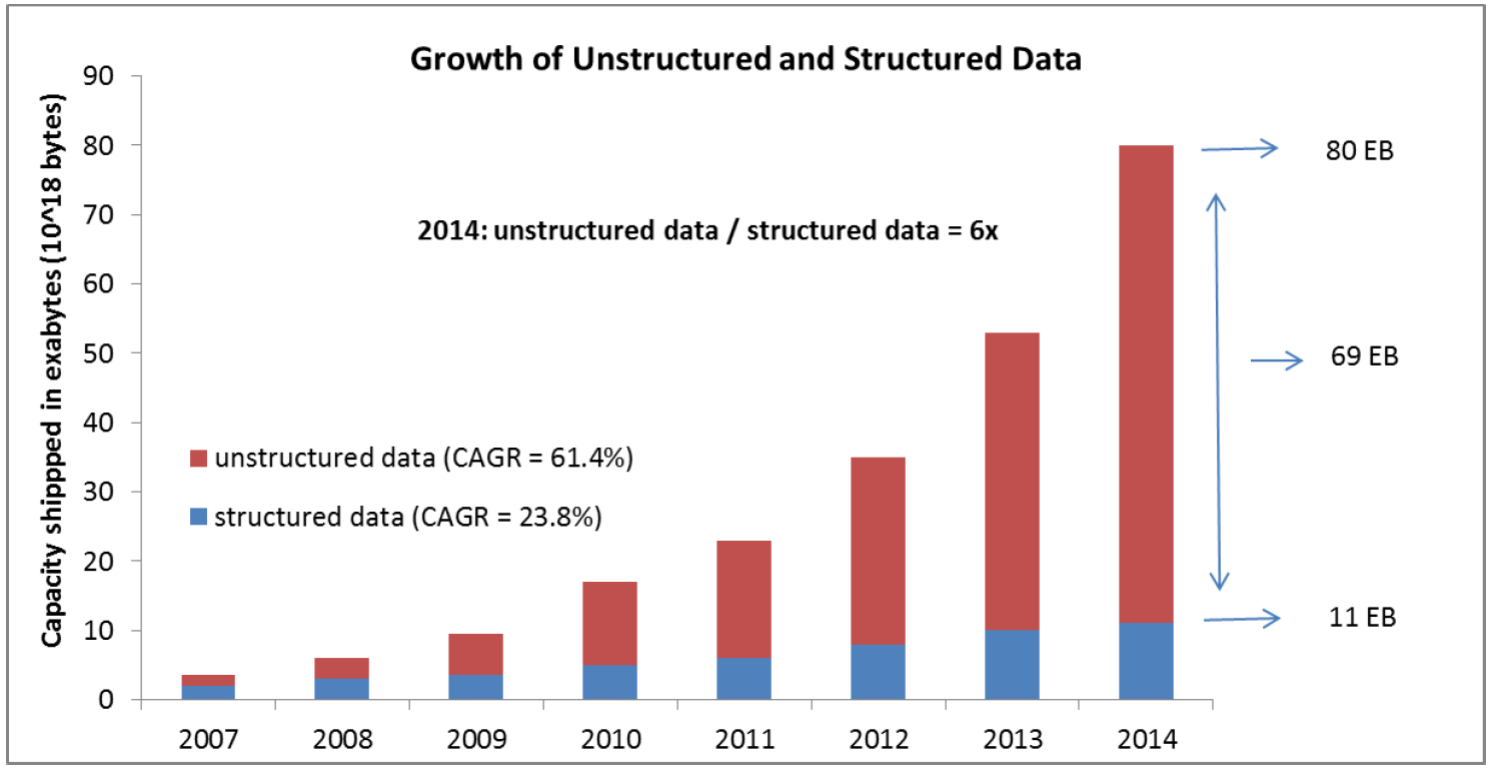
FFWD Neural Network

Deep Learning in Finance

Language Models

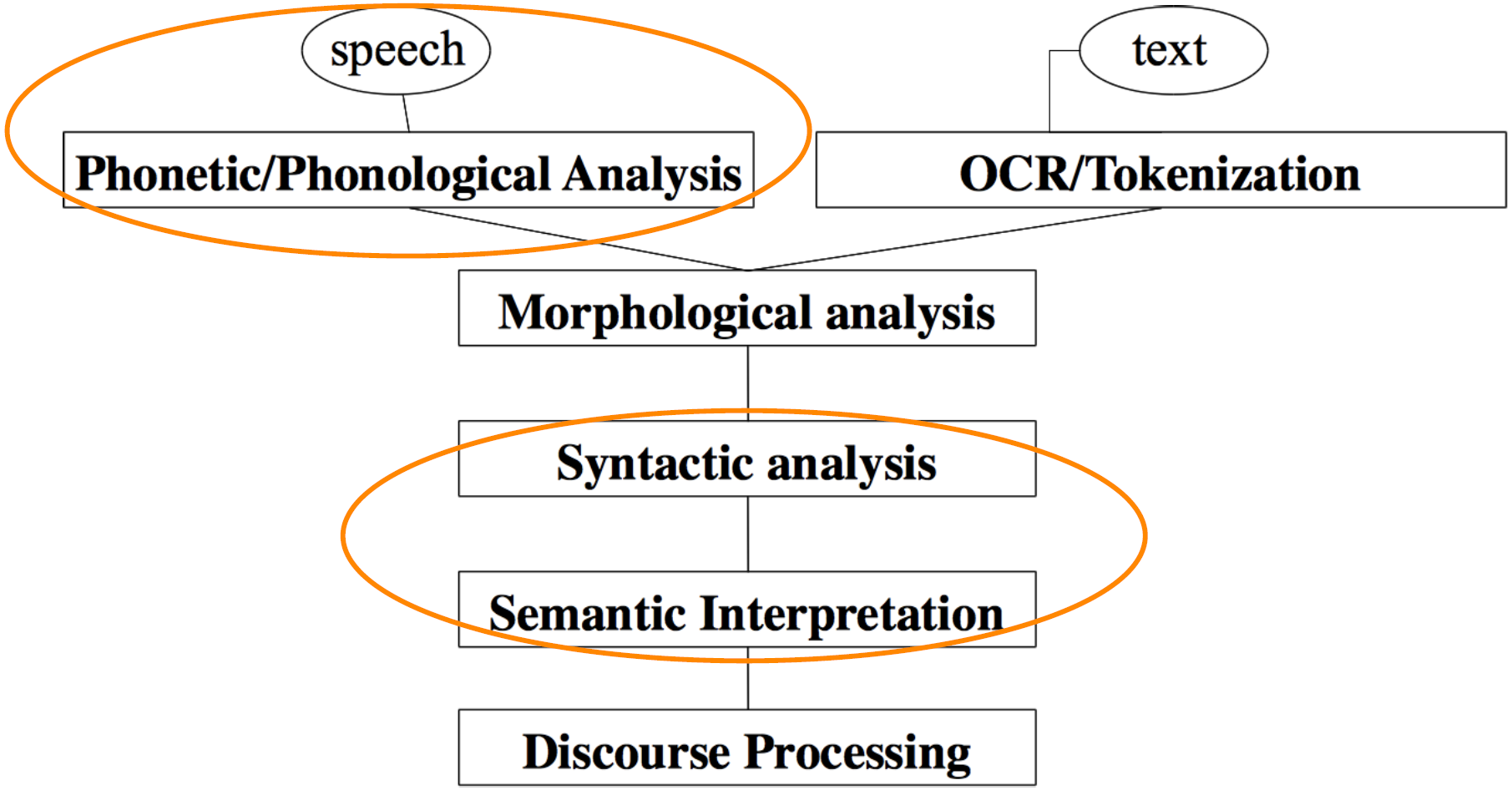
Historical Growth of Unstructured & Structured Data

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE



Natural Language Processing Levels

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE



Natural Language Processing Applications

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Applications range from simple to complex:

- Spell checking, keyword search, finding synonyms
- Extracting information from websites such as product price, dates, location, people or company names
- Classifying: reading level of school texts, positive/negative sentiment of longer documents
- Machine translation
- Spoken dialog systems
- Complex question answering

NLP in Industry

- Online advertisement matching
- Search
- Automated/assisted translation
- Sentiment analysis for marketing or finance/trading
- Speech recognition
- Chatbots / Dialog agents
- Automating customer support
- Controlling devices
- Ordering goods

