Big Data Analytics for AI Finance

Ryan McNeil
Department of Data Science
The Graduate Center, CUNY
New York, New York
ORCID: 0009-0002-7462-0360

Abstract**—** In this paper, I survey the role of big data analytics in finance, principally the growing application of AI and machine learning. I provide an overview of the broad subject and some key concepts, but focus primarily on the role of AI tools for analyzing large datasets in the finance industry. I will also cover the relatively recent integration of AI technology into finance and emerging technologies. In my exploration, I incorporate some applicable case studies as evidence of real-world successful implementations of Big Data Analytics in AI Finance highlighting the outcomes and benefits. Given the power and nascency of these tools, I also discuss challenges and ethical considerations. This paper cannot claim to present a comprehensive analysis of this field, a task which could fill multiple textbooks. However, I do provide a broad survey of the existing methods and applications.

Keywords**—** Artificial Intelligence, Machine Learning, Big Data Analytics, Computing Methodologies, Finance, Fourth Industrial Revolution

I. Introduction

In April of 1961, Yuri Gagarin became the first person to breach the Earth's atmosphere, orbiting over 300 kilometers into outer space aboard a ship without any digital computing systems. 8 years later, the Apollo 11 lander achieved soft touchdown on the Lunar surface with a 16-bit navigation computer utilizing 72KB of ROM and 4KB of RAM (about 1 millionth of 1% the power of the average modern smartphone). For the next half-century, Moore's Law would tend to hold true, with rapid exponential increases in computing power and commensurate rapid increases in the amount of digital data produced and needing to be stored and processed. As semiconductor size approaches a theoretical minimum (beyond which extreme noise and quantum mechanics may begin to interfere with performance), more creative alternatives for data processing and storage like quantum computing and scalable single-atom transistors may be necessary to sustain continued advances in computing power. However, these remain highly experimental and cost-prohibitive. Thus, more efficient and effective *methods* for data storage and processing are vital for the continued operation of our rapidly digitized economy. AI and ML technologies are considered part of a wealth of recent advancements in connectivity, automation, and analytics comprising what is being called the "fourth industrial revolution." Given the rapid digitalization of the economy, accelerated by Covid-19, the WEF and the World Trade Organization (WTO) observe significant shifts to online commerce, constant increases in data collection, transmission, and cross-border data flows, and the rapid emergence of new technologies. The World Economic Institute (WEI) estimates that 70% of new value created over the next decade will be based on digitally-enabled platform business models [1, 2, 3].

In the last half-century, financial services have come to rely on ever-evolving cutting-edge technologies in order to maximize profits. Remaining on the forefront of developing technologies, the industry has begun to adopt a variety of tools from the growing field of data science, particularly in machine learning (ML) and artificial intelligence. These include tools for asset management, algorithmic trading, and risk management, to name a few. The OECD expects that the deployment of AI in finance will lead to significant increases in productivity, efficiency, and service quality, accompanied by significant decreases in costs. However, these gains may be joined by escalated financial and societal risks. These potential risks and benefits have garnered abundant research and pending legislation across a host of national and international governing bodies, including the Organization for Economic Cooperation and Development (OECD) [1] and the World Economic Forum (WEF) [2].

Given the scope of this revolution, a literature review of relevant research on AI and ML for big data in finance is necessary to synthesize a cohesive summary of existing knowledge in the field. However, give the recency of this revolution, little such work currently exists. Thus, this paper aims to furnish this sparse field.

II. Key Concepts

In order to discuss the role of big data analytics (and AI and ML) in finance, we must first understand certain key concepts. As part of the aforementioned digital revolution, the quick and efficient storage and processing of massive datasets has become a necessity across the economy. Traditional data storage and processing methods are woefully inefficient for massive data. Thus, specialized methods and software have become a necessity. A common method for storing datasets is via Resilient Distributed Datasets (RDDs). These are single bodies of data which are split into chunks that can be stored across multiple storage systems and processed simultaneously. As the name implies, RDDs are designed to be resilient against challenges inherent in the networking of multiple systems, particularly hardware failure.

A. *Apache Hadoop & Spark*

Truly massive datasets (petabyte to exabyte size) have historically been the domain of brute force high-performance computing (HPC). More recently, advances in parallel computing and algorithms for massive data processing have produced solutions for the processing of large datasets (terabyte to petabyte size) on smaller systems (or for processing in HPC centers more efficiently). The most commonly used such tools for management and processing of large datasets are Apache Hadoop and Apache Spark, both created by the Apache Software Foundation. Hadoop is designed to handle the largest datasets and processes data directly from external storage. Hadoop can thus handle datasets much larger than can be reasonably stored in RAM (up to several petabytes of data), both structured and unstructured data, and is quite scalable (up to thousands of servers). Its utilization of external memory also provides data protection in the case of hardware failure. The Hadoop ecosystem includes the Hadoop Distributed File System (HDFS) (its primary data storage system), and Hadoop MapReduce (the popular Java-based programming model designed for processing large data sets with a parallel algorithm on Hadoop HDFS clusters) [4]. Spark, on the other hand, stores and processes data in local memory (RAM). While it can handle petabyte-sized data according to its documentation, it is designed to work faster and more efficiently for comparatively smaller datasets. Spark is also capable of real-time processing and processing of live data streams. Spark supports SQL queries, ML, and graph processing. Both systems apply specialized methods for performing these tasks [5].

B. *Algorithms*

Please note that though there has been abundant innovation in finance, the most common fundamental functions performed by algorithms in finance are still those used most commonly elsewhere: those for classification (i.e., predicting whether or not a borrower will default on a loan), prediction (i.e., predicting future asset prices; regression analyses), and search (i.e., finding data about a customer out of a distributed dataset of millions of customers). Beyond more traditional ML (collections of algorithms designed to operate on structured data), the industry has also seen utilization of more advanced neural network and AI models (collections of algorithms designed to operate on unstructured data), which are discussed more in-depth later [6].

C. Machine Learning and Artificial Intelligence.

Recent advances in AI research have garnered a wealth of attention, and with it a wave of new research into the potential applications of powerful new and emerging technologies. The terms “machine learning” and “artificial intelligence” are sometimes used interchangeably, but they are distinct. Professionals categorize ML as a *subfield* of AI. While the goal of AI is to create digital models which reproduce human intelligence, ML is the application of such models towards constructing programs which can learn on their own to solve specific tasks [36, 37]. What currently exists of AI is sometimes referred to as Artificial Narrow Intelligence (ANI) because extant models are only applicable to certain narrow sets of tasks. This would be in comparison to yet uninvented Artificial General Intelligence (AGI) or Artificial Superintelligence (ASI) which would achieve or exceed the intelligence and versatility of the human mind by some metric(s) [17, 36, 37].

D. Data Sources

The financial data sources upon which these analyses are performed are as varied as their applications (which will also be discussed later). Three sources stand out in terms of their ubiquity and efficacy: Bloomberg, Capital IQ and Thompson Reuters. All three provide aggregated up-to-date data from hundreds of exchanges and are relied upon by analysts across the industry. Bloomberg, for instance, provides live data streams which lend themselves algorithmic sampling and analysis (i.e., DGIM and reservoir sampling) [6, 7].

III. Application of Big Data Analytics in Finance

This paper will cover the four prominent areas of application for big data analytics in the finance industry: risk assessment and management, fraud detection and prevention, trading and investment, and behavior analysis. This will include a discussion of relevant ML and AI methods [14, 45].

A. Risk Assessment and Management

In the finance industry, any source of uncertainty can be considered an operational risk. The landmark study Beroggi and Wallace (1994) defines ORM as “Identification of an unexpected event, an assessment of its consequences and the decision to change the planned course of action” and proposes an algorithm for optimal decision making for effective and implementable ORM [9]. Their work has since come to be cited by a number of follow-up studies devising further advancements in algorithmic ORM. Today, operational risk is considered to come from a variety of perspectives, including the value of information assets, the cost vs. value of retaining data (big data storage), the cultural and political risks of collecting data, given the ongoing debate on this topic. Choi et. al. [8] provides the first literature review of big data analytics for business operations and risk management. They identify a whopping 12 distinct proposed ORM models, each with its own strengths and weaknesses. No one model has emerged as an industry standard [8, 9].

