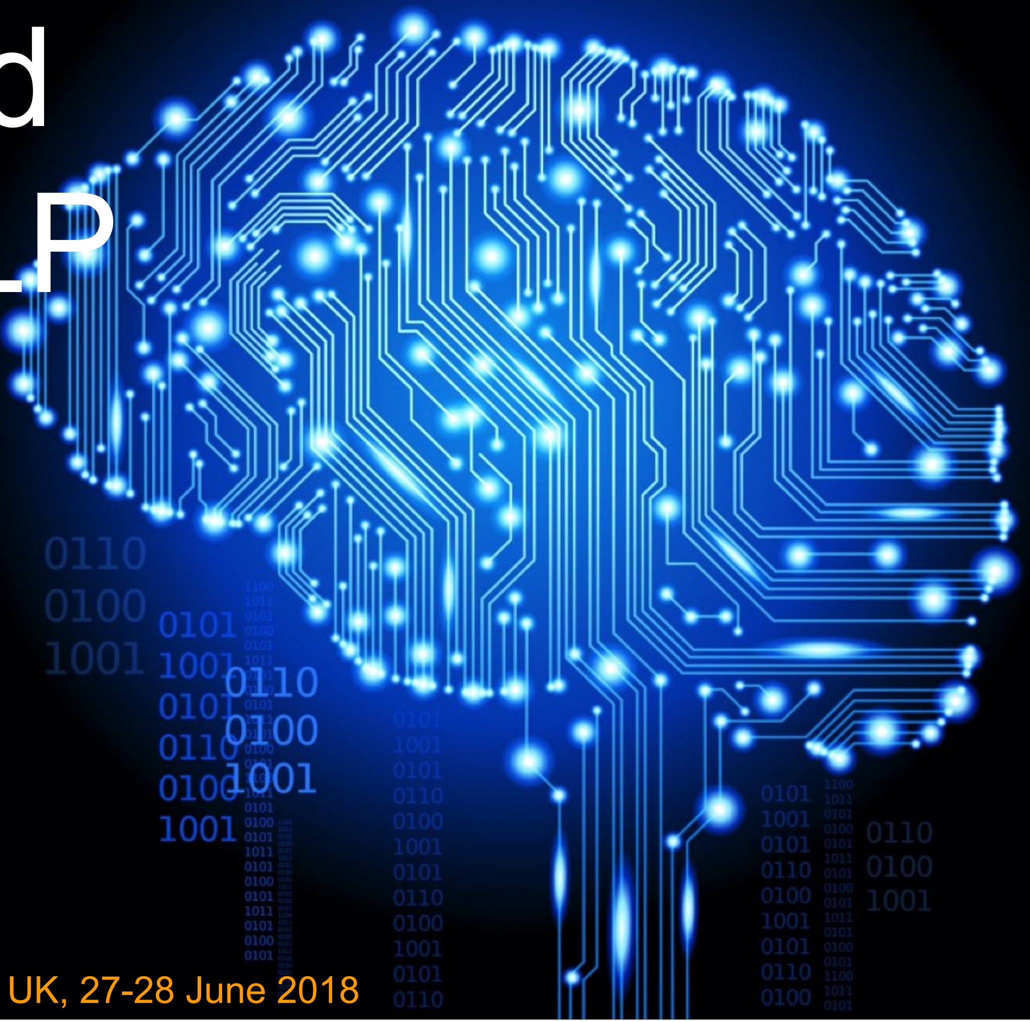
FinTech and the ML/AI/NLP Revolution

Sanjiv R. Das
Santa Clara University

http://srdas.github.io/Papers/fintech.pdf



London UK, 27-28 June 2018

What is FinTech?

- FinTech refers to various financial technologies used to automate processes in the financial sector, from routine, manual tasks to non-routine, cognitive decision-making.

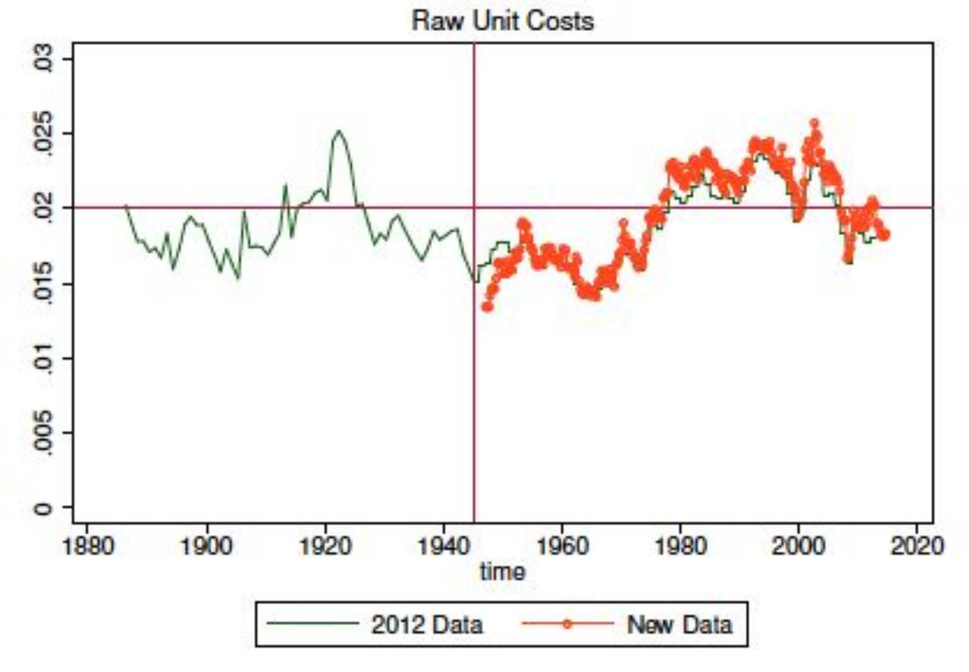
- FinTech may be characterized by technological change in three broad areas of finance:
    1. raising capital,
    2. allocating and investing capital,
    3. transferring capital.

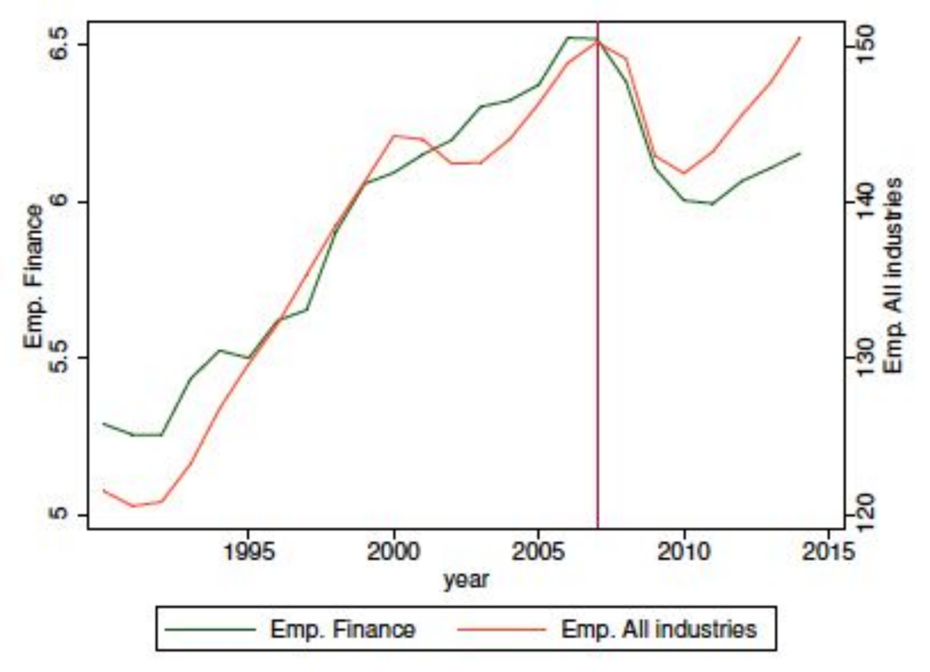
- Definition: "FinTech is any technology that eliminates or reduces the costs of the middleman in finance."

There is now a growing interest and literature: - http://lfe.mit.edu/research/fintech/

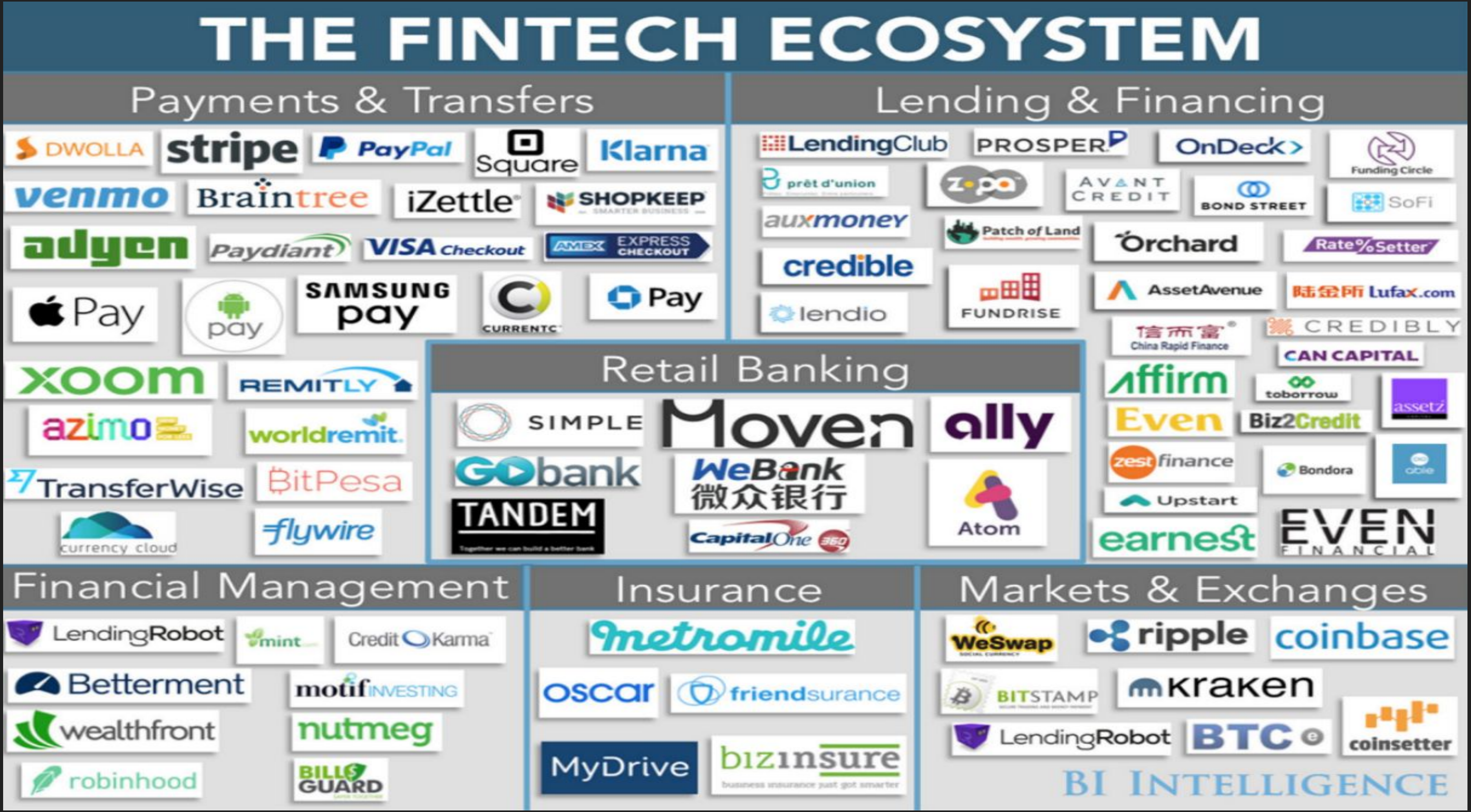
- Risk and Risk Management in the Credit Card Industry (Florentin Butaru, Qingqing Chen, Brian Clark, Sanmay Das, Andrew Lo, Akhtar Siddique), *Journal of Banking and Finance* 72(2016), 218–239.