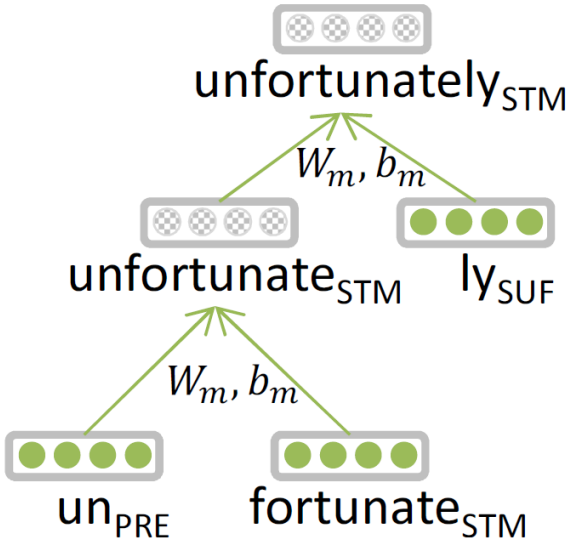
Representations of NLP Levels: Morphology

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Traditional: Words are made of morphemes

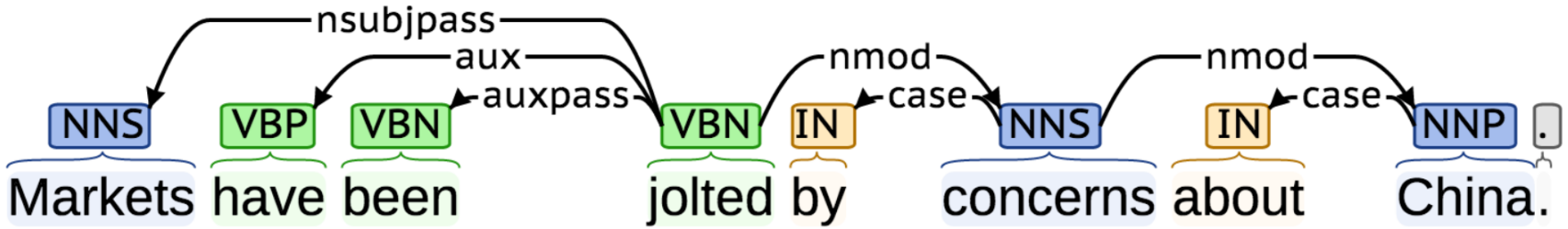
prefix	stem	suffix
un	interest	ed

- DL:
  - every morpheme is a vector
  - a neural network combines two vectors into one vector
  - Luong et al. 2013



NLP Tools: Parsing for sentence structure

Neural networks can accurately determine the structure of sentences, supporting interpretation



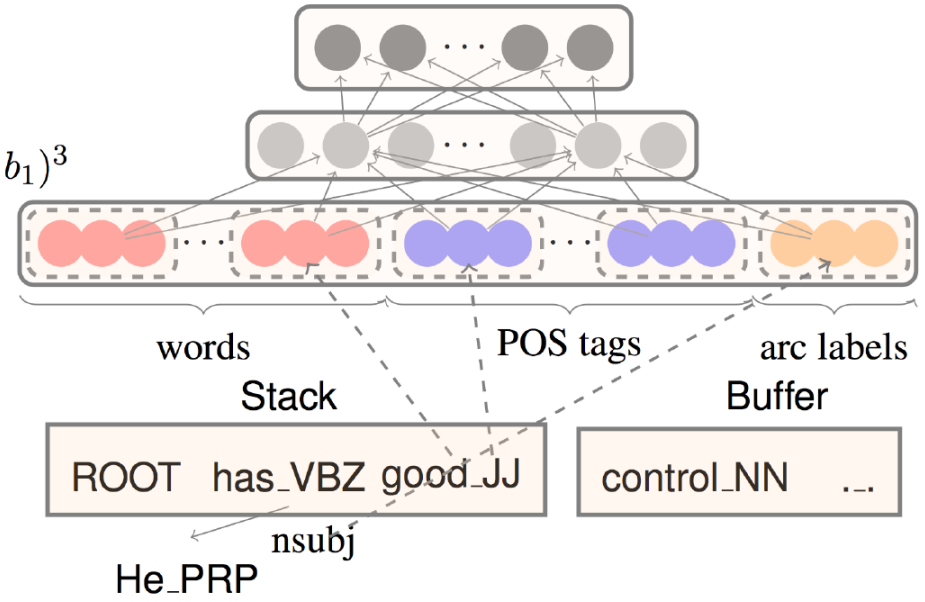
Softmax layer:

$$p = \texttt{softmax}(W_2 h)$$

Hidden layer:

$$h = (W_1^w x^w + W_1^t x^t + W_1^l x^l + b_1)^3$$

Input layer: $[x^w, x^t, x^l]$

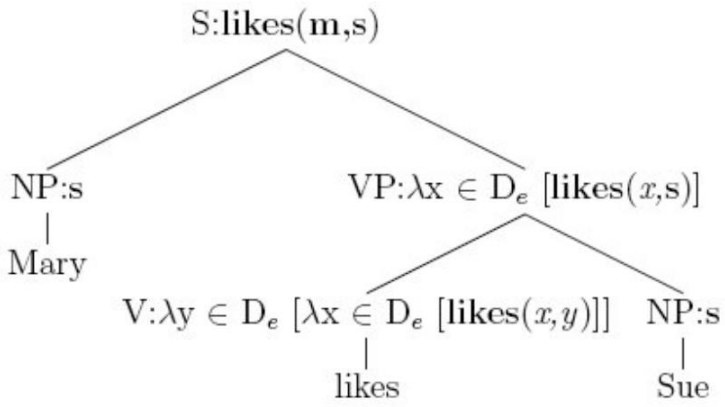


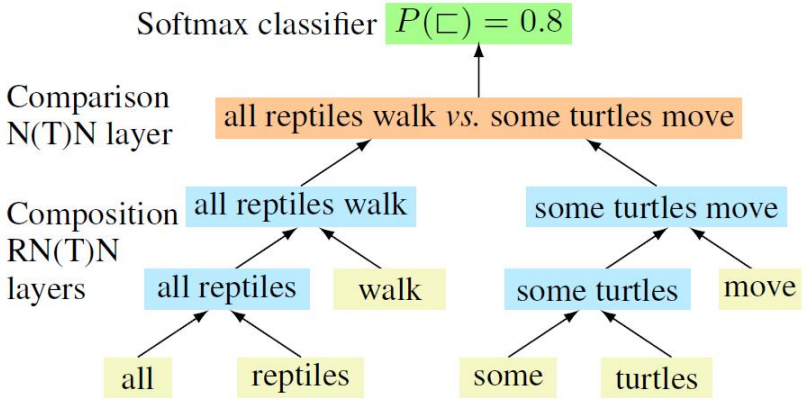
Configuration

Representations of NLP Levels: Semantics

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Traditional: Lambda calculus
  - Carefully engineered functions
  - Take as inputs specific other functions
  - No notion of similarity or fuzziness of language
- DL:
  - Every word and every phrase and every logical expression is a vector
  - a neural network combines two vectors into one vector
  - Bowman et al. 2014