Any framework for ORM analysis requires a large corpus of data on organizational operation. This data is collected via data mining. In short, data mining describes any process which sorts through large datasets in order to identify new patterns. A number of studies have explored how data mining can be used for risk assessment and management. For example, Koyuncugil and Ozgulbas [10] studied its application to financial early warning and risk detection, and Hailemariam et al. [11] studied data mining techniques and algorithms for estimating customer loyalty and loan default risk. Yu et al. [12] analyzed the performance of the four most common data mining techniques. They found that regression methods and support vector machine methods yield better classification accuracies, while support vector machine methods yield highest robustness. Meanwhile, decision trees are very sensitive to input data, and thus provide unstable classification results. Deshmukh and Telluru [13] investigated a data mining technique specifically for risk management utilizing big data. They concluded that data mining techniques are an effective tool for ORM, even when compared to alternate methods, like simple statistical inference and neural network modeling [8].

B. Fraud Detection and Prevention

Financial fraud encompasses any illegal or fraudulent actions taken for financial gain. Sources have observed significant increases in rates of financial fraud in recent years, with rates projected to continue to grow in the near future [14, 15, 16]. In just the last few years, financial fraud has been exacerbated by increased use of online banking, cashless commerce, and the proliferation of the “Internet of Things.” Large amounts of money are estimated to be lost to financial fraud globally every day [14, 15, 16]. Most existing methods of fraud detection are still manual, and thus completely infeasible for big data. Though recent breakthroughs in AI and ML methods for fraud detection are promising, much more progress is necessary to combat the sheer quantity of ongoing fraud. Moreover, the nascency of these technologies means that several gaps in both research and development into ML methods for fraud detection in large datasets, and most existing research only focuses on individual areas and types of financial fraud, leaving a deficit of work synthesizing existing research on *all* financial fraud [14]. Ali et al. [14] provides the most comprehensive recent literature review on ML for financial fraud detection and mitigation.

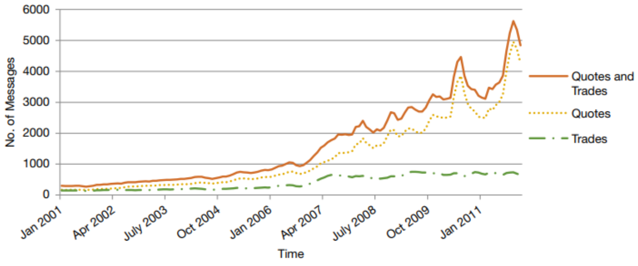
Broadly speaking, financial fraud can be grouped into 3 categories: Bank fraud (encompassing 50% of most research on fraud detection), financial statement fraud, and insurance fraud (with about 7% of fraud detection research not falling into any of these three categories). Ali et. al. [14] studied the application of a wide variety of ML methods for identifying each of these types of fraud. The most commonly used method of fraud detection is the Naïve Bayes Algorithm based on Bayes’ Theorem. The Naïve Bayes Algorithm is common in big data analysis, and is preferred for its ability to provide higher accuracy categorical variable classification results than other models with less training data, especially on datasets with independent features. While bank fraud is known to be the subject of significant amounts of fraud, these findings imply that other kinds of fraud (like stock and commodities fraud) may be woefully under-researched and thus under-detected [14].

AI research has led to recent breakthroughs in financial fraud detection. Real-world application of AI tools for financial fraud detection has rapidly expanded since around 2018, the same year that a prominent case study was published demonstrating the viability of AI for fraud detection with a novel purpose-built AI model. Soviany [17] produces an AI model which significantly outperformed alternative fraud detection methods in terms of both fraud detection volume and false positive rate. This model achieves these results by the application of advanced data analytics and model training improvements. First, PCA is performed on raw customer data in order to analyze the overlaying data structure, reduce dimensionality, and to help identify patterns in data which initially contained quite a bit of noise. Then, different subsets of the original input variables are tested with PCA in order to identify the variables which produce data with the best class separability (between fraudulent vs non-fraudulent transactions). Once optimally separable data has been produced, training data is non-randomly selected from it in order to produce a model which achieves a predetermined fixed-target overall positive rate. Finally, a performance evaluation is performed using ROC curves on subsamples of the training data and confusion matrices for the test data. The final model produced by this process detects fraud with an accuracy of up to 96.59%, depending on how high a user sets the maximum alert rate (proportion of transactions flagged by the AI), up to a reasonable maximum of 10% (depending on overall transaction volume vs number of humans available to review flagged transactions).

C. Trading and Investment

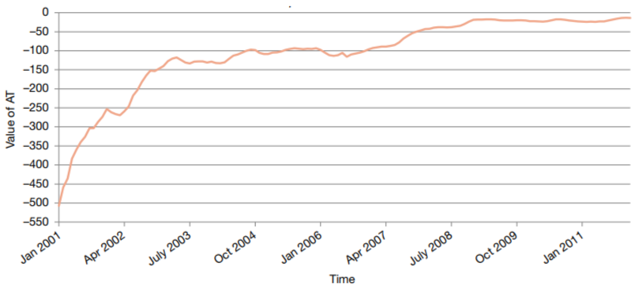
a) Algorithmic Trading

Algorithmic Trading (AT) has been in general use by large firms since about the turn of the millennium. However, algorithmic Trading and Investment strategies are the subject of much ongoing research, with a wealth of not just algorithmic, but *AI* trading and investment tools coming to market recently. These methods have been aggressively adopted and refined to keep up with the pressures of such a competitive industry. Boehmer et. al. [18] studied the effects of AT on market Quality between 2001 and 2011 in 42 equity markets globally. Utilizing market data from Reuters, the New York Stock Exchange (NYSE) and the Trades and Quotes (TAQ) database, they calculate several variables as metrics of market quality, measured in 5-minute intervals across the entire 10-year period: liquidity (the availability of liquid assets), informational efficiency (the extent to which prices incorporate all available information about future values), volatility (the pace at which



[a] Source: [18]

Fig. 1 Time-series AT Message volume [18]



[a] Source: [18]

Fig. 2. AT Message volume and value per message from [18]

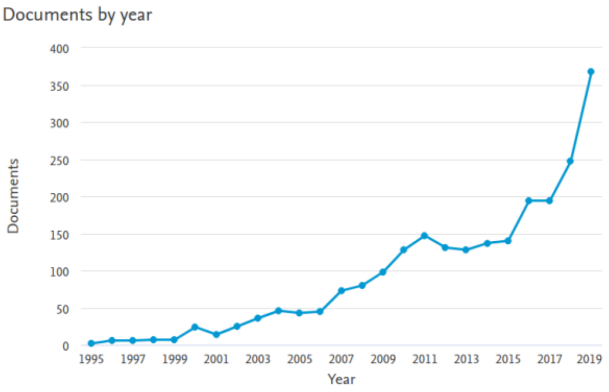
prices change), execution shortfall (the differences between the price at time of decision to trade and the actual price at the instant a trade is executed), and AT activity (a derived variable indicating the relative intensity of AT in the market at a given place and time). Their analysis found that over the 10-year period the adoption of AT methods and the volume of AT activity per day ballooned across nearly all markets. Fig. 1 shows their findings, averaged across all markets. Fig. 1 plots the number of messages (trades, price change reports, and price quotes) daily between 2001 and 2011, illustrating the exponential increase in AT use over this period, particularly the beginning of high-frequency AT around 2007 [1, 18]. Fig. 2 likewise illustrates the time-series of their AT variable, which can be interpreted as the negative of dollar trading volume per message. In Fig. 2, the number of AT messages per day increased, the dollar amount of trading per message decreased, reflecting both more frequent use of AT methods (not just for trading, but for quoting and exploration), and their increased use for smaller everyday trades.

Their analysis also revealed that, on average, AT led to significant increases in liquidity and informational efficiency, and decreases in execution shortfall (perhaps explaining the continued increases in adoption of AT methods). These improvements were observed to be accompanied by significant increases in short-term volatility where AT was more widely adopted. This is attributed to the fact that AT enables much more rapid trade, contributing to more rapid short-term price fluctuations. However, their research does not suggest that this necessarily translates to more rapid long-term volatility and they argue that appropriate regulation capping the size and number of AT transactions per day may mitigate this downside [18]. Concerns about this potential flaw and recommendations to legislate to control it are echoed in the recommendations by governing bodies [1, 2].

b) AI Trading & Price Prediction

Since the publication of Boehmer et. al. [18], two prominent literature reviews have synthesized more recent research on not just algorithmic, but *AI* tools for stock trading, including and stock price prediction (a topic of especial interest, given its utility for profitable trading) [19, 20]. Stock price prediction has been attempted using the breadth of applicable prediction models:

1) Regression models: Linear regression attempts to find a line of best fit between points. Logistic regression classifies data in a binary by calculating a line of best fit which maximizes the distance between points.
2) K-Nearest Neighbors (KNN): Given a data point, calculates the K number of points nearest to that point in the data space (by either Euclidean or Manhattan distance) in order to determine prediction or classification values about that point.
3) Support Vector Machines (SVMs): Constructs hyperplanes through the data space in order to partition data points into categories. Particularly robust, resilient to outliers, and effective on high-dimensional data.
4) Decision Trees and Random Forests: Decision trees operate somewhat like a flowchart, taking input data points and passing each through a series of branching nodes (with direction being determined by some variable value) until data is sufficiently sorted to a single end node (determining output). Most decision tree training algorithms test various splits, selecting the splits which maximize a given training metric on the training data. Random forests extend this logic by combining multiple decision trees. The output of each decision tree in a random forest is considered a “vote”. The forest gives the output most agreed upon across all decision trees in the forest.
5) Neural Networks (NNs): Specifically Artificial Neural Networks (ANNs), also called Simulated Neural Networks (SNNs). Attempts to emulate in simplified form a model of the way a human brain processes information. Consists of multiple layers of interconnected nodes connected by synapses. At the input layer, each node corresponds to an input variable. Intermediate layers are a black box with no simple abstract definitions or interpretation. Data are directed from node to node between layers based on variable values until reaching an output layer, where a synthesized decision or interpretation is output. Neural Networks are trained by iteratively adjusting layer connections to minimize a loss function. The loss function can be any one of a family of functions which



a. Source: [19]

Fig. 3. Published documents on AI for financial investments by year.

measure the difference between algorithm output and expected output.