The Costs of Financial Intermediation





Philippon (2016)



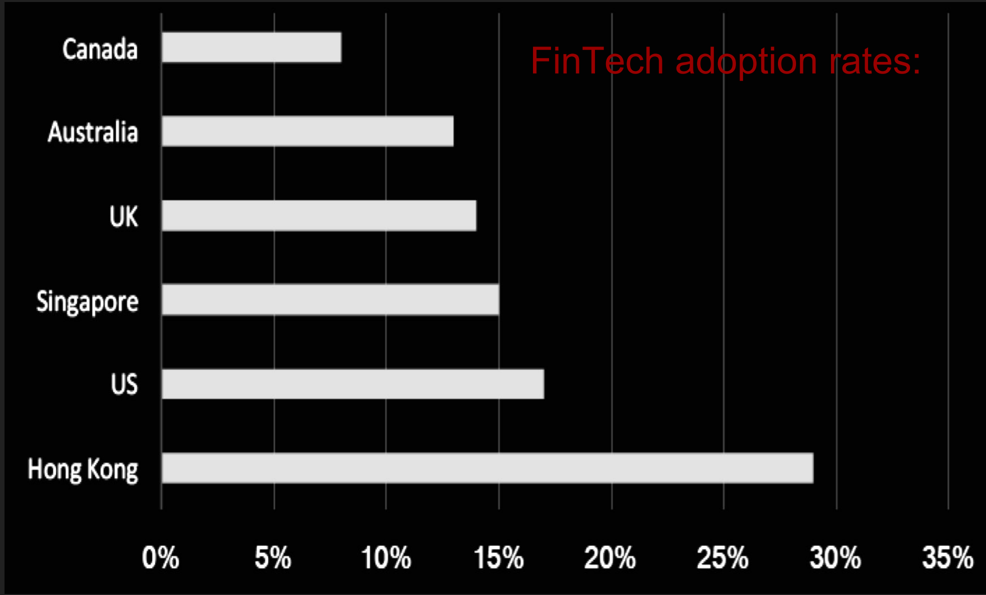
THE FINTECH ECOSYSTEM
Payments & Transfers
DWOLLA
stripe
PayPal
Square
Klarna
venmo
Braintree
iZettle
SHOPKEEP
adyen
Paydiant
VISA Checkout
AMEX EXPRESS CHECKOUT
Pay
pay
SAMSUNG pay
CURRENTC
Pay
xoom
REMITLY
azimo
worldremit
TransferWise
BitPesa
currency cloud
flywire
Lending & Financing
LendingClub
PROSPER
OnDeck
Funding Circle
prêt d'union
zopa
AVANT CREDIT
BOND STREET
SoFi
auxmoney
Patch of Land
Orchard
RateSetter
credible
FUNDRISE
AssetAvenue
陆金所 Lufax.com
lendio
信而富 China Rapid Finance
CREDIBLY
CAN CAPITAL
affirm
toborrow
assetz
Even
Biz2Credit
zest finance
Bondora
Upstart
earnest
EVEN FINANCIAL
Retail Banking
SIMPLE
Moven
ally
GObank
WeBank 微众银行
TANDEM
Atom
Capital One 360
Financial Management
LendingRobot
mint
Credit Karma
Betterment
motif INVESTING
wealthfront
nutmeg
robinhood
BILLGUARD
Insurance
metromile
oscar
friendsurance
MyDrive
bizinsure
Markets & Exchanges
WeSwap
ripple
coinbase
BITSTAMP
kraken
LendingRobot
BTC
coinsetter
BI INTELLIGENCE

FinTech Landscape

- 2017, Q1: Over 100 FinTech startups with $3.2 billion in funding. https://assets.kpmg.com/content/dam/kpmg/xx/pdf/2017/04/pulse-of-fintech-q1.pdf

- 1400 FinTech companies with $33 billion in funding.

- Losses from credit card fraud are $31 billion a year.

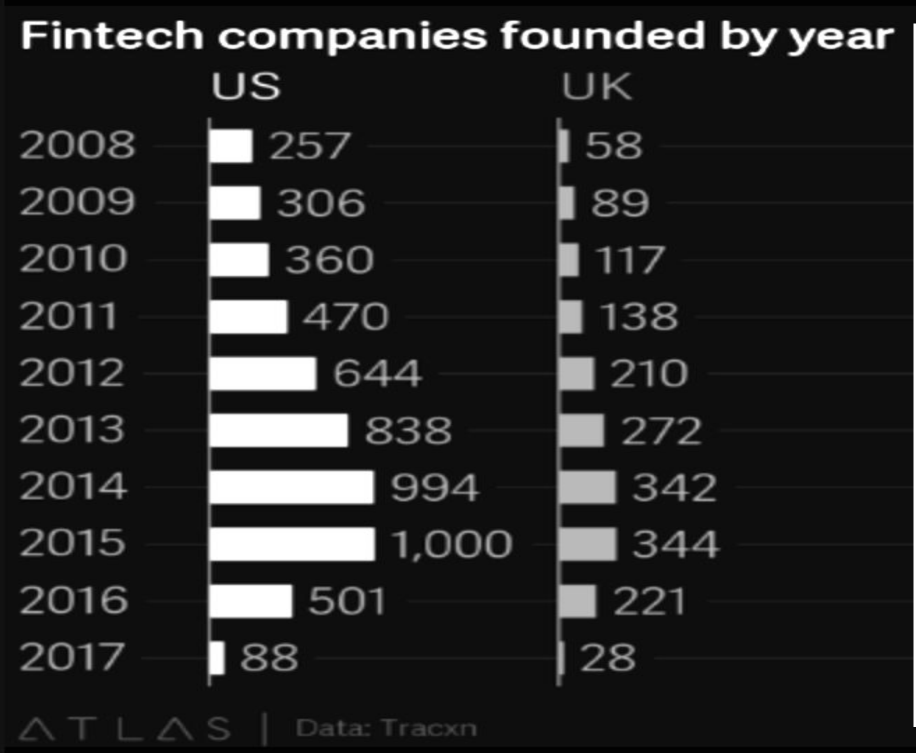


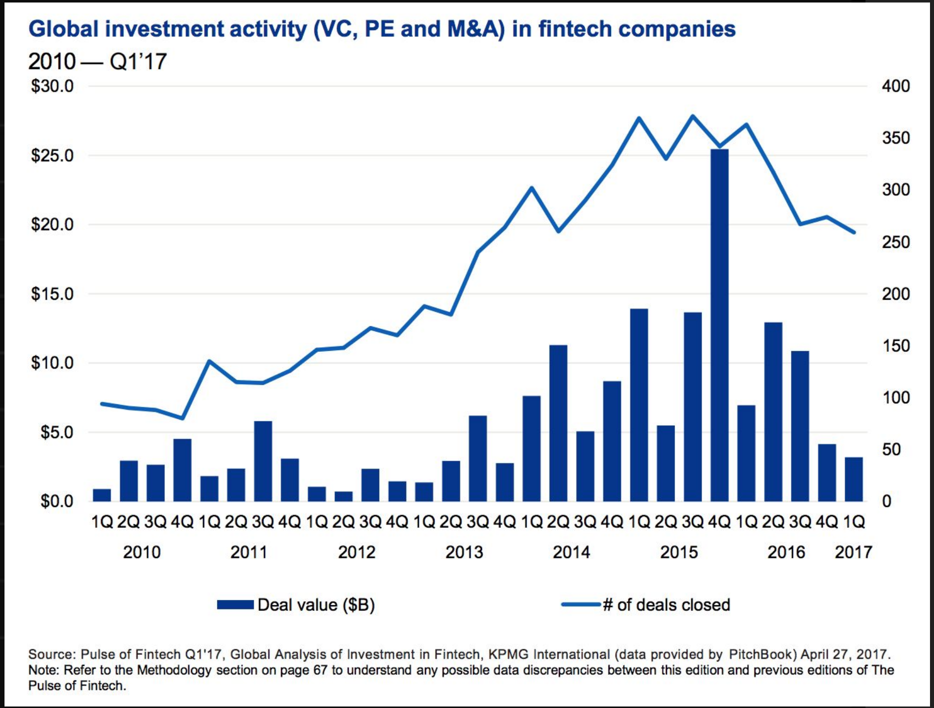
“Using Big Data to Detect Financial Fraud Aided by FinTech Methods” - S. Srinivasan, Texas Southern U.

FinTech Startups by Year

Sources: Atlas https://www.newconstructs.com/big-banks-will-win-the-fintech-revolution/
https://www.accenture.com/_acnmedia/PDF-57/Accenture-Fintech-Did-Someone-Cancel-The-Revolution.pdf

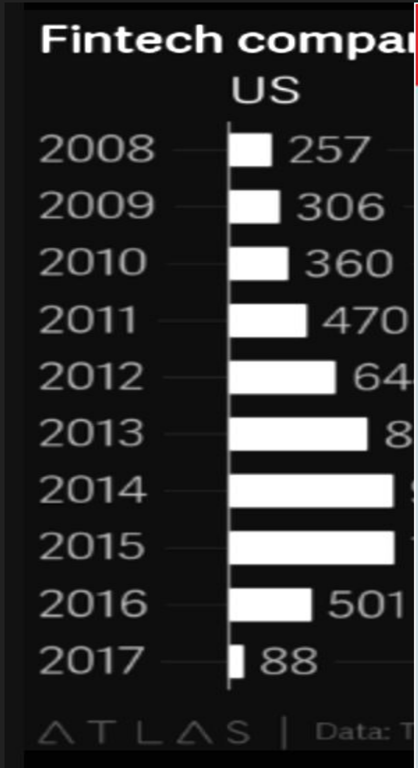
FinTech Startups by Year



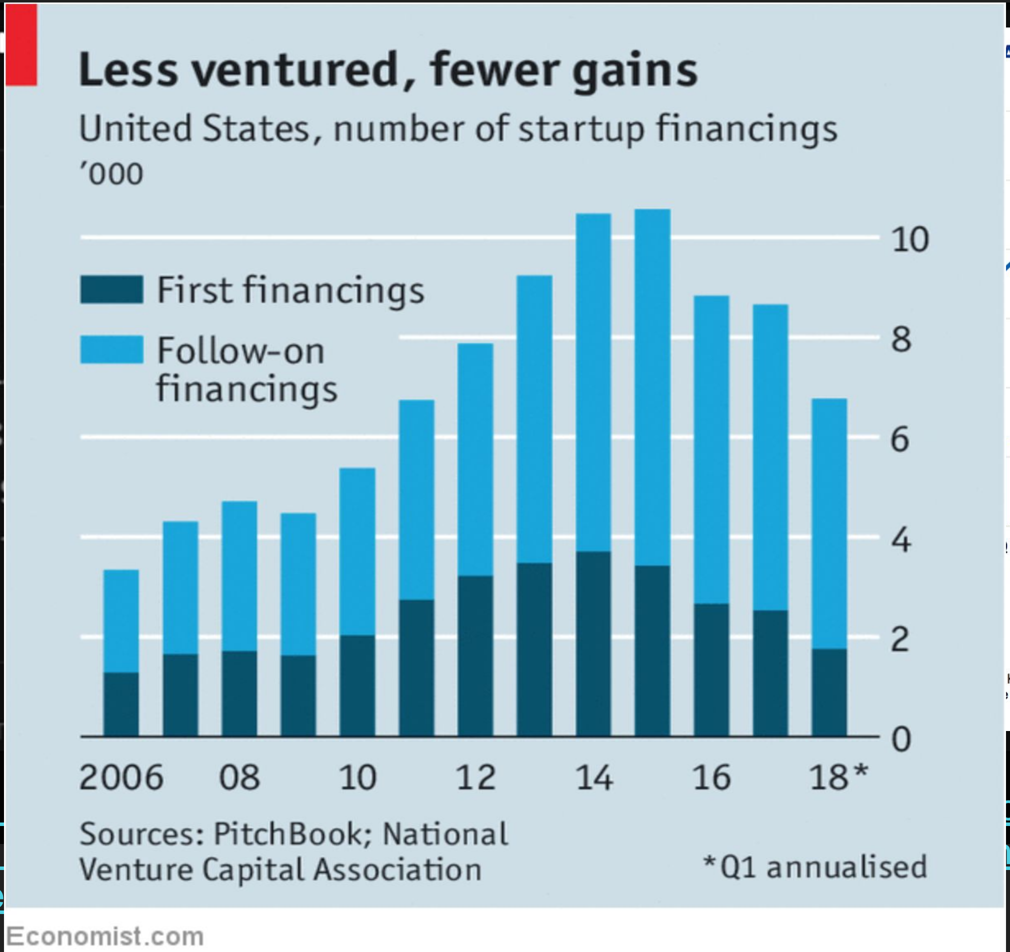


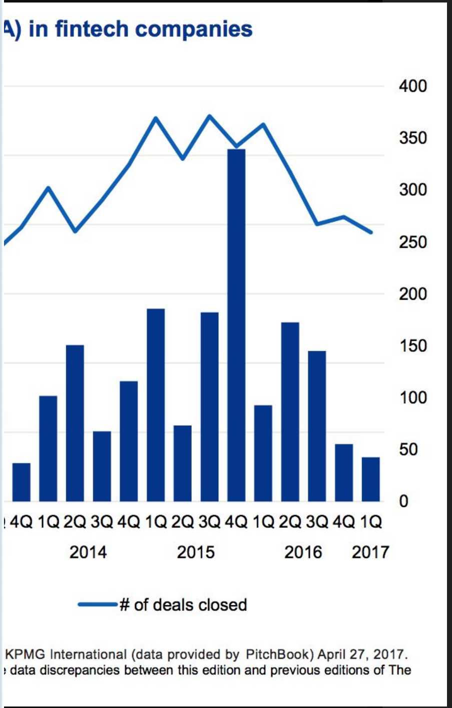
Sources: Atlas https://www.newconstructs.com/big-banks-will-win-the-fintech-revolution/
https://www.accenture.com/_acnmedia/PDF-57/Accenture-Fintech-Did-Someone-Cancel-The-Revolution.pdf