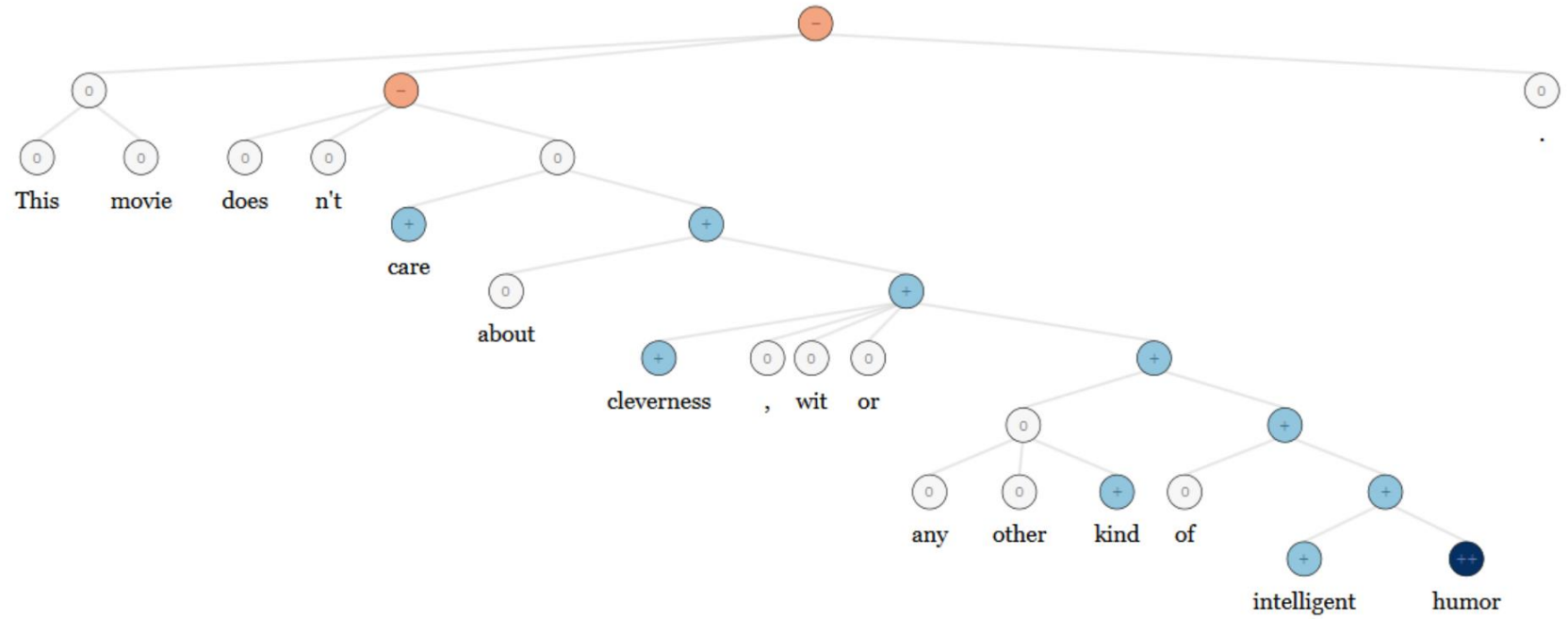




NLP Applications: Sentiment Analysis

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Traditional: Curated sentiment dictionaries combined with either bag-of-words representations (ignoring word order) or handdesigned negation features (ain't gonna capture everything)
- Same deep learning model that was used for morphology, syntax and logical semantics can be used! - RecursiveNN



How do we represent the meaning of a word?

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Definition: **meaning** (Webster dictionary)

- the idea that is represented by a word, phrase, etc.
- the idea that a person wants to express by using words, signs, etc.
- the idea that is expressed in a work of writing, art, etc.

Commonest linguistic way of thinking of meaning:

signifier (symbol) ⟺ signified (idea or thing)

= denotational semantics

Representing words as discrete symbols

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Example: in web search, if user searches for "Seattle motel", we would like to match documents containing "Seattle hotel".

But:

motel = [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
hotel = [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]

These two vectors are orthogonal.

There is no natural notion of **similarity** for one-hot vectors!

Solution:

- Could try to rely on WordNet's list of synonyms to get similarity?
  - But it is well-known to fail badly: incompleteness, etc.
- **Instead: learn to encode similarity in the vectors themselves**

Representing words by their context

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Distributional semantics: **A word's meaning is given by the words that frequently appear close-by**
  - *"You shall know a word by the company it keeps"* (J. R. Firth 1957: 11)
  - One of the most successful ideas of modern statistical NLP!
- When a word *w* appears in a text, its **context** is the set of words that appear nearby (within a fixed-size window).
- Use the many contexts of *w* to build up a representation of *w*

…government debt problems turning into ***banking*** *crises as happened in 2009…*
…saying that Europe needs unified ***banking*** *regulation to replace the hodgepodge…*
…India has just given its ***banking*** *system a shot in the arm…*

These context words will represent ***banking***

Word Vectors

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

We will build a dense vector for each word, chosen so that it is similar to vectors of words that appear in similar contexts