6) Hybrid Models: Hybrid models (or Advanced Models) feature multiples or combinations of the other 4 model types. These include what are being called Generative Adversarial Networks (GAN) which are composed of two neural networks which compete to outperform one another on performance metrics, recursively adapting their models appropriately after each win or loss [19, 20].

Given both their complexity and predictive potential, AI and neural networks (NNs) in particular are currently the forefront of trading technologies research [45]. A review of the recent literature reveals multiple original NN models for stock price prediction and trading which outperform traditional ML methods in terms of overall returns and overall risk in simulations [19, 21]. NN and AI models are also useful for their applicability to very large datasets. Given that trading data is typically recorded in small (usually ~5 minute to ~1 day) time intervals relative to the total length of time observed (often many years), stock market variables can comprise quite long datasets (many in the terabyte range) [19, 20, 21]. Fig. 3 summarizes Ferreira et. al.'s [19] findings on the number of journal articles and conference papers published on the application of artificial intelligence to financial investments.

c) Portfolio Optimization

Given the aforementioned increases in market volatility that may be associated with increased adoption of AT, algorithmic portfolio management can help keep up with the growing pace of the marketplace, managing associated risk. Portfolio management is the process of continuously reallocating funds into different assets in order to increase expected returns and minimize risk [22]. Traditionally, a portfolio manager needs to balance maximizing value generated by investments, balancing risk and expected returns, anticipating and seizing on future events and opportunities, and selecting the best combination of assets and degree of portfolio diversification. This is a difficult and expansive set of responsibilities for a human portfolio manager. Thus, ML techniques are becoming very common in this domain. While SVMs and Bayesian Analysis-based methods have achieved some successes, NNs have yielded better results than either [22, 23, 24].

While the scope of new such technologies is too large to cover comprehensively in this paper, I will review one particularly successful AI portfolio management model recently devised by Haddadian et. al. [25] which builds on prior existing models and will illustrate for the reader an example algorithmic model for both trading and portfolio optimization. Their model begins by carefully selecting and optimizing variables via a genetic algorithm. In each of a sequence of time-series points in which a buy/sell/hold signal can be sent (T_1, T_2,…,T_n), one could look for such a signal (S_j), at any point such that (T_{i-1}, < S_j < T_{i+1}). A fitness function for any point T_i will have several conditions:

If T_i is an anticipated trading point for *buying* and S_j is a proposed buy signal:

$$Fitness\ (T_i) = Close(S_j) - Close(T_i) \tag{1}$$

Where Close(T_i) is the closing price at T_i, and Close(S_j) is the closing price at S_j. In this case, the closer the closing prices are at points S_j and T_i, the smaller the fitness value. If T_i is an anticipated trading point for buying and S_j is a proposed sell signal:

$$Fitness(T_i) = 2(max(close(T_{i-1} : T_{i+1})) - Close(T_i)) \tag{2}$$

Where:

$$\frac{|Close(Sj) - Close(Ti)|}{Close(Ti)} < 0.05 \tag{3}$$

In this case, the maximum closing price max(close(T_{i-1} : T_{i+1})) occurs between T_{i-1} and T_{i+1}. Such a function will penalize sell signal at a minimum price point, since this would be the wrong time to trade. If T_i is an anticipated trading point for buying and there is no signal (or "hold") between T_{i-2} and T_{i+2}, the fitness equation will be:

$$Fitness(T_i) = 2(max(close(T_{i-1} : T_{i+1} - 1)) - Close(T_i)) \tag{4}$$

And a penalty will be banked for a for a loss of a trading opportunity. If T_i is an expected trading point for *selling* and S_j is a proposed sell signal, the fitness function will be:

$$Fitness(T_i) = Close(T_i) - Close(S_j)) \tag{5}$$

If T_i is an expected trading point for *selling* and price at S_j and T_i are close, but S_j is a proposed buy signal, the fitness function will be:

$$Fitness(T_i) = 2(Close(T_i)) - min(Close(T_i{:}T_i)) \tag{6}$$

Where *min(Close(T$_i$:T$_i$)* is the minimum closing price between T$_{i-1+1}$ and T$_{i+1-1}$ (which can be considered as simply T$_i$, but is written thusly for formality). If T$_i$ is an expected trading point for *selling* and there is no signal (or "hold"), the fitness function will be:

$$\text{Fitness}(T_i) = \text{Close}(T_i) - \min(\text{Close}(T_i{:}T_i)) \qquad (7)$$

The final fitness function for all points in the sequence S = {S$_1$, S$_2$, …, S$_n$} will be:

$$Fitness\ (S) = \sum_{i=0}^{n} Fitness(\mathrm{Ti}) \qquad (8)$$

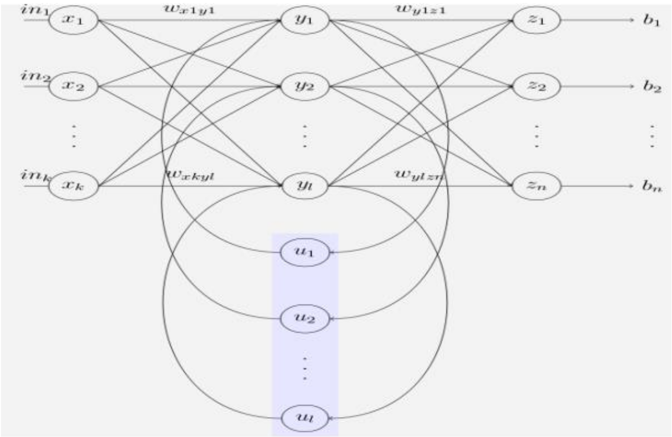
Next, a fuzzy inference system is used to identify market trends. Fuzzy inference systems are not widely used in big-data finance, so this remains an investigational approach. Given the lack of consensus on how to calculate market status, Haddadian et al. [25] argues that fuzzy logic system may be effective. Fuzzy logic can prove useful at improving prediction in situations with high uncertainty. At this stage, data about moving average prices, slope of moving average, and index of directional movement (trending vs not trending) are normalized and fuzzified starting with the following standard equation:

$$v^{*} = \frac{|2v - \max(v) + \min(v)|}{\max(v) - \min(v)} \qquad (9)$$

Where max(v) and min(v) are the maximum and minimum values in the training data, respectively. Each v^* is then processed through a trapezoidal membership function in order to fuzzify it. The data is then defuzzified to produce a crisp number using the center of gravity (COG) method first properly adapted by Subbotin et. al. [26]:

$$COG = \frac{\int_a^b \mu_A(x).x\ dx}{\int_a^b \mu_A(x)\ dx} \qquad (10)$$

Where μ_A(the y-axis-value of our fuzzified data) is the membership function value defined by a series of conditional functions unique to each of the input variables (with conditions defined manually by researchers), and *x* (the x-axis-value) is the given parameter being de-fuzzified between points a and b. This method calculates a "center of gravity" of a curve of *x* over μ. The condition of the market (trending vs not trending) is determined by whether or not the COG value exceeds a chosen threshold. Depending on the determination made by COG, variables about market data are used to calculate the set of indicators best suited for that market situation. For a trending market: double moving average, directional average indicator, parabolic SAR, moving average convergence/divergence (MACD), and OBV indicator. For a non-trending market: momentum index, relative strength indicator (RSI), stochastic index, MACD indicator, and money flow index (MFI). These indices are then used to generate a buy/sell/hold signal for each asset [25].