FinTech Startups by Year



Sources: Atlas





https ... olution/
https ... -Did-Someone-Cance
l-The

FinTech Framework

1. Machine Learning, AI, and Deep Learning.
2. Network Models.
3. Personal and Consumer Finance.
4. Nowcasting.
5. Cybersecurity.
6. Fraud Detection.
7. Payment and Funding Systems.
8. Automated and High-Frequency Trading.
9. Blockchain and Cryptocurrencies.
10. Text Analytics.

Examples

- Monitoring corporate buzz.
- Analyzing data to detect, analyze, and understand the more profitable customers or products.
- Targeting new clients.
- Customer retention.
- Lending activity (automated)
- Market prediction and trading.
- Risk management.
- Automated financial analysts.
- Financial forensics to prevent rogue employees.
- Credit cards: optimizing use, marketing offers.
- Fraud detection.
- Detecting market manipulation.
- Social network analysis of clients.
- Measuring institutional risk from systemic risk.

This is implicit : Banks will soon be technology companies and will need to invest heavily in R&D Tech

Definition of AI

- Intelligence exhibited by machines
- Narrow or Weak AI: "Expert systems that match or exceed human intelligence in a narrowly defined area, but not in broader areas" (Dvorsky G., 2013) e.g. Siri.
- Artificial General Intelligence: An artificial neural network not preprogrammed with fixed rules. Rewire itself to reflect patterns in the data, adaptable to its environment, in which (hopefully) advanced skills emerge organically.
- "Humans don't learn to understand language by memorizing dictionaries and grammar books, so why should we possibly expect our computers to do so?" (LEWIS-KRAUS G, 2016).
- And, Super AI?
- Rule-based AI can never be more intelligent than its creators, but data-driven AI can! http://io9.gizmodo.com/how-much-longer-before-our-first-ai-catastrophe-464043243 (Dvorsky G., 2013)https://mobile.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html?_r=0&referer= (LEWIS-KRAUS G, 2016)

Customization: Conversational AI

NarrativeScience

TalkIQ

TOPBOTS

CHORUS

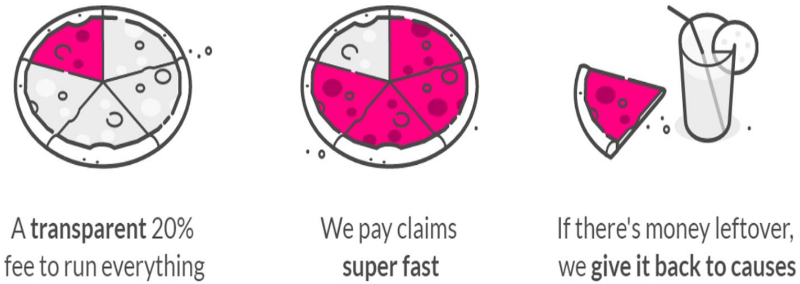
TROOPS

conversica

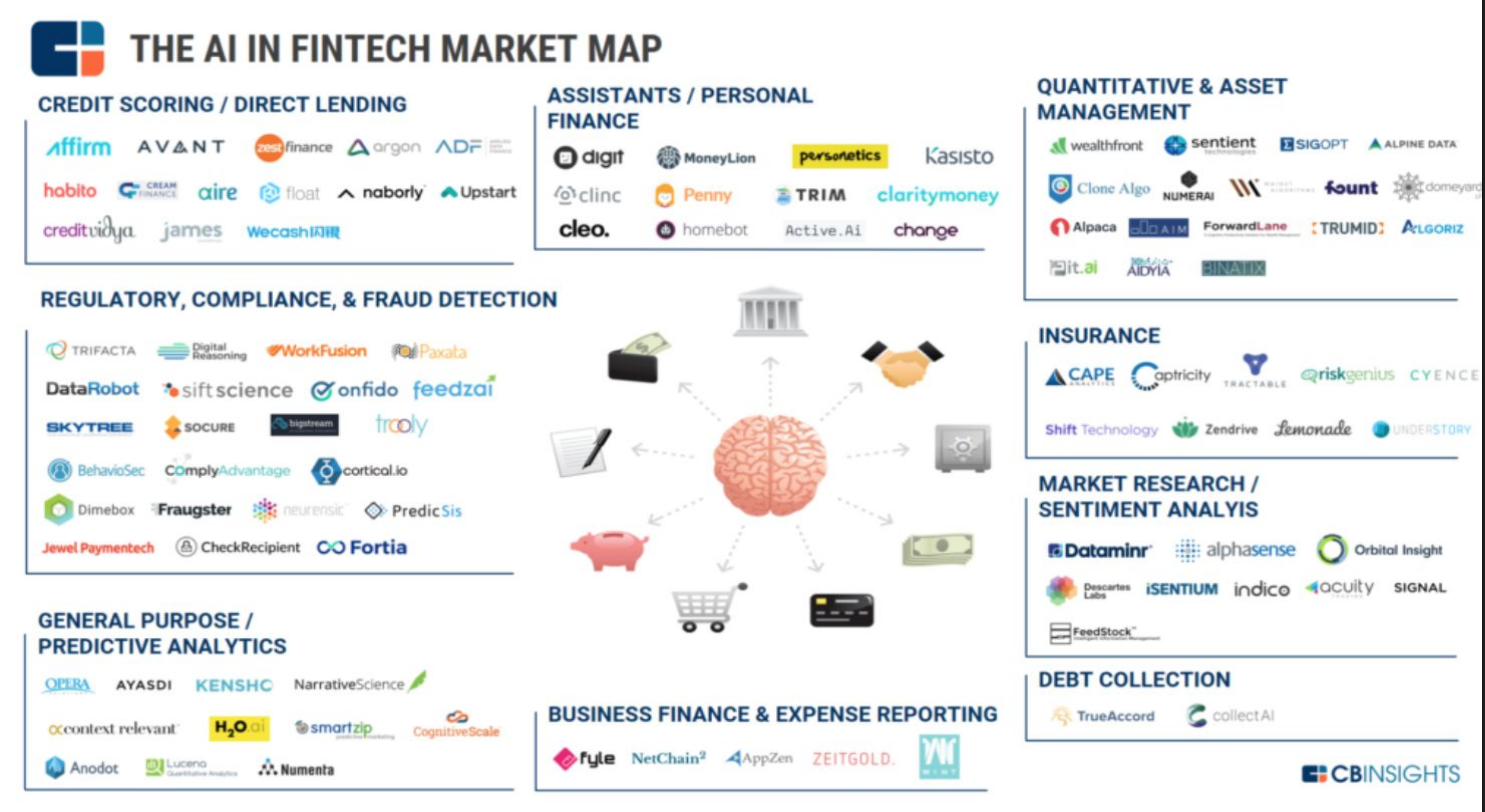
Insurance

Lemonade

- Improving fairness of the insurance industry through Maya, their artificial intelligence bot
- Social Impact is the legal mission and business model
- Faster, easier



https://www.lemonade.com/


THE AI IN FINTECH MARKET MAP
CREDIT SCORING / DIRECT LENDING
affirm
AVANT
zestfinance
argon
ADF
habito
CREAM FINANCE
aire
float
naborly
Upstart
creditvidya
james
Wecash闪银
ASSISTANTS / PERSONAL FINANCE
digit
MoneyLion
personetics
Kasisto
clinc
Penny
TRIM
claritymoney
cleo.
homebot
Active.Ai
change
QUANTITATIVE & ASSET MANAGEMENT
wealthfront
sentient
SIGOPT
ALPINE DATA
Clone Algo
NUMERAI
fount
domeyard
Alpaca
AIM
ForwardLane
TRUMID
ALGORIZ
it.ai
AIDYIA
BINATIX
REGULATORY, COMPLIANCE, & FRAUD DETECTION
TRIFACTA
Digital Reasoning
WorkFusion
Paxata
DataRobot
siftscience
onfido
feedzai
SKYTREE
SOCURE
bigstream
trooly
BehavioSec
ComplyAdvantage
cortical.io
Dimebox
Fraugster
neurensic
PredicSis
Jewel Paymentech
CheckRecipient
Fortia
INSURANCE
CAPE ANALYTICS
captricity
TRACTABLE
riskgenius
CYENCE
Shift Technology
Zendrive
Lemonade
UNDERSTORY
MARKET RESEARCH / SENTIMENT ANALYIS
Dataminr
alphasense
Orbital Insight
Descartes Labs
iSENTIUM
indico
acuity
SIGNAL
FeedStock
DEBT COLLECTION
TrueAccord
collectAI
GENERAL PURPOSE / PREDICTIVE ANALYTICS
OPERA
AYASDI
KENSHO
NarrativeScience
context relevant
H2O.ai
smartzip
CognitiveScale
Anodot
Lucena
Numenta
BUSINESS FINANCE & EXPENSE REPORTING
fyle
NetChain
AppZen
ZEITGOLD.
CBINSIGHTS

Deep Learning in Finance

- Bridgewater Associates: World’s largest hedge fund has a project to automate decision-making to save time and eliminate human emotion volatility.

- Goldman Sachs: Two out of the 600 equity traders left. Found that four traders can be replaced by one computer engineer.

Deep Learning in Finance

- Transactions: By 2020 at least five percent of all economic transactions will be handled by autonomous software. AI will process payment functions and learn from customer behaviours, through Intelligent Payment Management (IPM).

- Savings: AI will help consumers make daily financial decisions and monitor spending. New Personal Financial Management apps use contextual awareness, which measures spending habits and online footprints to create personalised advice. Combining pooled financial data with end-user control to offer tailor-made services is a classic AI solution.