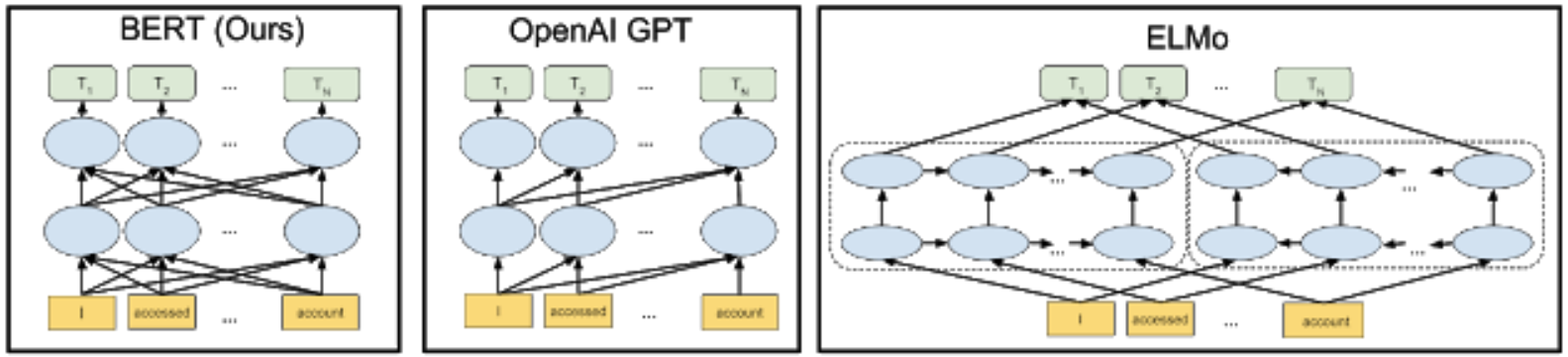
$$\textit{banking} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$

Note: word vectors are sometimes called word embeddings or word representations. They are a distributed representation.

BERT Model

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

- BERT stands for Bidirecccional Encoder Representations from Transformers

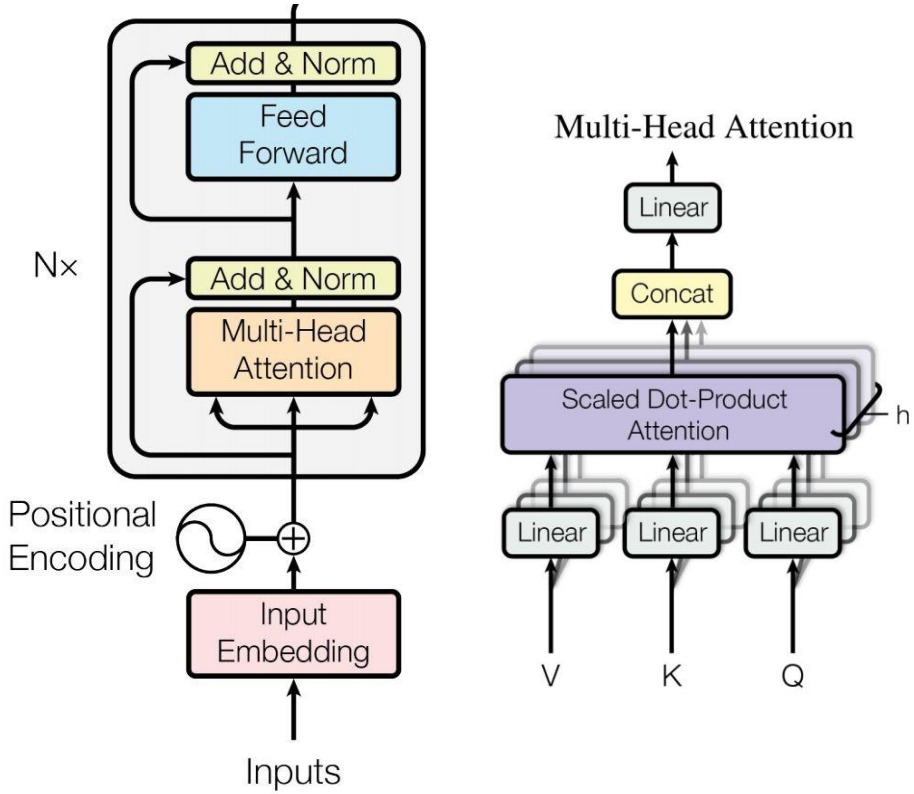


BERT Model Architecture

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Transformer encoder

- Multi-headed self attention
  - Models context
- Feed-forward layers
  - Computes non-linear hierarchical features
- Layer norm and residuals
  - Makes training deep networks healthy
- Positional embeddings
  - Allows model to learn relative positioning