a. Source: [25]

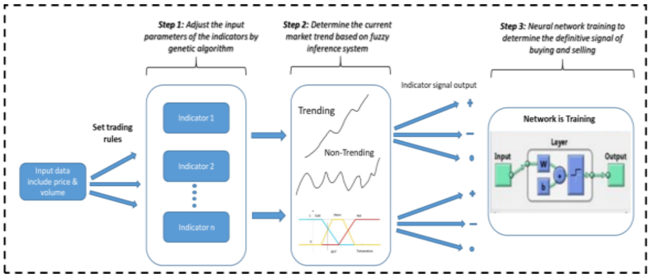
Fig. 4. Elman Recurrent Neural Network

Third, these signals are aggregated through a neural network, which outputs a buy, sell, or hold signal for each of the given stocks. This is the step most common and relevant to the application of AI and big data analytics in finance. They opt to use a 3-layer Elman Recurrent Neural Network (RNN). An RNN is a neural network which allows for bi-directional dataflow; allows output at one layer to affect later input at the same layer. An Elman RNN is a variant of RNN with a 3-layer architecture and a hidden layer for storing contextual information [27]. Fig. 4 illustrates the structure of an Elman RNN. These kinds of NNs are resilient with unsegmented and less structured data, and thus particularly effective for applications like Natural Language Processing (NLP) and forecasting (Elman RNNs are commonly used for predicting change over time in things like stock prices or weather patterns), and typically outperforms standard Perceptron models at these tasks [25, 27].

Lastly, the candidate list for buying & selling is added to the portfolio and the portfolio is rebalanced by a portfolio optimization model. The portfolio must be rebalanced by taking into account returns and candidate stocks for entry since the last rebalancing. Haddadian et. al. [25] uses the common "Markowitz" model, which aims to maximize returns while mitigating risk via diversification. Here, it should be noted that the Markowitz model assumes normal distribution of data, which may be broken under conditions of a volatile market. As discussed, AT may *contribute* to market volatility. Additionally, RNNs are subject to the risk of negative feedback loops, especially contributing to gradient explosion or gradient vanishing. Either could contribute to poor trading behavior and increased market volatility [28]. Given these concerns, Haddadian et. al. [25] opt to place a number of constraints on their portfolio optimization model. These constraints function to (1) limit the amount of equity in the portfolio (2) impose maximum and minimum proportion of portfolio that a share is allowed to be compose (3) maintain a maximum trading volume per month, in order to maintain liquidity and limit market impact [25]. The final portfolio balancing model is:

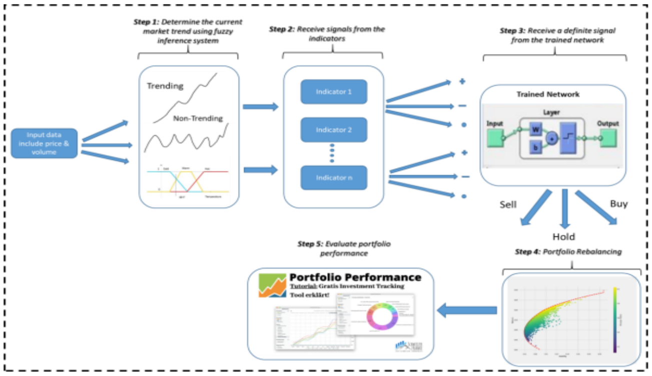
$$Max(Z) = \lambda \sum_{i=1}^{n} W_i \mu_j - (1-\lambda) \sum_{i=1}^{N} \sum_{j=1}^{n} W_i W_j \sigma_{ij} \quad (11)$$

Z_i being the proportion of the portfolio represented by a given asset (the sum of all Z_i being 1), λ is a weighting coefficient between 0 and 1 selected depending on investor risk preference, W_i is the fraction of the portfolio that the given share represents (the sum of all W being 1), μ being the membership function value (from fuzzified data), and σ being the maximum proportion of the portfolio allowed per share of the given asset. This model continuously reinvests money obtained from sales and optimizes diversification to mitigate risk. The complete training structure of this model is illustrated in Fig. 5, and the final implementation is illustrated in Fig. 6. The final trained model was tested on a real portfolio over the course of 18 months. Comparison on measures of investment performance (calculated on weekly time-series data over the course of the full 18-month test period) found that this model outperformed competing algorithmic trading and portfolio management models in both maximizing returns and minimizing risk. For example, the complete algorithmic portfolio accrued higher overall returns than competing models, while still achieving a maximum drawdown (maximum loss from a peak to a trough in value; a measure of risk) of -8.90%, compared to the next-best-performing funds, which achieved maximum drawdowns of between -13.10% and -19.60% [25].


Step 1: Adjust the input parameters of the indicators by genetic algorithm
Step 2: Determine the current market trend based on fuzzy inference system
Step 3: Neural network training to determine the definitive signal of buying and selling
Input data include price & volume
Set trading rules
Indicator 1
Indicator 2
Indicator n
Trending
Non-Trending
Indicator signal output
+
–
•
Network is Training
Layer
Input
Output
W
b

[a] Source: [25]

Fig. 5 Construction and training of portfolio management model.


Step 1: Determine the current market trend using fuzzy inference system
Step 2: Receive signals from the indicators
Step 3: Receive a definite signal from the trained network
Input data include price & volume
Trending
Non-Trending
Indicator 1
Indicator 2
Indicator n
+
–
•
Trained Network
Layer
Input
Output
W
b
Sell
Hold
Buy
Step 4: Portfolio Rebalancing
Step 5: Evaluate portfolio performance
Portfolio Performance
Tutorial: Gratis Investment Tracking Tool erklärt!

[a] Source: [25]

Fig. 6 Implementation and evaluation of portfolio management model.

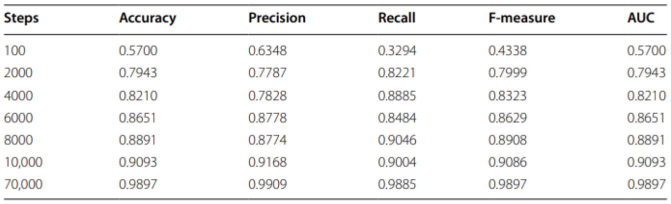
d) Sentiment analysis

One area where big data analytics is especially useful is in social media analysis. In such an environment, there is typically much more data than can reasonably be poured over manually by a human user, and trends can be even harder to piece out by traditional means. Some research [29, 30, 31] even draws a straight line between the structure of social media data and stock market data in that they are both composed of social networks. Sentiment analysis may be useful for gleaning useful financial information from sentiments on social media platforms. Moreover, the realm of financial advising has been dominated by massive investment banks. Today the industry is led by large conglomerates like Goldman Sachs and JP Morgan Chase. Advising from these firms is often only expensive and only accessible to large business entities and wealthier individuals. Further, while advising from these firms is expert, these experts still cannot be expected to be right 100% of the time and thus may disagree with one another. Where financial advice from representatives of these organizations is publicly available, it would theoretically be possible to aggregate and analyze that data using big data analytics and ML methods in order to produce a model which is more accurate and reliable than any one of those sources alone [30, 31].

In recent years, researchers have attempted to do just this, utilizing data from similarly growing financial social media networks like “StockTwits” and “SeekingAlpha” and using experimental new tools like NLP. Several studies have explored ways of doing this, so in the interest of brevity, I will review one recent case study demonstrating a particularly powerful model built from a combination of big data analytics and deep learning (based in neural networks) methods. Sohangir et. al. [29] begins with a dataset of ~9 years of data from the StockTwits financial social network website. This data includes user posts (140 character maximum of text, plus a binary variable of “bullish” or “bearish”, as well as user data) and market data (e.g. prices, movement, exchange history and recommendations). They began with a test run using only 6 months of user posts. Using a Pearson Correlation Coefficient to test for a correlation between user sentiment and stock movement (within a short time after the timestamp of the post), they revealed that stock price only fluctuated in agreement with user sentiment about 53% of the time. By narrowing down the users sampled to a subset of “top authors” (selected based on how often their sentiments correlated with future performance), they were able to raise this correlation to only about 75% (and lost a significant amount of data in the process). They thus posit that a more powerful method would be necessary for sentiment analysis in this environment, and point to deep learning methods for a potential solution for extracting meaningful patterns from a massive amount of data.

They begin with a basic model produced by Wang et al. [32], which begins by collecting text post data from StockTwits, removing stopwords, symbols, and company names. Wang et.