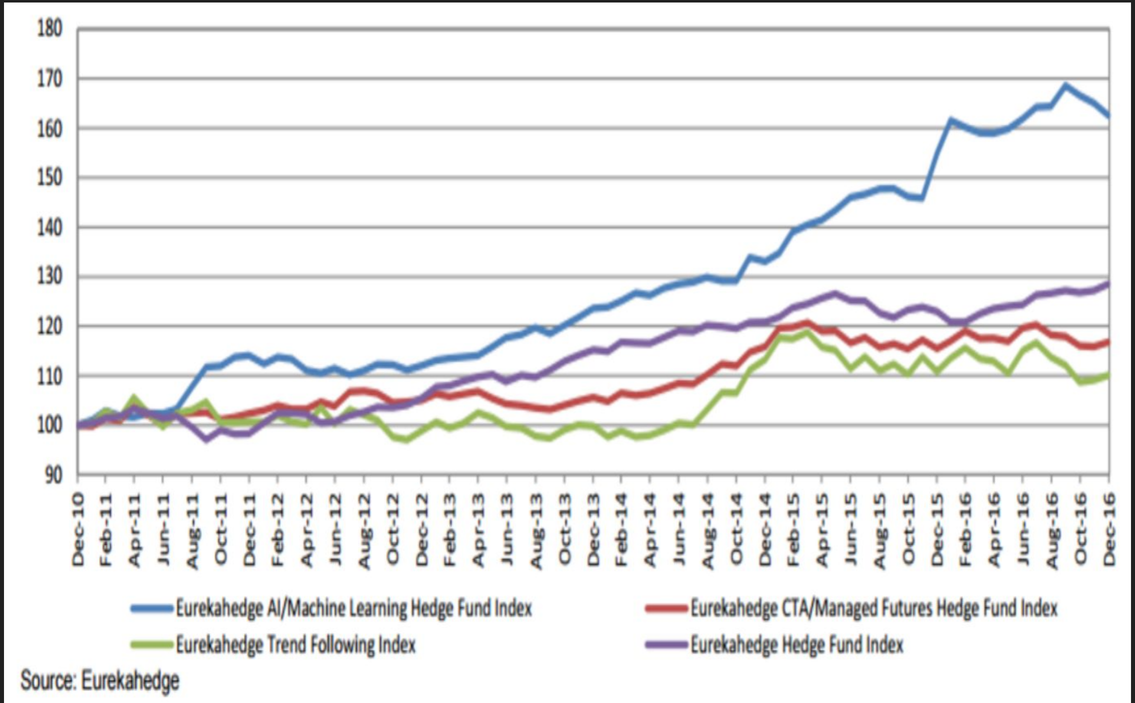
Deep Learning in Finance

- Mizuho Financial Group sent Pepper, its humanoid robot into its Tokyo branch to handle customer inquiries. Partnering with IBM to enable Pepper to understand human emotions, and build interaction into apps.

- RBS is trialing Luvo AI, a customer service assistant to interact with staff and customers.

- Fraud prevention: Mining user data to detect abnormal behavior, anomalies, and unusual transactions.

Hedge Funds use Machine Learning



Blackrock: replacing human stock pickers with machine algorithms.

Sentient Inc: Hedge fund run entirely using AI. Secret algo with adaptive learning. Uses thousands of machines.

Numerai: Hedge fund makes trades by aggregating trading algorithms submitted by anonymous contributors, prizes are awarded in cryptocurrency.

Emma: Evolved a hedge fund using a software that writes news articles.

Very little data about the track record of these hedge funds, as the business remains secretive.

Investor reluctance to turn over money completely to a machine.

Financial Automation

BloombergMarkets

JPMorgan Software Does in Seconds What Took Lawyers 360,000 Hours

JPMorgan Software Does in Seconds What Took Lawyers 360,000 Hours

by **Hugh Son**

February 27, 2017, 4:31 PM PST *Updated on* February 28, 2017, 4:24 AM PST

At JPMorgan Chase & Co., a learning machine is parsing financial deals that once kept legal teams busy for thousands of hours.

The program, called COIN, for Contract Intelligence, does the mind-numbing job of interpreting commercial-loan agreements that, until the project went online in June, consumed 360,000 hours of work each year by lawyers and loan officers. The software reviews documents in seconds, is less error-prone and never asks for vacation.

Deep Learning: A Quick Introduction

In finance, deep learning is used to add value using pattern recognition.

This evolution is similar to that in medicine, image processing, and autonomous vehicles.

Learning the Black-Scholes Equation (Culkin & Das, 2017)

THE JOURNAL OF FINANCE • VOL. XLIX, NO. 3 • JULY 1994

A Nonparametric Approach to Pricing and Hedging Derivative Securities Via Learning Networks

JAMES M. HUTCHINSON, ANDREW W. LO, and TOMASO POGGIO*

ABSTRACT

We propose a nonparametric method for estimating the pricing formula of a derivative asset using learning networks. Although not a substitute for the more traditional arbitrage-based pricing formulas, network-pricing formulas may be more accurate and computationally more efficient alternatives when the underlying asset's price dynamics are unknown, or when the pricing equation associated with the no-arbitrage condition cannot be solved analytically. To assess the potential value of network pricing formulas, we simulate Black-Scholes option prices and show that learning networks can recover the Black-Scholes formula from a two-year training set of daily options prices, and that the resulting network formula can be used successfully to both price and delta-hedge options out-of-sample. For comparison, we estimate models using four popular methods: ordinary least squares, radial basis function networks, multilayer perceptron networks, and projection pursuit. To illustrate the practical relevance of our network pricing approach, we apply it to the pricing and delta-hedging of S&P 500 futures options from 1987 to 1991.

```
from scipy.stats import norm
def BSM(S,K,T,sig,rf,dv,cp):   #cp = {+1.0 (calls), -1.0 (puts)}
    d1 = (math.log(S/K)+(rf-dv+0.5*sig**2)*T)/(sig*math.sqrt(T))
    d2 = d1 - sig*math.sqrt(T)
    return cp*S*math.exp(-dv*T)*norm.cdf(d1*cp) - cp*K*math.exp(-rf*T)*nor

df = pd.read_csv('/Users/srdas/GoogleDrive/Papers/DeepLearning/DLinFinance
```

Normalizing spot and call prices

C is homogeneous degree one, so

$$aC(S,K) = C(aS,aK)$$

This means we can normalize spot and call prices and remove a variable by dividing by K.

$$\frac{C(S,K)}{K} = C(S/K,1)$$

Data, libraries, activation functions

Slide Type Fragment

```
n = 300000
n_train =  (int)(0.8 * n)
train = df[0:n_train]
X_train = train[['Stock Price', 'Maturity', 'Dividends', 'Volatility', 'R:
y_train = train['Call Price'].values
test = df[n_train+1:n]
X_test = test[['Stock Price', 'Maturity', 'Dividends', 'Volatility', 'Risl
y_test = test['Call Price'].values
```

Slide Type Fragment

```
#Import libraries
from keras.models import Sequential
from keras.layers import Dense, Dropout, Activation, LeakyReLU
from keras import backend

def custom_activation(x):
    return backend.exp(x)
```

Fit the Model

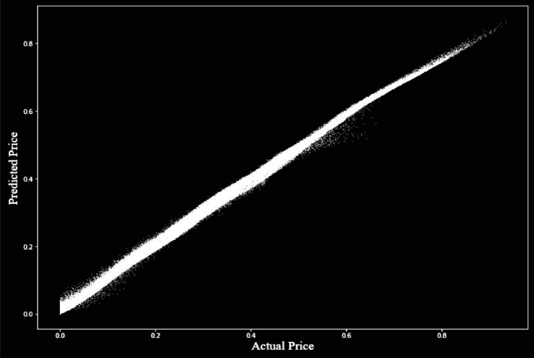
```
nodes = 120
model = Sequential()

model.add(Dense(nodes, input_dim=X_train.shape[1]))
model.add(LeakyReLU())
model.add(Dropout(0.25))

model.add(Dense(nodes, activation='elu'))
model.add(Dropout(0.25))

model.add(Dense(nodes, activation='relu'))
model.add(Dropout(0.25))

model.add(Dense(nodes, activation='elu'))
model.add(Dropout(0.25))

model.add(Dense(1))
model.add(Activation(custom_activation))

model.compile(loss='mse',optimizer='rmsprop')

model.fit(X_train, y_train, batch_size=64, epochs=10, validation_split=0.
```

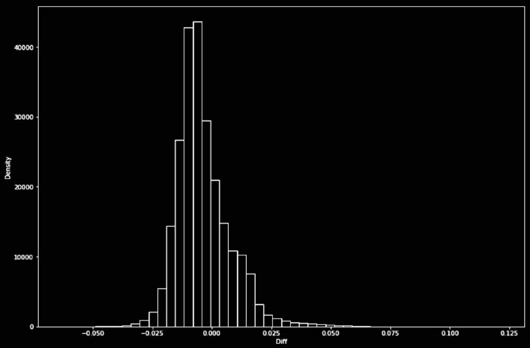
```
Train on 216000 samples, validate on 24000 samples
Epoch 1/10
22s - loss: 0.0051 - val_loss: 0.0013
Epoch 2/10
20s - loss: 0.0013 - val_loss: 1.3056e-04
Epoch 3/10
21s - loss: 0.0010 - val_loss: 7.0662e-04
Epoch 4/10
20s - loss: 8.4915e-04 - val_loss: 2.9489e-04
Epoch 5/10
21s - loss: 7.4240e-04 - val_loss: 6.5513e-04
Epoch 6/10
21s - loss: 6.7416e-04 - val_loss: 4.9628e-04
Epoch 7/10
21s - loss: 6.3666e-04 - val_loss: 3.7541e-04
Epoch 8/10
21s - loss: 5.9715e-04 - val_loss: 1.2603e-04
Epoch 9/10
26s - loss: 6.0532e-04 - val_loss: 1.3601e-04
Epoch 10/10
29s - loss: 6.1739e-04 - val_loss: 1.3957e-04

<keras.callbacks.History at 0x10ba4cef0>
```

In-Sample

```
y_train_hat = model.predict(X_train)
#reduce dim (240000,1) -> (240000,) to match y_train's dim
y_train_hat = squeeze(y_train_hat)
CheckAccuracy(y_train, y_train_hat)
```

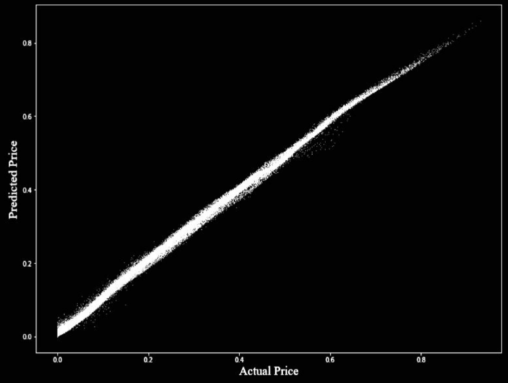
```
Mean Squared Error:       0.000141080530002
Root Mean Squared Error:  0.0118777325278
Mean Absolute Error:      0.00954217479875
Mean Percent Error:       0.0444015623019
```

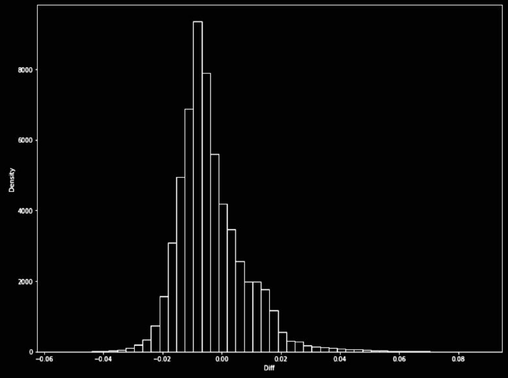




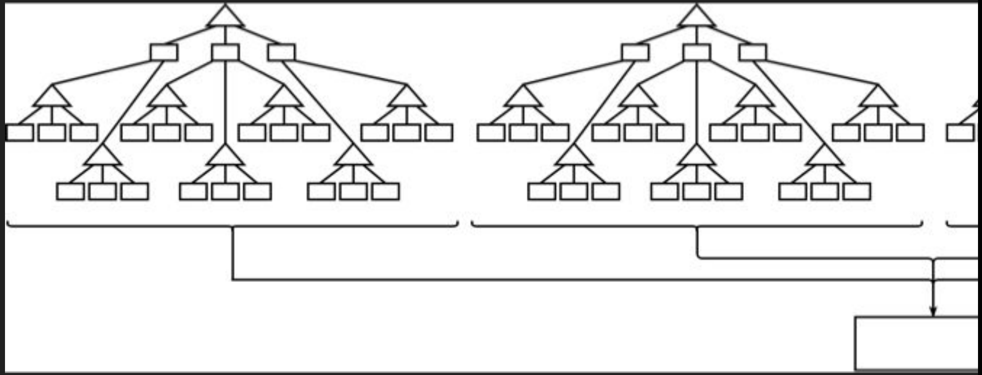
Out-of-Sample

```
Mean Squared Error:       0.00014218354892
Root Mean Squared Error:  0.0119240743423
Mean Absolute Error:      0.00958741885268
Mean Percent Error:       0.0446387882149
```