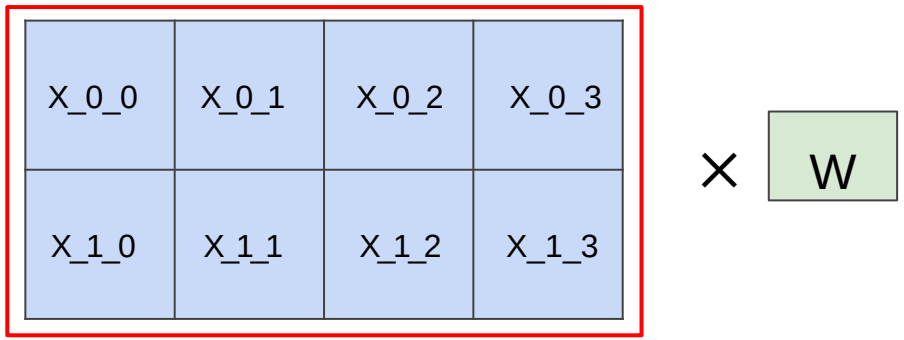
BERT Model Architecture

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

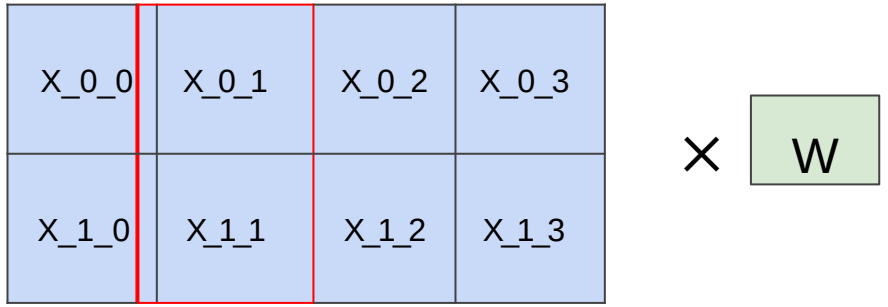
- Empirical advantages of Transformer vs. LSTM:

1. Self-attention = no locality bias
   - Long-distance context has "equal opportunity"
2. Single multiplication per layer = efficiency on TPU
   - Effective batch size is number of *words*, not *sequences*

Transformer



LSTM



SQuAD 1.1

What was another term used for the oil crisis?
Ground Truth Answers: first oil shock | shock | shock | first oil shock | shock
Prediction: shock

The 1973 oil crisis began in October 1973 when the members of the Organization of Arab Petroleum Exporting Countries (OAPEC, consisting of the Arab members of OPEC plus Egypt and Syria) proclaimed an oil embargo. By the end of the embargo in March 1974, the price of oil had risen from US$3 per barrel to nearly $12 globally; US prices were significantly higher. The embargo caused an oil crisis, or "shock", with many short- and long-term effects on global politics and the global economy. It was later called the "first oil shock", followed by the 1979 oil crisis, termed the "second oil shock."

- Only new parameters: Start vector and end vector.
- Sof itions.

maxover all pos

$$P_i = \frac{e^{S \cdot T_i}}{\sum_j e^{S \cdot T_j}}$$

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar et al. '16)	82.304	91.221
1 Oct 05, 2018	BERT (ensemble) *Google AI Language* https://arxiv.org/abs/1810.04805	**87.433**	**93.160**
2 Oct 05, 2018	BERT (single model) *Google AI Language* https://arxiv.org/abs/1810.04805	85.083	91.835
2 Sep 26, 2018	nlnet (ensemble) *Microsoft Research Asia*	85.954	91.677
5 Sep 09, 2018	nlnet (single model) *Microsoft Research Asia*	83.468	90.133
3 Jul 11, 2018	QANet (ensemble) *Google Brain & CMU*	84.454	90.490