TABLE I. CNN Performance



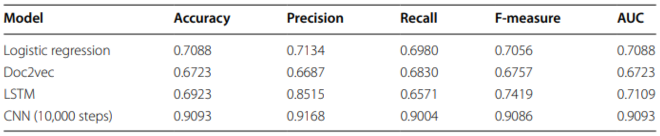
Steps	Accuracy	Precision	Recall	F-measure	AUC
100	0.5700	0.6348	0.3294	0.4338	0.5700
2000	0.7943	0.7787	0.8221	0.7999	0.7943
4000	0.8210	0.7828	0.8885	0.8323	0.8210
6000	0.8651	0.8778	0.8484	0.8629	0.8651
8000	0.8891	0.8774	0.9046	0.8908	0.8891
10,000	0.9093	0.9168	0.9004	0.9086	0.9093
70,000	0.9897	0.9909	0.9885	0.9897	0.9897

a. Source: [29]

al. then structures the data thusly: each post as a row, unigrams (n-grams of n=1) as features. However, by this method with low-order n-grams Wang et al. loses meaning stored in the *order* of words. Conversely, at higher n-gram orders, they would face a data-sparsity problem. Their final SVM model achieves an accuracy of 76.2%; that is 76.2% of the time, the model could correctly predict future asset performance based on StockTwits post data. Sohangir et al. demonstrate that using Chi-squared or ANOVA for feature selection yield no improvement (accuracies topping out at ~71.88%) in performance over Wang et al.'s SVM model on unigram data and thus propose an alternative of deep learning nonlinear feature extraction. They test 3 ML methods: Doc2vec (a popular neural-network-based model commonly used for NLP), Long short-term memory (LSTM; a kind of RNN commonly used for NLP), and a convolutional neural network (CNN). Convolutional neural networks are a kind of feed-forward neural network (FNN) which are commonly used in image processing due to their strength for finding the internal structures of big datasets. CNNs are thusly often considered a form of deep learning. In a CNN, each neuron in the first hidden layer only connects to *some* of the neurons in subsequent layers, rather than all of them. CNNs convert input data to numeric values, then pass this data through at least one layer which performs a "convolution", a linear operation which multiplies each input datapoint by an array of weights (sometimes called a "filter"). Using such a filter of weights avoids the problem (encountered by other FNNs in the case of big data) of massive numbers of individual weights. Despite their power, CNNs can be trained and applied on a typical desktop, and Sohangir et. al. use a CNN method freely available in the Python package Tensorflow. They train and test CNN models with varying numbers of convolutions (which they call "Steps", but can also be understood as "depth"). Please note that the largest model (70,000) is unreasonably large and potentially prone to overfitting, so the authors opt to take the second-largest model as their final optimized CNN.

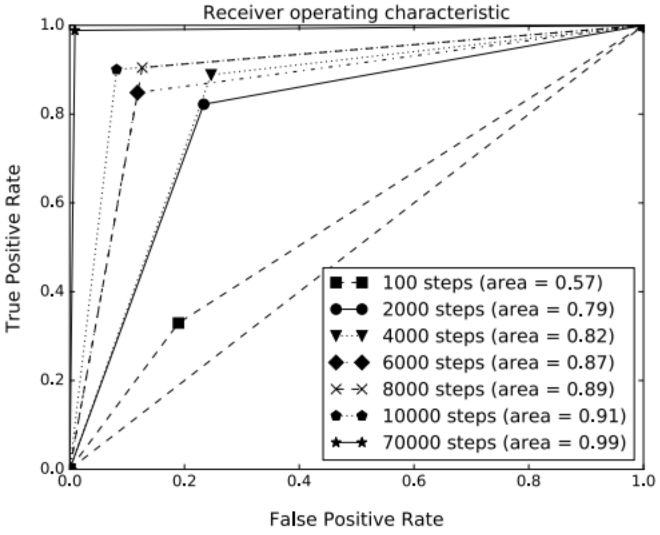
As we can see from Table I, CNN outperforms a basic unigram data logistic regression model at a minimum depth of about 2000, and at its maximum depth can achieve an accuracy of ~99% (with comparable precision and recall). Fig. 7 shows the ROC curve for each of these CNN models, demonstrating their increasing power by depth. Further, Table II compares the performance of the optimal CNN model with each of the other

TABLE II. Comparing Deep Learning Models Performance



Model	Accuracy	Precision	Recall	F-measure	AUC
Logistic regression	0.7088	0.7134	0.6980	0.7056	0.7088
Doc2vec	0.6723	0.6687	0.6830	0.6757	0.6723
LSTM	0.6923	0.8515	0.6571	0.7419	0.7109
CNN (10,000 steps)	0.9093	0.9168	0.9004	0.9086	0.9093

a. Source: [29]



a. Source: [29]

Fig. 7 ROC curves of CNN models by steps.

methods attempted. Note that Doc2vec and LSTM (both neural network models) actually *underperform* compared to the basic unigram data logistic regression model. This finding serves as a reminder that increasing model complexity does not necessarily mean increasing model performance. Sohangir et. al.'s model is presented as a proof of concept, and has not been adapted to work with data streams. I propose that a CNN model like theirs with a depth of about 10,000 could be trained and tested on a larger dataset (i.e. expanding the test/training set to the full year range of StockTwits data). A longer time period would allow for a model which accounts for more long-term changes in things like market performance and posting behavior. This may be especially necessary in order to detect more long-term patterns in both of these behaviors, which are themselves entwined in long-term social and cultural evolution.

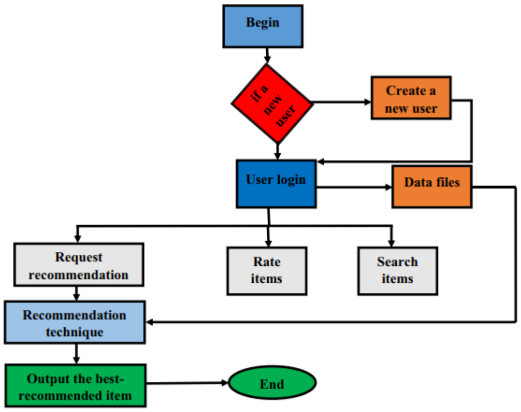
Should this CNN model still prove useable for the full dataset, this final model could be applied to a data stream from StockTwits. Should the volume of market data be too large to continuously store all new data, this model could mine a stream of data (live market and post data from StockTwits) using a sampling method, common with large data steams. Should that data prove too large to be stored reasonably in memory or on disk, a DGIM sliding window of a fixed timescale may be utilized. This method may be especially useful, given that DGIM does not assume uniformity, which is often the case with regard to market data. Alternatively, to manage the scale of data, a model could first be trained and tested on data that only pertains to one industry or one asset. This could be achieved by filtering the whole data stream via a Bloom Filter. Given that these big data streaming methods all involve taking a subset of the whole data stream, they cannot provide a global view of all the available data. So, model performance may be reduced relative to its performance on the static dataset used by Sohangir et. al. This hypothetical model for use with a data stream should first be tested for performance over a reasonable test period, then tested on an experimental portfolio of manually predetermined assets (studies often test such new models for between 6 – 24 months) [19, 25, 29, 31].

D. Behavior Analysis and Personalization

The final area of application for ML and big data methods in finance is behavior analysis and personalization. Its potential applications in finance for market prediction are somewhat more nebulous than other methods. Unlike sentiment analysis, behavior analysis is more difficult to apply directly to assets trading, and instead tends to be applied for understanding and influencing how individuals interact with financial products and services. This understanding is then used to tailor and personalize offerings to meet the unique needs and preferences of individual customers. In the context of the financial industry, this means using ML recommendation systems to suggest personalized financial products (like loans, portfolios, and assets) based on their transaction history, financial goals, and/or risk tolerance. Recommendation systems are ML algorithms designed to recommend items to users based on their previous behavior or the behavior of other users similar to them [32, 33, 38].

There are three primary methods to recommendation systems: content-based, collaborative, and latent-factor-based. Content-based systems recommend items to a user based on previous item ratings by that user. Collaborative filtering recommends items based on the behavior of other users who rated the similar items similarly. Latent-factor-based systems (e.g. SVD) utilize matrix algebra to extract useful information from (sparse) matrices of users and items. It is also possible to construct hybrid models, which generate recommendations based on a combination of one of these three methods.