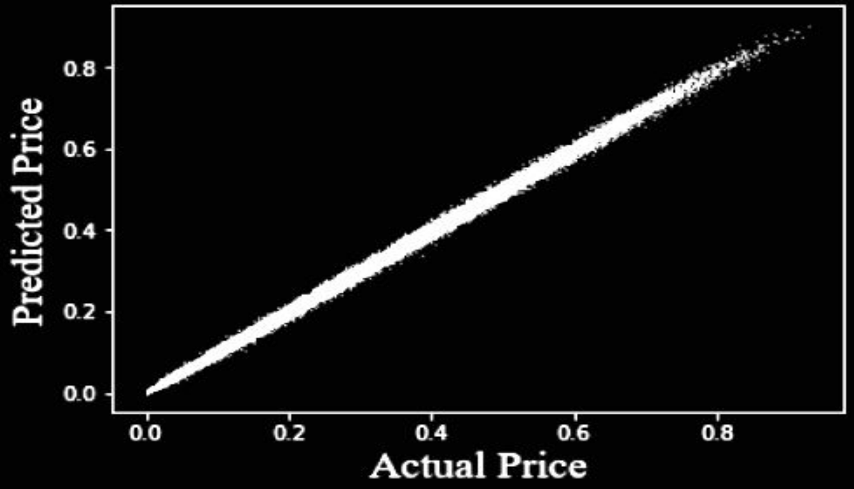
Random Forest

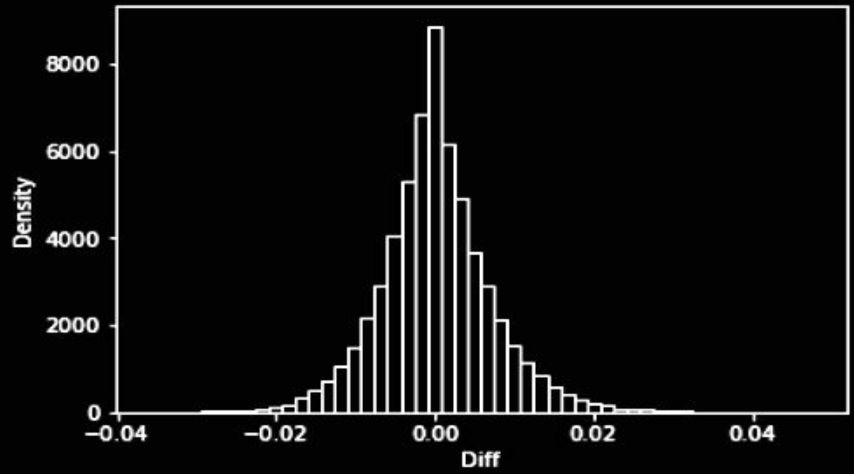


```
from sklearn.ensemble import RandomForestRegressor

forest = RandomForestRegressor()
forest = forest.fit(X_train, y_train)
y_test_hat = forest.predict(X_test)
```

```
stats = CheckAccuracy(y_test, y_test_hat)
```

```
Mean Squared Error:         4.97733713185e-05
Root Mean Squared Error:    0.00705502454415
Mean Absolute Error:        0.00516501148658
Mean Percent Error:         0.0264110854593
```





Investing in AI

QUOTES & COMPANIES

VIEW ALL COMPANIES

AI Powered Equity ETF

AIEQ (U.S.: NYSE Arca)

REAL TIME 3:29 PM EST 11/06/17

$24.7101 USD

0.1101 0.45% ▲

Volume
164,231

65 Day Avg Vol
420,342

1 Day Range
24.55 - 24.707

52 Week Range
24.50 - 25.99
(11/03/17 - 10/23/17)

1 D | 5 D | 1 M | 3 M | YTD | 1 Y | 3 Y | $ | %

ADVANCED CHARTING

COMPARE

Open **24.55** Prior Close **24.60** (11/03/17)

1 Day **AIEQ 0.45%** ▲ **DJIA 0.13%** ▲ **S&P 500 0.20%** ▲ **Nasdaq 0.37%** ▲

OVERVIEW

ALL SECTIONS

Investment Information AIEQ

POWERED BY LIPPER

Category	Multi-Cap Growth	Net Expense Ratio *	
Style	Growth	Turnover %	.00

* Expense ratio updated annually from fund's year-end report.

Investment Policy

The Fund seeks capital appreciation. The Fund invests primarily in US exchange listed equity securities based on the results of a quantitative model which identifies approximately 30 to 70 companies with the greatest potential over the next 12 months for appreciation and their corresponding weights.

FUND DETAILS

Net Assets
70.90 M

NAV
$24.65 (11/03/17)

Shares Outstanding
N/A

Yield
N/A

Latest Dividend

Predicting the Direction of the S&P 500

Das, Mokashi, Culkin (2017)

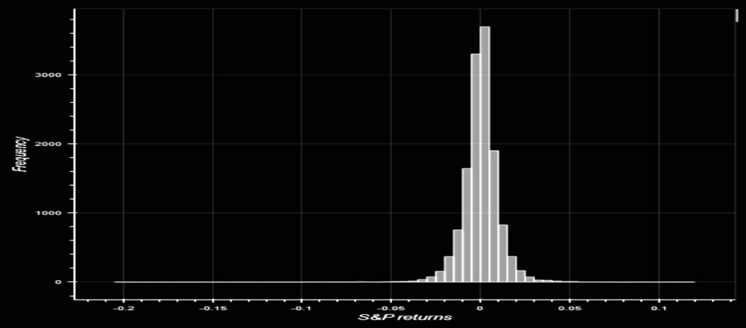
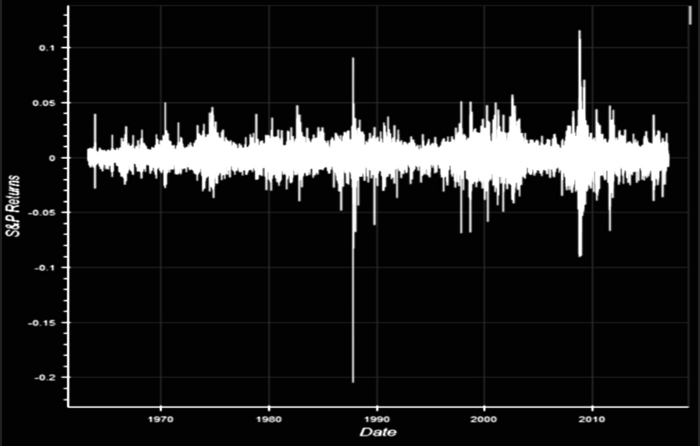


Figure 2: The distribution of daily S&P 500 index returns from 1963-2016. The mean return is 0.00031 and the standard deviation is 0.01. The skewness and kurtosis are −0.62 and 20.68, respectively.



Data, lookback, lookforward

```
n = 30                    #Lookback period
nfwd = 30                 #Look ahead forecast period
train_size = 10000        #Training size

m = shape(df)[0]
print(m) #Number of rows

df2 = df[n:m][df.columns[[0,1,21]]]
print(shape(df2))
df2.head()
```

```
13532
(13502, 3)
```

	date	SPX	Sign
30	1963-05-14	-0.003831	0.0
31	1963-05-15	0.003133	1.0
32	1963-05-16	-0.002556	0.0
33	1963-05-17	0.000569	1.0
34	1963-05-20	-0.004695	0.0

Organize data and fit the model

```
## Fit the model to the first 10,000 rows of data
X_train = df2[x:x+train_size].as_matrix()
X_train = X_train[:,3:len(df2.columns)]
print(shape(X_train))
Y_train = df2[x:x+train_size].as_matrix()
Y_train = Y_train[:,2:3].astype(int32)
print(shape(Y_train))

data_dim = shape(X_train)[1]
print(data_dim)

X_test = df2[x+train_size:(x+trai
X_test = X_test[:,3:len(df2.colum
print(shape(X_test))
Y_test = df2[x+train_size:(x+trai
Y_test = Y_test[:,2:3].astype(int
print(shape(Y_test))
```

```
(10000, 570)
(10000, 1)
570
(30, 570)
(30, 1)
```

```
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from keras.layers.advanced_activations import LeakyReLU
from keras.utils import to_categorical

Y_train2 = to_categorical(Y_train,2)
```

```
model = Sequential()
n_units = 64 #200

model.add(Dense(n_units, input_dim=data_dim))
model.add(LeakyReLU())
model.add(Dropout(0.25))
```

```
               metrics=['accuracy'])
bsize = 32
model.fit(X_train, Y_train2, batch_size=bsize, epochs=25, validation_
```

```
Train on 9000 samples, validate on 1000 samples
Epoch 1/25
2s - loss: 0.6939 - acc: 0.5124 - val_loss: 0.6924 - val_acc: 0.5180
Epoch 2/25
2s - loss: 0.6935 - acc: 0.5171 - val_loss: 0.6899 - val_acc: 0.5430
Epoch 3/25
```

Model Accuracy

$$parameters = 570 \times 65 + 64 \times 65 + 64 \times 65 + 65 = 45435$$

```
model.evaluate(X_train, Y_train2, verbose=0)
```

```
[0.68329057350158695, 0.55654999999999999]
```

Two Curses of Predictive Analytics

Non-stationarity

- Strong: Joint distribution of all variables (Y,X) remains the same over time.
- Weak: only mean and autocorrelation need to be same over time. If we are predicting the first moment, then this works.

Randomness

- If noise swamps the signal, then we get poor predictions.

Three Sets of Experiments

- In-sample: 5,000 observations, (i) randomly chosen set of 10 observations from the training sample, (ii) randomly chosen set of 30 observations, (iii) 1000 observations; (iv) the entire training sample. Roll forward 20 days and repeat this experiment. 423 experiments.
- Stationary out-of-sample: (i) 4990 observations for training, 10 for testing; (ii) 4970 training, 30 testing; (iii) 4000, 1000. Roll forward 20 days, rinse and repeat this experiment.
- Non-stationary out-of-sample: 5,000 observations for training, (i) test on the next 10; (ii) next 30; (iii) next 1000. Rinse & repeat.

Metrics:

- Overall accuracy (OA)
- Forecast period average accuracy (FPAA)

Results using h2o.ai

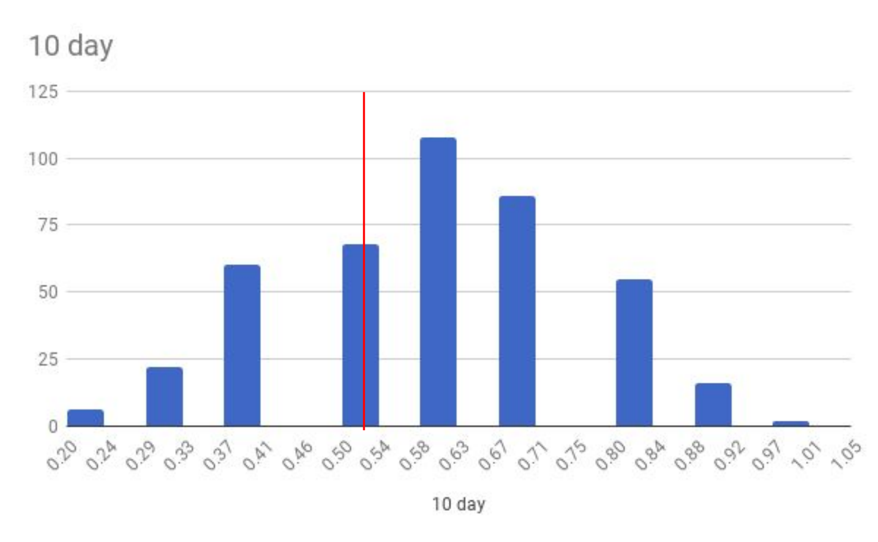
OA = %age correct predictions

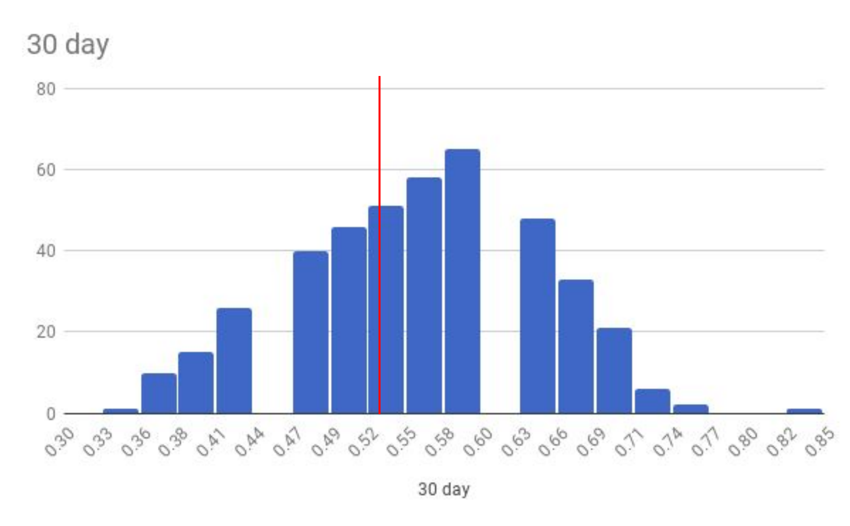
FPAA = %age rolling periods with >50% accuracy