BERT Results with IMDB Data Set – Joint work with Tinghao Li

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

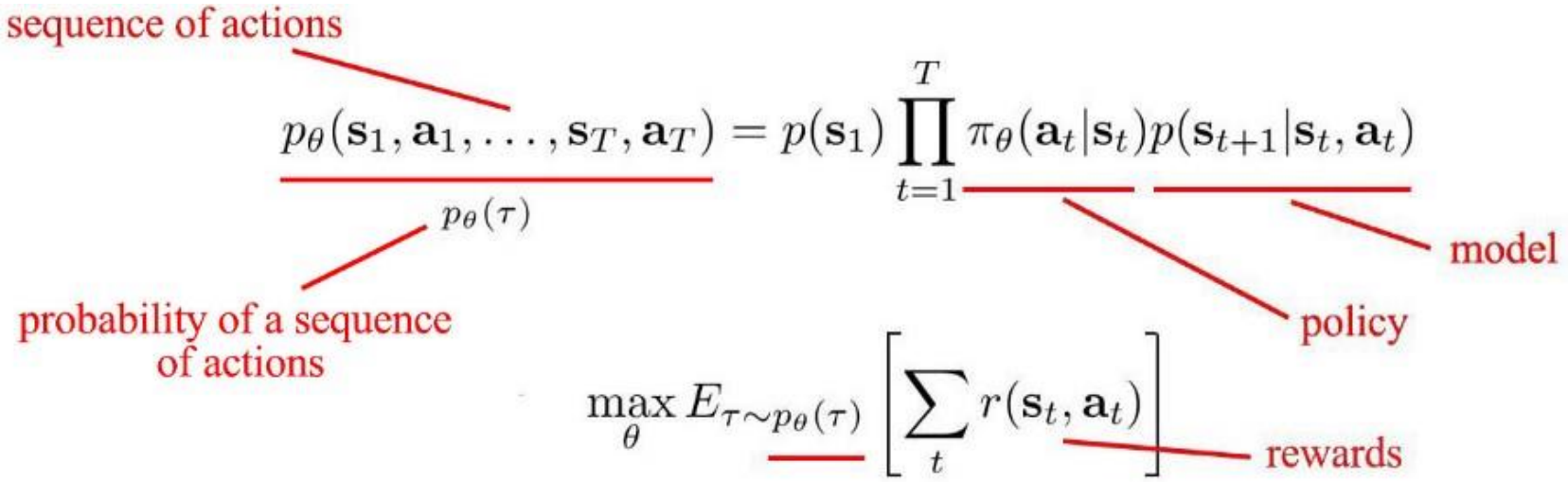
- BERT Base version 110 million parameters
- After 17 minutes training in one GPU core, it achieved 97% accuracy for the classification with F1 score at 96%

Deep Learning in Finance

Deep Reinforcement Learning

Defining the RL problem

In RL, we want to find a sequence of actions that maximize expected rewards or minimize cost.



But there are many ways to solve the problem. For example, we can

- Analyze how good to reach a certain state or take a specific action (i.e. Value-learning),
- Use the model to find actions that have the maximum rewards (model-based learning), or
- Derive a policy directly to maximize rewards (policy gradient).

Deep Learning in Finance

Conclusions

Deep Learning in Finance Instructions of Use

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

	Input	Processing	Output	Applications	Benefits	Challenges
Multi Layer Perceptrons	Prices/ Returns/ Factors	Classification and Regression	Forecasting / Explanation	Unstructured/ Prices/Returns/ Factors	Unstructured/ Non Linearity / Hidden Structure	Non Stationarity / Overfitting / Optimization
Memory Networks	Prices/ Returns/ Factors	Classification and Regression	Forecasting / Explanation	Unstructured/ Prices/Returns/ Factors	Non Linearity / Cycles and Regimes	Non Stationarity / Overfitting / Optimization
Convolutional Networks	Prices/ Returns/ Factors	Classification and Regression	Forecasting / Explanation	Unstructured/ Prices/Returns/ Factors	Non Linearity / Cycles and Regimes	Non Stationarity / Overfitting / Optimization
Auto-Encoders	Covariance	Dimension Reduction	Forecasting / Explanation	Non-Linear "PCA"	Non Linear DEpendencies	Estimation / Learning