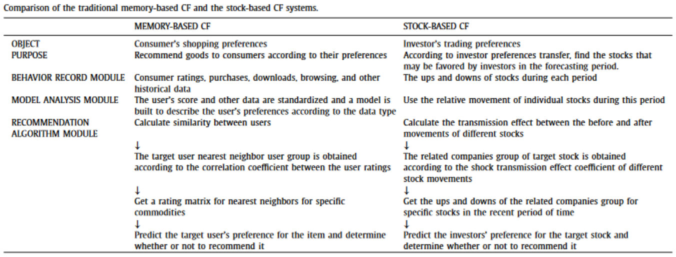
Marwa et. al. [33] provides the most recent survey of recommendation systems for major industries. First and foremost, they acknowledge that it is challenging to extract useful data from the abundance of available data on online commerce, and that this challenge may be best addressed by ML methods designed for big data. The structure of recommendation processes are illustrated in Fig. 8. The primary innovations are entirely in the methods employed for processing user and/or product data to provide a recommendation(s).



a. Source: [33]

Fig. 8 Recommendation process

TABLE III. Comparison of traditional CF to stock-based CF



Comparison of the traditional memory-based CF and the stock-based CF systems.

	MEMORY-BASED CF	STOCK-BASED CF
OBJECT	Consumer's shopping preferences	Investor's trading preferences
PURPOSE	Recommend goods to consumers according to their preferences	According to investor preferences transfer, find the stocks that may be favored by investors in the forecasting period.
BEHAVIOR RECORD MODULE	Consumer ratings, purchases, downloads, browsing, and other historical data	The ups and downs of stocks during each period
MODEL ANALYSIS MODULE	The user's score and other data are standardized and a model is built to describe the user's preferences according to the data type	Use the relative movement of individual stocks during this period
RECOMMENDATION ALGORITHM MODULE	Calculate similarity between users ↓ The target user nearest neighbor user group is obtained according to the correlation coefficient between the user ratings ↓ Get a rating matrix for nearest neighbors for specific commodities ↓ Predict the target user's preference for the item and determine whether or not to recommend it	Calculate the transmission effect between the before and after movements of different stocks ↓ The related companies group of target stock is obtained according to the shock transmission effect coefficient of different stock movements ↓ Get the ups and downs of the related companies group for specific stocks in the recent period of time ↓ Predict the investors' preference for the target stock and determine whether or not to recommend it

a. Source: [35]

There are four subfields of finance to which recommendation engines can be applied: banking, stock trading, insurance, and real-estate. We have already seen versions of these applications in previous examples. For example, Vismayaa et. al. [34] produces a kind of recommendation system which provides trading recommendations based on market performance. Their model is comparable to automated trading algorithms, like the market analysis portion of Haddadian et. al.'s [25] model, with the exception that Vismayaa et. al.'s model provides *recommendations* for trades, rather than performing them automatically. Another model, constructed by Zheng et. al. [35] uses collaborative filtering to produce a stock prediction algorithm (average annualized return of 11.42%). Their algorithm proceeds from the assumption that the movement of one stock can affect the movement of another and the understanding that collaborative filtering is effective at making predictions based on similarity between items. The mapping of collaborative filtering logic to their stock prediction algorithm is explained in Table III.

The algorithm portion of their model first calculates the transmission effect between the before and after movements of different stocks. Then, the related companies (nearest neighbor group) of a target stock are obtained based on the stock's transmission effect coefficient. Then, then the movement of those related stocks are used to determine whether or not to recommend them. They define a target stock S_i, such that the collection of all stocks are $C_{Si} = \{S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n\}$; a movements of the target stock within the k^th long period of time T_k, containing t short period of time:

$$X_{S_i,T_k,t} = \frac{P_{S_i,T_k,t,ending} - P_{S_i,T_k,t,beginning}}{P_{S_i,T_k,t,beginning}} \quad (12)$$

Next, they calculate the Pearson correlation coefficients between that value and T_{k-1} long-period vectors: $X_{Sj,Tk-1} = [X_{Sj,Tk-1,1}, X_{Sj,Tk-1,2}, X_{Sj,Tk-1,3}, \ldots, X_{Sj,Tk-1,14}]$ and all of the stocks in C_{Si}. The Pearson correlation coefficient between these two vectors is:

$$T_{X_{s_i,T_k},X_{s_j,T_{k-1}}} = P_{X_{s_i,T_k},X_{s_j,T_{k-1}}} = \frac{cov(X_{s_i,T_k}, X_{s_j,T_{k-1}})}{\sigma_{X_{s_i,T_k}}\sigma_{X_{s_j,T_{k-1}}}} = \frac{E((X_{s_i,T_k} - \mu_{X_{s_i,T_k}})(X_{s_j,T_k} - \mu_{X_{s_j,T_{k-1}}}))}{\sigma_{X_{s_i,T_k}}\sigma_{X_{s_j,T_{k-1}}}} \quad (13)$$

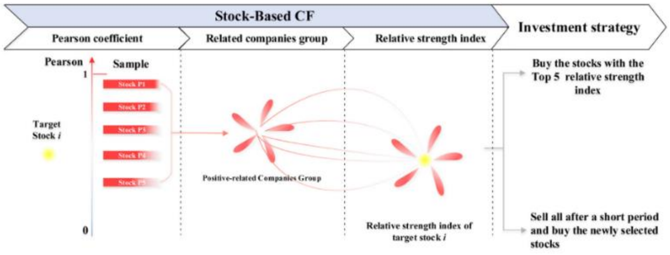
Where *cov*$(X_{Si,Tk}, X_{Sj,Tk-1})$ is the covariance, and $\sigma_{X_{Si,Tk,t}}\sigma_{X_{Sj,Tk-1,t}}$ is the product of the standard deviations of the two vectors. The linear relationship between these vectors increases as the transmission effect goes to 1 (identical) or -1 (exactly opposite). Next, they construct a group of related companies. Their test model selects the 5 stocks from C_{Si} to compose the set $C_{Si,Tk}^{Positive}$, the list of the 5 stocks with largest Pearson correlation coefficients with the target stock. Subsequently, they calculate a ranking index which measures the target stock's relative strength during the given period:

$$R_{t,i} = \sum_j^5 X_{Sj,t}^{Positive} \quad (14)$$

Where $X_{Sj,t}^{Positive}$ is the rise or fall of each stock S_j in $C_{Si,Tk}^{Positive}$ during the given short period. This equation produces a sum of ratings across N = 5 similar stocks. Note how this compares to the standard CF equation for predicting the estimated ranking of item *i* for a user *x*, given the set N of k users most similar to x who have rated item i:

$$r_{xi} = \frac{1}{k}\sum_y^N r_{yi} \quad (15)$$

Fig. 9 illustrates the structure of their final model. They tested this model over a 5-year period from 2014 to 2015. Their results demonstrate firstly that their CF-based market prediction algorithm outperformed the industry average and several leading funds. Secondly, they demonstrated significant transmission effects throughout the market, meaning that related stocks often strongly covary, suggesting the utility of a CF and potentially other recommendation system models for market prediction.


Stock-Based CF
Investment strategy
Pearson coefficient
Related companies group
Relative strength index
Pearson
1
0
Sample
Target Stock *i*
Stock P1
Stock P2
Stock P3
Stock P4
Stock P5
Positive-related Companies Group
Relative strength index of target stock *i*
Buy the stocks with the Top 5 relative strength index
Sell all after a short period and buy the newly selected stocks

a. Source: [35]

Fig. 9 Recommendation system model

Given that this model utilizes the logic of collaborative filtering, and that CF is commonly applied for recommendation systems in big data analytics, their model should be quite scalable. For big data, one way this is done is by sampling only a subset of the total data and interpolating the unsampled points. In the case of Zheng et. al., this would mean modifying their algorithm with an interpolation function. I adapt the standard functions for interpolation in CF to construct the necessary equations. The interpolation function:

$$\overline{X_{S_i,T_k,t}} = b_{S_i,T_k,t} + \sum_j^{N(i,x)} w_{ij}(X_{S_j,T_k,t} - b_{S_j,T_k,t}) \quad (16)$$

Where w_{ij} is the interpolation weight, such that $\sum_j^{N(i,x)} w_{ij} \neq 1$; $b_{S_i,T_k,t}$ is a baseline estimate for $X_{S_i,T_k,t}$, such that $b_{S_i,T_k,t} = \mu + b_{S_i} + b_{T_k,t}$; where μ is the overall mean coefficient for a given stock, b_{S_i} is the rating deviation of stock *i*, and $b_{T_k,t}$ is the average coefficient in the given short period *t*, in the given long period T_k. The interpolation weights for each *i* and *j* would in turn be calculated:

$$J(w) = \sum_{x,i}([b_{S_i,T_k,t} + \sum_j^{N(i,x)} w_{ij}(X_{S_j,T_k,t} - b_{S_j,T_k,t})] - X_{S_j,T_k,t})^2 \quad (17)$$

The limitations of sampling would also mean that this new model would need to be tested and its performance compared to that of the original model. The fewer datapoints sampled and the more points interpolated, the larger the potential margin of error and the weaker the performance of the model may be. In theory, these modifications would enable their model to sample an even wider array of assets over a longer period of time, moving us closer to the promise of a true global CF model of market performance.