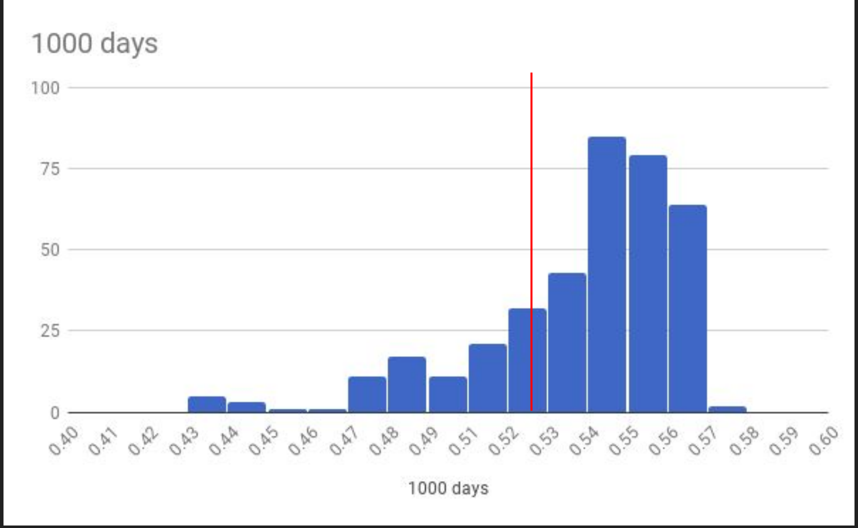
Table 1: Accuracy levels for three cases of experiments on stock market predictability. We report the overall accuracy (OA) and forecast period average accuracy ($FPAA$). The first number in each cell is the OA and the second number is the $FPAA$, computed at a threshold of 50%, i.e., $FPAA = 1$ if the percentage of correct forecasts in the prediction period (F) is greater than 0.5, else $FPAA = 0$. We may also compute FPAA for a threshold of 0.527, i.e., the baseline percentage of times the stock market rises, but we get identical results for $F = 10, 30$ and slightly lower values for $F = 1000$.

Experimental case	$F = 10$ (OA, FPAA)	$F = 30$ (OA, FPAA)	$F = 1000$ (OA, FPAA)	$F = 5000$ (OA, FPAA)
A_1. In-sample (IS)	(0.604,0.662)	(0.565,0.693)	(0.525,0.905)	(0.522,0.929)
A_2. Stationary (OS)	(0.593,0.624)	(0.551,0.652)	(0.526,0.896)	—
A_3. Nonstationary (NS)	(0.594,0.631)	(0.558,0.674)	(0.535,0.888)	—

Non-stationary out-of-sample





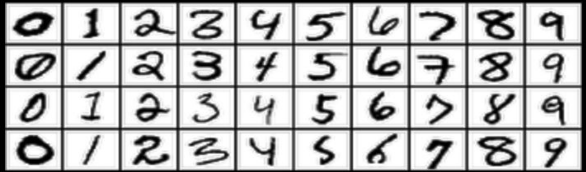


Most of the prediction error comes from randomness.

Numerai Follow
A new kind of hedge fund built by a network of data scientists.
Jan 7, 2016 · 7 min read

Encrypted Data For Efficient Markets

By the end of this article, you'll understand how Numerai is using advances in structure-preserving encryption to allow for open participation in the problem of stock market efficiency.



Some MNIST Handwritten Digits

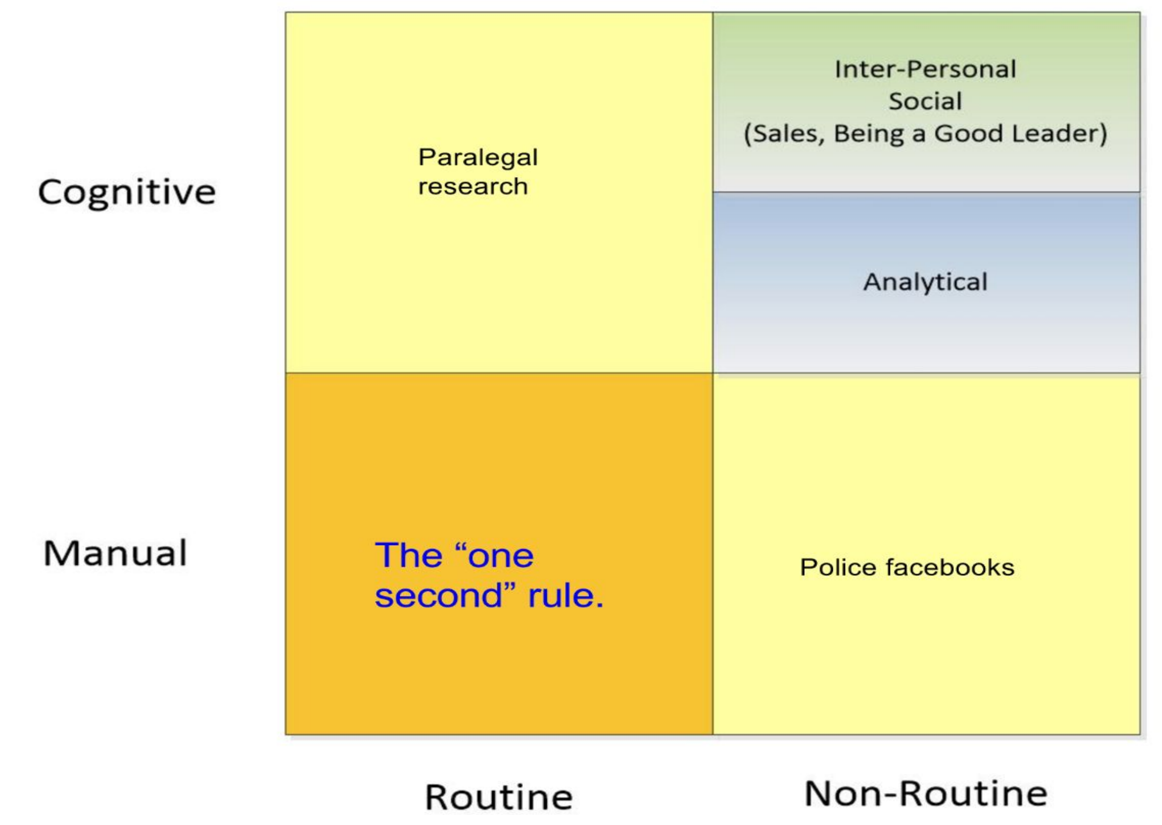
Over the last few years, machine learning algorithms solved big problems in computer vision. One such problem was getting an algorithm to learn how to recognize handwritten digits in the MNIST dataset. Everyone writes digits differently, so the problem was difficult for computers to grasp.

Other applications of deep learning in finance

- Forecasting the VIX curve.
- Predicting Credit Card Default, see Khandani, Lee, Lo (2016).
- Autoencoders for generating new factors
- Portfolio optimization with Reinforcement Learning
- Text analytics.
- Text generation (e.g., Narrative Science).

What happens to financial sector employment?

The Atomic Level of Work



What tasks get automated first?

David Beyer

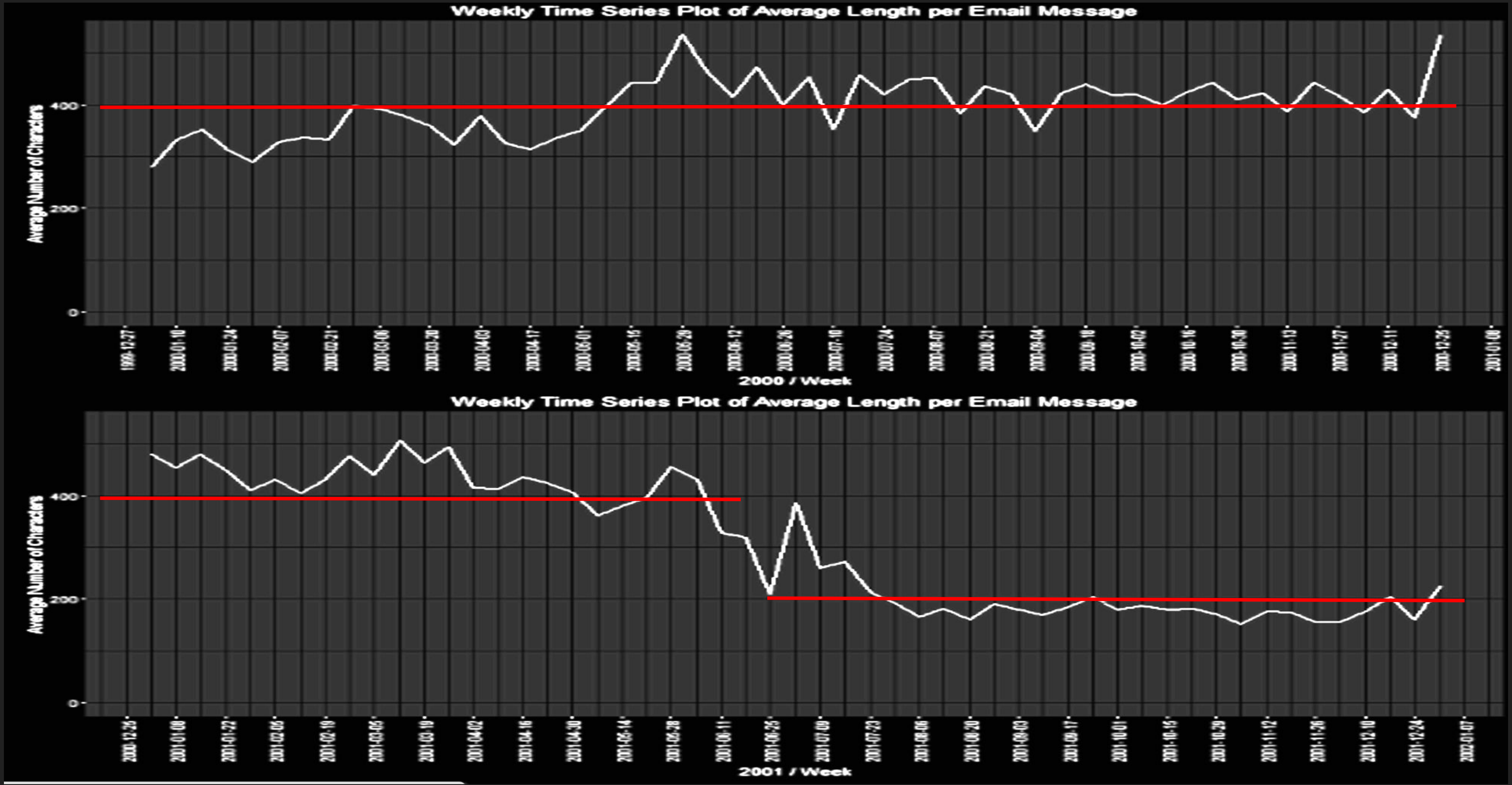
AI researchers salaries go through the roof:
https://www.nytimes.com/2017/10/22/technology/artificial-intelligence-experts-salaries.html?hp&action=click&pgtype=Homepage&clickSource=story-heading&module=second-column-region&region=top-news&WT.nav=top-news