IV. Challenges, Ethical Considerations, and the Future

The power and potential of big data analytics should not be taken lightly, and the plethora of new tools should not be adopted carelessly. Besides the many technical challenges we have discussed, there are also moral and ethical challenges associated with the emergence of new AI tools that should be mentioned before I conclude.

A. *Model Interpretability*

ML methods are vastly heterogeneous. Many (e.g. neural network models) are "black-box" models, meaning that there is no way to interpret *how* they came to a given conclusion (i.e.

"not interpretable") [39]. Even the interpretability of "white-box" models may be limited. These challenges concern not only the experts using these methods, but also the general public. The complexity and novelty of ML models may mislead laymen into believing that they are perfectly objective, completely accurate, and unbiased. The recent rapid advancement of ML and AI tools (and their increasing complexity) leads to the growing call for explainable models. Lipton [40] argues that current research does not provide robust definitions of what "interpretable" even means. So long as the concept of interpretability remains slippery, it will fall upon individual researchers to better define it. Industry standards could introduce uniform standards for interpretability. On the other hand, Lipton also acknowledges that prioritizing explainability may preclude any model which exceeds human capacities for complex tasks. For example, "the short-term goal of building trust with doctors by developing transparent models might clash with the longer-term goal of improving healthcare [with more powerful models]" [40].

B. Bias and Fairness

Related to explainability, it is also important to keep in mind that ML models are fundamentally human-made tools, which can reflect and perpetuate human mistakes and biases. This is as relevant in finance as it is everywhere else ML is employed. For instance, a biased ML model in finance could disproportionately deny credit card applications to customers of a particular race, or consistently assign lower credit scores to creditors of one gender [40, 41]. One high-profile example of the ways ML can perpetuate harmful human prejudices occurred in 2018, when Google's image recognition algorithm was repeatedly labeling images of dark-skin humans as gorillas—a problem which they could only fix by removing gorillas from their model altogether [41, 42].

Some recent research [41], has attempted to utilize new explainable AI (XAI) methods for detecting and mitigating biases. One such tool is Local Interpretable Model-agnostic Explanations (LIME) which aims to approximate a black-box ML model by creating a local interpretable model. The exact mechanics of such an algorithm is beyond the scope of this paper, but multiple such tools have cropped up in about the last 8 years. These tools show promise for mitigating the problems model bias poses for researchers and by extension the negative impact it can have on affected populations and the reputations of organizations employing ML [40, 41].

C. Regulatory compliance

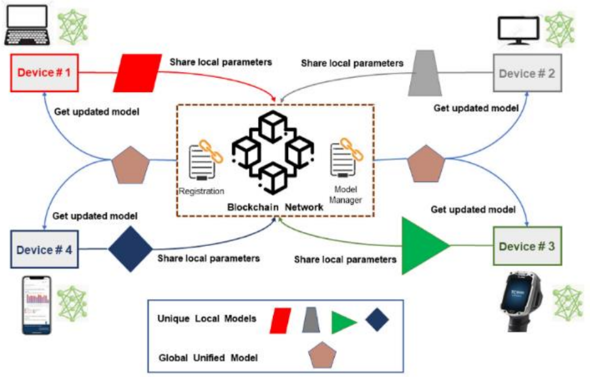
As previously discussed, ML integration in finance has garnered a wealth of concern from governing bodies and policy organizations. Their concerns range from data security (how is sensitive information stored and protected?) to market effects (the aforementioned contribution of AT to market volatility). Experts will need to develop a regulatory apparatus as dynamic as AI itself amid the widespread integration of AI across society and industry that is already in progress. Whether or not they are ready for it, legislators and industry experts are saddled with the great responsibility to design regulations and industry standards such that they maximally manage the risks and challenges of AI without stifling innovation and growth [1, 2, 40].

D. Horizons

The future of ML remains bright. While extant methods are being utilized to produce more and more powerful models, entirely new methods of ML are simultaneously being investigated. Two powerful examples of potential future avenues for ML research and development are blockchain technologies and quantum computing.

a) Blockchain

Blockchain, possibly best known for bitcoin, which is built upon blockchain, but it constitutes a much larger edifice. In short, blockchain is simply a public shared ledger. This ledger facilitates recording of transactions and tracking assets within a network. A set of "blocks" containing this data is shared between users on a blockchain network (users constituting network nodes). This structure makes it extremely difficult for hackers to change, manipulate, or delete data. Essentially anything to which a value can be assigned can be tracked and traded via a blockchain network. Proponents argue that this system facilitates faster, safer, and more efficient trade [44]. Researchers have suggested ways in which blockchain technology can be used for machine learning. These studies tend to capitalize on blockchain's decentralized nature. For instance, Shafay et. al. [46] explore the possibility of utilizing blockchain networks for deep learning algorithms. They explain how various neural network models could be trained across multiple systems in a blockchain network, pointing to existing research on "federated learning" in which multiple models are trained on multiple separate datasets across multiple systems and then integrated into a single more powerful model. The structure of such a system is illustrated in Fig. 10 [46].



a. Source: [46]

Fig. 10 Federated learning model.

By dispersing training in this way, the final model can potentially be trained faster and on more data than a model trained on a single machine. Furthermore, the security and reliability touted by proponents of blockchain technology mean that models could potentially be trained on sensitive data without compromising the data security or privacy of individuals. The example they give is in the healthcare industry, where a model might be trained in a secure server on confidential patient data, then used to update the global unified model without necessitating actually sharing any patient data over the network [46].

b) Quantum Computing

Quantum computing remains highly experimental, and the capabilities of existing quantum computers are quite limited in terms of precision, memory size (number of qubits), and clock timing (and thus error rate) [47]. Undeterred, some researchers have been investigating the possibility of machine learning via quantum computing since as early as the 1990s [49]. One more recent such paper, Farhi and Neven [48], goes so far as to propose a potential model for a "quantum neural network" (QNN) for classifying labeled data via supervised learning. Due to the extreme limitations in computing power (very few qubits per quantum processor), their model has to be designed with future potential quantum processors in mind, and has so far only been tested fully in simulation. If the clock cycle problem [47] can be overcome, and manufacturing technology progress to allow for the construction of more complex (and more powerful) quantum computers, quantum computing may be expected to revolutionize machine learning as much as it is expected to revolutionize the whole of computer science.

VII. Conclusion

In the near future, we can expect continued and even accelerating progress in ML and AI science. The commensurate impact on the economy and society is so far looking to be just as monumental [1, 2, 3, 14, 18, 19, 43]. Within the finance industry alone, machine learning is a regular part of everyday business, and newer, more powerful models are being designed and deployed every year [1, 19]. We observe the application of ML for big data across the industry, encompassing: Risk assessment and management, fraud detection and management, asset trading (including price prediction, automated trading, portfolio optimization, and sentiment analysis), and behavior analysis and personalization. I have reviewed case studies demonstrating ML methods in these areas. I observe some challenges, but more so *potential* in cutting edge ML methods for outperforming both humans and previous models across each of these domains. I have also presented potential improvements and extensions of other researchers' models in order to improve applicability with big data and data streams. These far-reaching subjects impact researchers, stakeholders, customers and the general public alike, necessitating responsible usage of ML tools now more than ever. With great power comes great responsibility. The future of ML will depend on smart, dynamic rules and standards devised in close communication with data scientists and other industry experts, alongside the logarithmically growing mass of ML and AI research.

References

[1] "Artificial Intelligence, Machine Learning and Big Data in Finance: Opportunities, Challenges and Implications for Policy Makers", *OECD*, 2021. https://www.oecd.org/finance/financial-markets/Artificial-intelligence-machine-learning-big-data-in-finance.pdf

[2] "AI Governance Alliance", *World Economic Forum*. 2023. https://initiatives.weforum.org/ai-governance-alliance/home

[3] "World Economic Forum Topics", *World Economic Forum*. 2023. https://intelligence.weforum.org/topics

[4] "Apache Hadoop 3.3.6", *Apache Software Foundation*. 2023. https://hadoop.apache.org/docs/stable/

[5] "Apache Spark 3.5.0", *Apache Software Foundation*. 2023. https://spark.apache.org/docs/3.5.0/.

[6] "Machine Learning in Finance", *Corporate Finance Institute*, 2023. https://corporatefinanceinstitute.com/resources/data-science/machine-learning-in-finance/

[7] "Finance and Financial Data", *NYU Libraries*. 2023. https://guides.nyu.edu/finance/data-sources

[8] T. Choi, H. K. Chan, X. Yue, "Recent Development in Big Data Analytics for Business Operations and Risk Management," *IEEE Transactions on Cybernetics*, vol. 47, no. 1, Jan 2017.

[9] G. E. G. Beroggi, W. A. Wallace, "Operational Risk Management: A New Paradigm for Decision Making," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 10, October 1994.

[10] A. S. Koyuncugil, N. Ozgulbas, "Financial early warning system model and data mining application for risk detection," *Expert Systems Applications*, vol. 39, no. 6, pp. 6238-6253, 2012.

[11] G. Hailemariam, S. Hill, and S. Demissie, "Exploring data mining techniques and algorithms for predicting customer loyalty and loan default risk scenarios at wisdom microfinance, Addis Ababa, Ethiopia," *Proceedings of the International Conference on Managements of Emerging Digital Ecosystems (MEDES)*, Addis Ababa, Ethiopia, 2012, pp. 183-184.

[12] H. Yu, X. Huang, X. Hu, and H. Cai, "A comparative study on data mining algorithms for individual credit risk evaluation," in Proceedings of the 4th International Conference on Managing e-Commerce and e-Government (ICMeCG), Chengdu, China, Oct. 2010, pp. 35-38.

[13] A. Deshmukh, L. Talluru, “An application of a data mining technique for assessing the risk of management fraud,” *Review of Accounting Information Systems*, vol. 2, no. 1, pp. 1-16, 1998.

[14] A. Ali, S. A. Razak, S. H. Othman, T. A. E. Eisa, A. Al-Dhaqm, M. Nasser, T. Elhassan, H. Elshafie, A. Saif, “Financial Fraud Detection Based on Machine Learning: A Systematic Literature Review,” *Applied Sciences*, vol. 12, Sept 2022.

[15] “2023 Fraud and Financial Crime Report,” *Kroll*, 2023. https://www.kroll.com/en/insights/publications/fraud-and-financial-crime-report

[16] D. Choi, K. Lee, “An Artificial Intelligence Approach to Financial Fraud Detection under IoT Environment: A Survey and Implementation,” *Security and Communication Networks*, 2018.

[17] C. Soviany, “The benefits of using artificial intelligence in payment fraud detection: A case study,” *Journal of Payments Strategy & Systems*, vol. 12, no. 2, Apr 2018.

[18] E. Boehmer, K. Fong, J. Wu, “Algorithmic Trading and Market Quality: International Evidence,” *Journal of Financial and Quantitative Analysis*, vol. 56, no. 8, Dec 2021.

[19] F. G. D. C. Ferreira, A. H. Gandomi, R. T. N. Cardoso, “Artificial Intelligence Applied to Stock Market Trading: A Review,” *IEEE Access*, Jan 2021.

[20] P. Sonkiya, V. Bajpai, A. Bansal, “Stock Price Prediction Using Artificial Intelligence: A Survey,” *IEEE*, Apr 2022.

[21] C. Li, L. Shen, G. Qian, “Online Hybrid Neural Network for Stock Price Prediction: A Case Study of High-Frequency Stock Trading in the Chinese Market,” *Econometrics*, vol. 11, no. 13, May 2023.

[22] A. Gunjan, S. Bhattacharyya, “A brief review of portfolio optimization techniques,” *Artificial Intelligence Review*, vol. 56, Sept 2023.

[23] F. Soleymani, E. Paquet, “Financial portfolio optimization with online deep reinforcement learning and restricted stacked autoencoder—DeepBreath,” *Expert Systems with Applications*, vol. 156, Apr 2022.

[24] M. B. Schrettenbrunner, “Artificial-Intelligence-Driven Management: Autonomous Real-Time Trading and Testing of Portfolio or Inventory Strategies,” *IEEE Engineering Management Review*, vol. 51, no. 3, Sept 2023.

[25] H. Haddadian, M. B. Haskuee, G. Zomordian, “A Hybrid Artificial Intelligence Approach to Portfolio Management,” *Iranian Journal of Finance*, vol. 6, no. 1, pp. 1-27, Feb 2022.

[26] Ya I. Subbotin, H. Badkoobehi, N. N. Bilotckii, “Application of fuzzy logic to learning assessment,” *Didactics of Mathematics: Problems and Investigations*, vol. 22, pp. 38-41.

[27] N. Van Otten, “What is the Elman Neural Network?” *Spot Intelligence*, Feb 1 2023.

[28] R. Grosse, “Lecture 15: Exploding and Vanishing Gradients,” *University of Toronto*, 2017. https://www.cs.toronto.edu/~rgrosse/courses/csc321_2017/readings/L15%20Exploding%20and%20Vanishing%20Gradients.pdf

[29] S. Sohangir, D. Wang, A. Pomeranets, T. M. Khoshgoftaar, “Big Data: Deep Learning for Financial Sentiment Analysis,” *Journal of Big Data*, vol. 5, no. 3, 2018.

[30] C. Qian, N. Mathur, N. H. Zakaria, R. Arora, V. Gupta, M. Ali, “Understanding public opinions on social media for financial sentiment analysis using AI-based techniques,” *Information Processing and Management*, vol. 59, Sept 2022.

[31] M. K. Agoraki, N. Aslanidis, G. P. Kouretas, “U.S. Banks’ lending, financial stability, and text-based sentiment analysis,” *Journal of Economic Behavior and Organization*, vol. 197, pp. 73-90, Mar 2022.

[32] G. Wang, T. Wang, B. Want, D. Sambasivan, Z. Zhang, H. Zheng, B. Zhao, “Crowds on wall street: extracting values from social investing platforms, foundations and trends in information retrieval,” *New York: ACM*, 2014.

[33] M. Sharaf, E. Hemdan, A. El-Sayed, N. El-Bahnasawy, “A survey on recommendation systems for financial services,” *Multimedia Tools and Applications*, vo. 81, pp. 16761-16781, March 2022.

[34] V. Vismayaa, K. Pooja, A. Alekhya, C. Malavika, B. Nair, P. Kumar, “Systems for Indian Stocks: An Empirical Evaluation,” *Computational Economics, Springer; Society for Computational Economics*, vol. 55, no. 3, 2020.

[35] Z. Zheng, Y. Gao, L. Yin, M. Rabarison, “Modeling and analysis of a stock-based collaborative filtering algorithm for the Chinese stock market,” *Expert Systems with Applications*, vol. 162, no. 7, Oct 2019.

[36] “What is machine learning?” *IBM*, 2023, https://www.ibm.com/topics/machine-learning

[37] S. Brown, “Machine learning, explained,” *MIT Sloan*, April 24, 2021. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

[38] S. Yang, “The Global Organizational Behavior Analysis for Financial Risk Management Utilizing Artificial Intelligence,” *Journal of Global Information Management*, vol. 30, no. 7, 2022.

[39] J. W. Yeo, W. Van Der Heever, M. Rui, E. Cambria, R. Satapathy, G. Mengaldo, “A Comprehensive Review on Financial Explainable AI,” *Journal of the Association for Computing Machinery*, vol. 37, no. 4, Aug 2023.

[40] Z. C. Lipton, “The Mythos of Model Interpretability,” *ACM Queue*, vol. 16, no. 3, June 2018.

[41] S. Mohseni, N. Zarei, E. Ragan, “A multidisciplinary survey and framework for design and evaluation of

explainable AI systems," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 11, no. 3-4, pp. 1-45, 2021.

[42] J. Vincent, "Google 'fixed' its racist algorithm by removing gorillas from its image-labeling tech." *The Verge*, Dec 2018.

[43] J. W. Goodell, S. Kumar, W. M. Lim, D. Pattnaik, "Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis," *Journal of Behavioral and Experimental Finance*, vol. 32, Aug 2021.

[44] M. Gupta, *Blockchain for Dummies*. Hoboken, NJ: John Wiley & Sons, Inc.; IBM, 2023.

[45] S. Ahmed, M. M. Alshater, A. El Ammari, H. Hammami, "Artificial intelligence and machine learning in finance: A bibliometric review," *Research in International Business and Finance*, vol. 61, Apr 2022.

[46] M. Shafay, R. Ahmad, K. Salah, I. Yaqoob, R. Jayaraman, M. Omar. "Blockchain for deep learning: review and open challenges," *Cluster Computing*, vol. 26, pp. 197-221, 2023.

[47] Vienna University of Technology, "Limits for quantum computers: Perfect clocks are impossible, research finds," *Phys.org*, Nov 26, 2023.

[48] E. Farhi, H. Neven, "Classification with Quantum Neural Networks on Near Term Processors," *MIT*, 2018.

[49] S. Kak, "Quantum Neural Computing," *Advances in Imaging and Electron Physics*, vol. 94, pp. 259-313, 1995.