Text, Sentiment, and RegTech

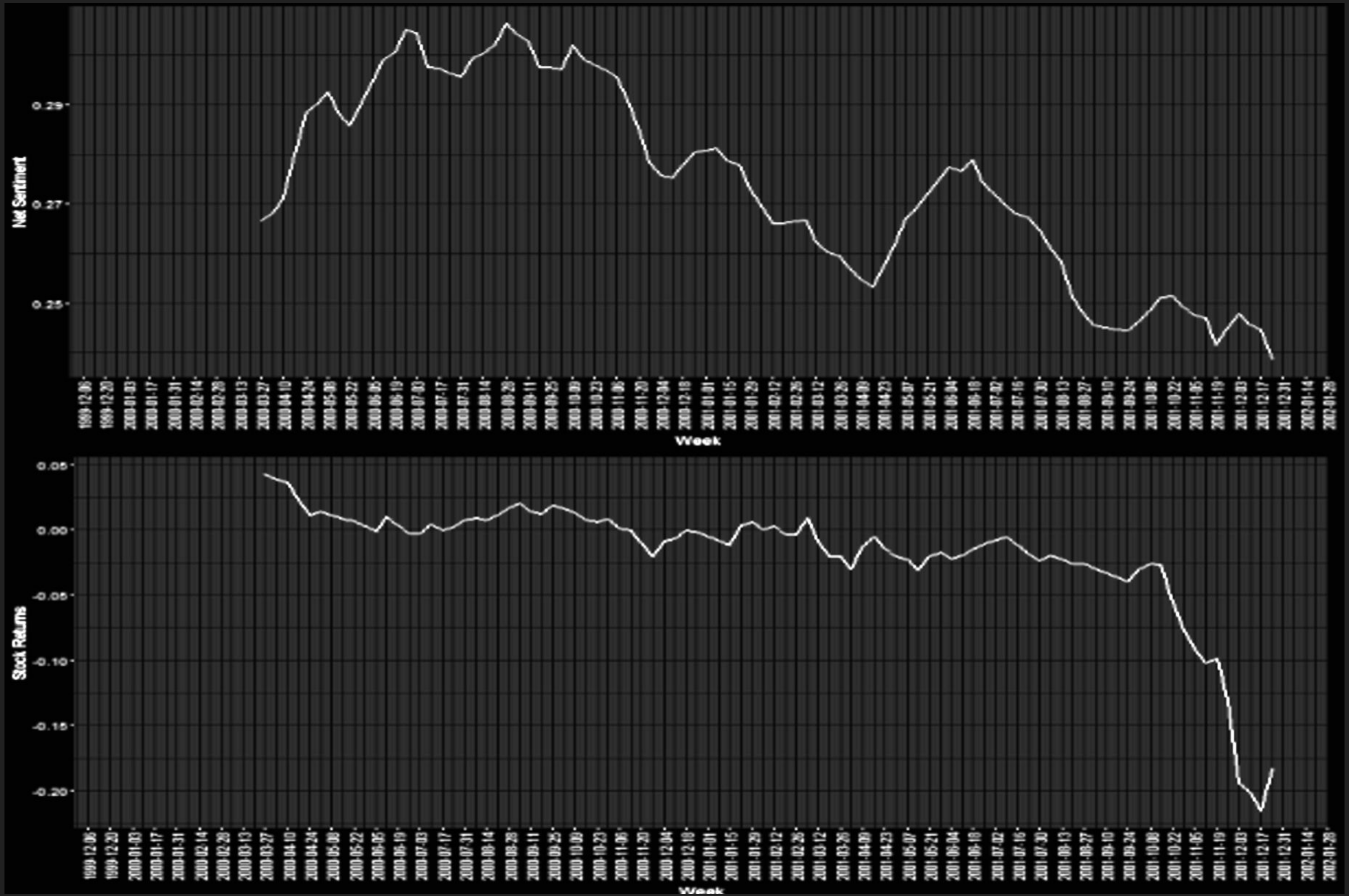
Zero-Revelation Linguistic Regulation: Detecting Risk Through Corporate Emails and News (Das, Kim, Kothari 2016)

- Financials are often delayed indicators of corporate quality.
- Internal discussion may be used as an early warning system for upcoming corporate malaise.
- Emails have the potential to predict such events.
- Software can analyze vast quantities of textual data not amenable to human processing.
- Corporate senior management may also use these analyses to better predict and manage impending crisis for their firms.
- The approach requires zero revelation of emails.

Enron: Email Length



Enron: Sentiment and Returns



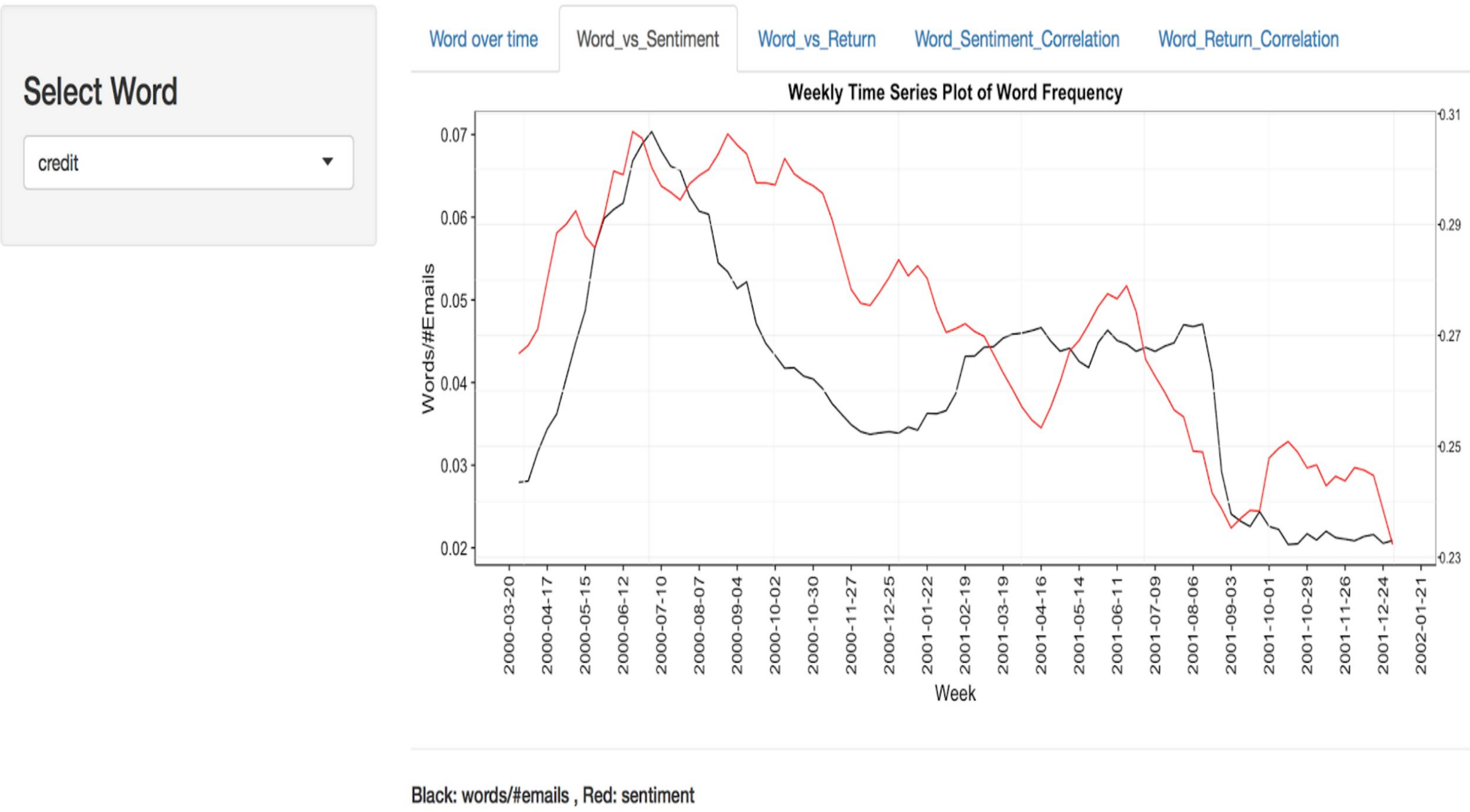
Enron: Returns and Characteristics

Variable	Coefficient Estimate (*t*-statistic)			
	(1)	(2)	(3)	(4)
MA Net Sentiment $_t$	XXX*** (XXX)	0.575 (0.63)	2.330*** (3.14)	-1.397 (-1.25)
MA Email Length $_t$		0.584*** (2.97)		1.046*** (4.19)
MA Total Emails $_t$			-0.004 (-0.10)	-0.131*** (-2.83)
Intercept		-0.406* (-1.93)	-0.671*** (-3.08)	0.117 (0.43)
Adjusted *R*-squared	XXX		0.09	0.24
Number of observations	88	88	88	88

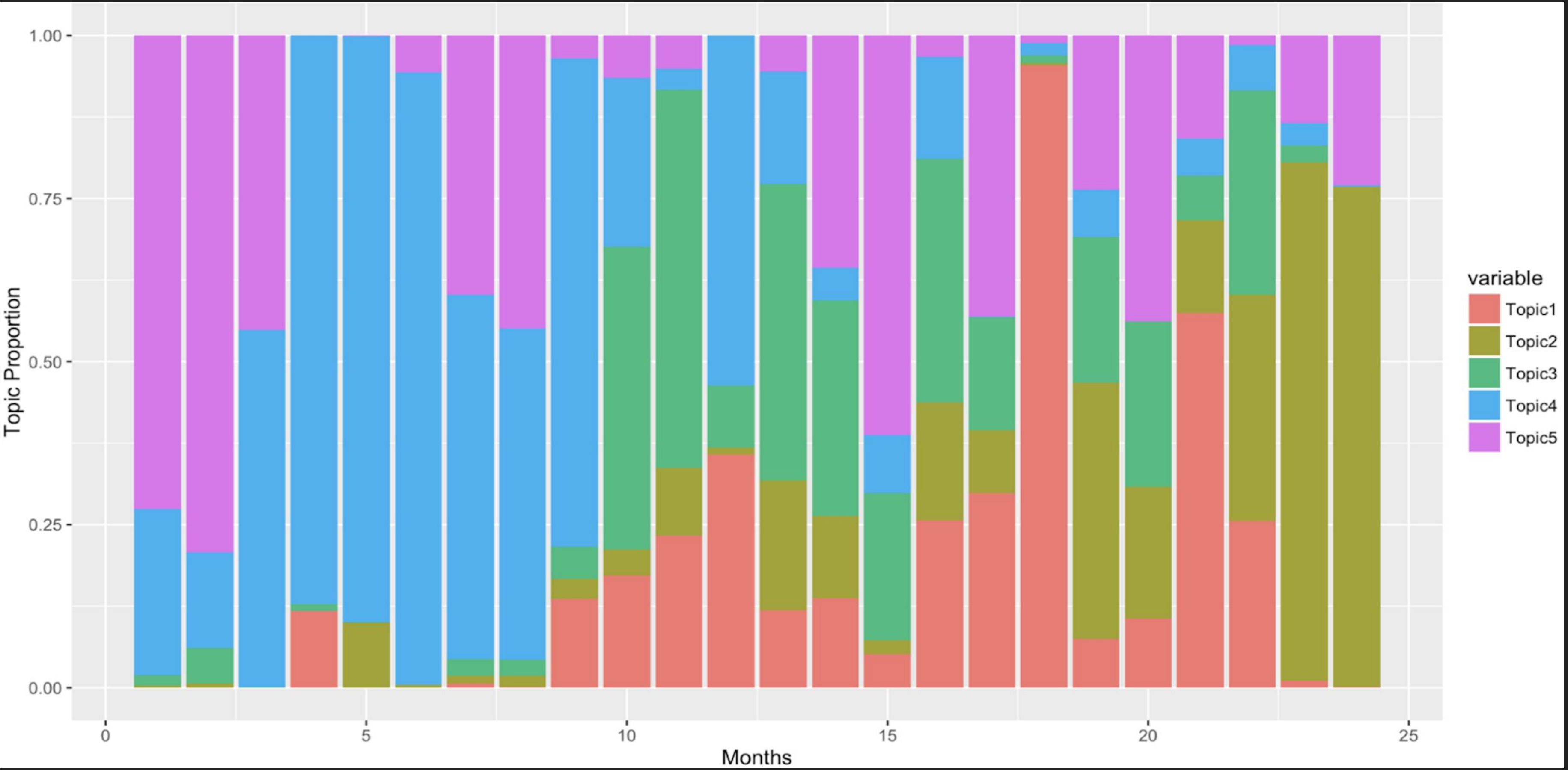
Enron: Returns and Characteristics

Variable	Coefficient Estimate (t-statistic)			
	(1)	(2)	(3)	(4)
MA Net Sentiment $_t$	XXX***	0.575	2.330***	-1.397
	(XXX)	(0.63)	(3.14)	(-1.25)
MA Email Length $_t$		0.584***		1.046***
		(2.97)		(4.19)
MA Total Emails $_t$			-0.004	-0.131***
			(-0.10)	(-2.83)
Intercept		-0.406*	-0.671***	0.117
		(-1.93)	(-3.08)	(0.43)
Adjusted R-squared	XXX		0.09	0.24
Number of observations	88	88	88	88

Enron: WordPlay

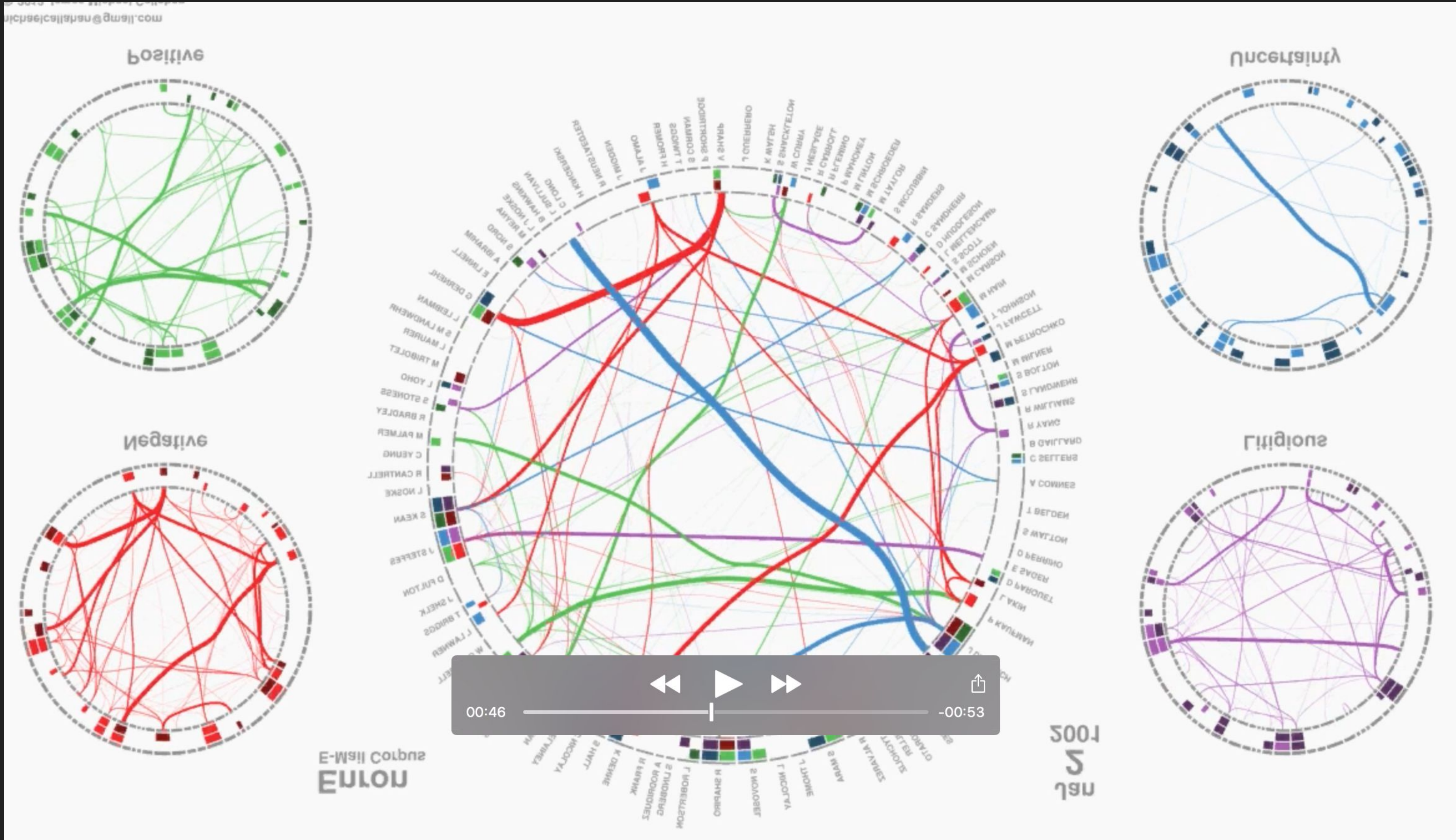


Enron: Topic Analysis



Enron Movie (by Jim Callahan)

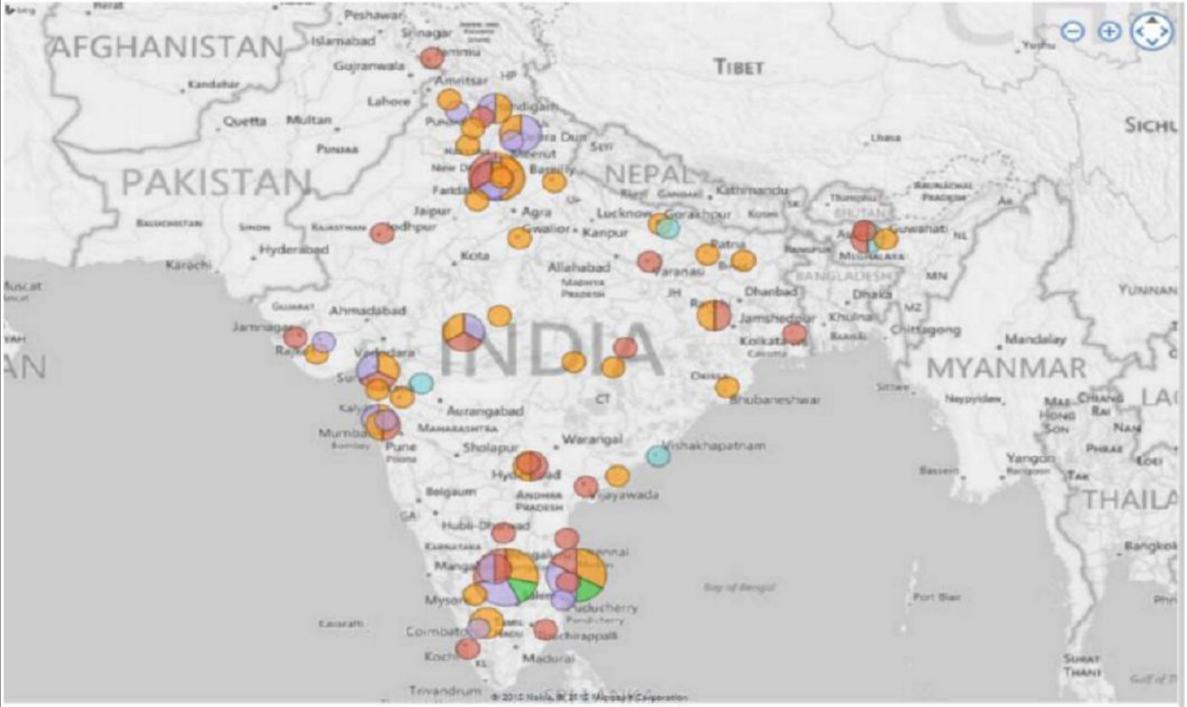
http://srdas.github.io/Presentations/JimCallahan_enron-sm.mov



India: Topic Analysis

Conversations across India and around RBI

topycs



Deputy RBI Governor · Monetary Policy · Raghuram Rajan · Rbi · RBI Governor · RBI India

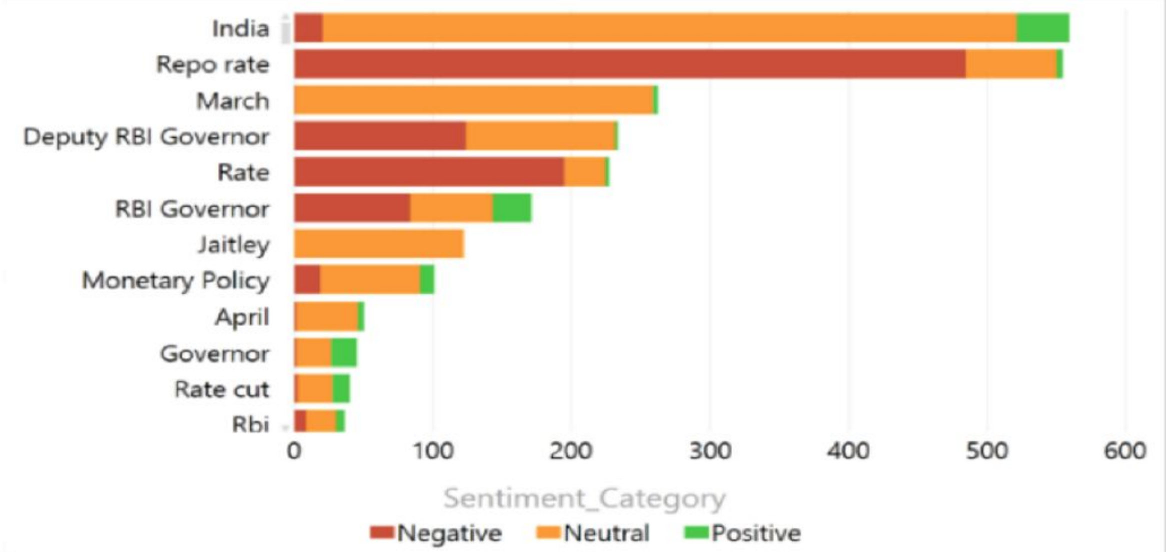
- Conversations across India on RBI, its people and the monetary policy
- Governor features in many conversations across both rural and urban areas
- Some conversations specifically around monetary policy

- Bubbles show split of conversations around Deputy RBI Governor, Monetary Policy, Raghuram Rajan, RBI and RBI Governor.
- Based on count of unique conversations
- Date Range: 1st – 14th April, 2015

India: Topic Analysis

Top Topics along with RBI

topycs



text

"@NDTVProfit: RBI unlikely to change repo rate at policy review smlion

"Digging India's RBI Out of Morass of Debt" by on

"Financial stability is like Pornography. You can't define it but when you see it you know it" - D Subbarao (RBI Governor)

"I was disappointed by the fiscal relaxation." Ex-RBI Governor on India's budget and growth:

"Rajan is perfect, he explains complex economic," PM Modi on RBI governor.

- Repo rate evokes negative sentiment as people don't expect it to be changed
- Repo rate, rate cut and monetary policy are discussed frequently with RBI

- Vertical Axis – Topics of Discussion
- Horizontal Axis – Count of Unique Conversations
- Date – 25th March - 14th of April
- Colors represent sentiment for conversation, Negative – Red, Neutral – Orange, Positive - Green

Value Drivers in FinTech

- Using Theory to develop models to apply to Big Data.
- Questions/problems are primary, data is secondary, in the success of FinTech ventures.
- Simplicity, transparency of models fosters implementability.
- Analytics per se is multidisciplinary.
- Disparate data is the norm.
- Significant investment in hardware and talent.

Pitfalls to avoid

- GIGO: Garbage in, garbage out. See [Alexander et al (2017) (http://srdas.github.io/Papers/big.2016.0074_FINAL.pdf).
- IO (Information Overload): Collecting too much data and not using it correctly. Use theoretical models.
- BiNB (Bigger is Not Better): Big data leads to bigger errors if misused. Taleb critique. TDA (topological data analysis, Ayasdi https://www.ayasdi.com/, Simility https://simility.com/.
- CCC: Confusing correlation with causality. Tighter review cycles for predictive models.
- $$$: May involve expensive infrastructure. Go all in.
- TiP (Trust is Paramount): Privacy issues. Implement trust through technology.
- CS (Customer Satisfaction): Excessive misdirected automation leading to poor client service. Robo-advising, chatbots. Use Design Thinking for consumer centric technology.

Estimating the effects of technology

- (Roy) Amara's Law: "We tend to overestimate the effect of a technology in the short run and underestimate the effect in the long run."

- Arthur C. Clarke's Three Laws:
    a. When a distinguished but elderly scientist states that something is possible, he is almost certainly right. When he states that something is impossible, he is very probably wrong.
    b. The only way of discovering the limits of the possible is to venture a little way past them into the impossible.
    c. Any sufficiently advanced technology is indistinguishable from magic.

The End !!

Thank you.