AI

Review

Deep Learning in Finance: A Survey of Applications and Techniques

Ebikella Mienye [1,†], Nobert Jere [2,†], George Obaido [3,†], Ibomoiye Domor Mienye [4,*,†] and Kehinde Aruleba [5,†]

1 College of Business and Economics, University of Johannesburg, Johannesburg 2003, South Africa; 219105099@student.uj.ac.za
2 Department of Computer Science, University of Fort Hare, Alice Campus, Alice 5700, South Africa; njere@ufh.ac.za
3 Center for Human-Compatible Artificial Intelligence (CHAI), Berkeley Institute for Data Science (BIDS), University of California, Berkeley, CA 94720, USA; gobaido@berkeley.edu
4 Institute for Intelligent Systems, University of Johannesburg, Johannesburg 2006, South Africa
5 School of Computing and Mathematical Sciences, University of Leicester, Leicester LE1 7RH, UK; ka388@leicester.ac.uk
* Correspondence: ibomoiyem@uj.ac.za
† These authors contributed equally to this work.

Abstract: Machine learning (ML) has transformed the financial industry by enabling advanced applications such as credit scoring, fraud detection, and market forecasting. At the core of this transformation is deep learning (DL), a subset of ML that is robust in processing and analyzing complex and large datasets. This paper provides a comprehensive overview of key deep learning models, including Convolutional Neural Networks (CNNs), Long Short-Term Memory networks (LSTMs), Deep Belief Networks (DBNs), Transformers, Generative Adversarial Networks (GANs), and Deep Reinforcement Learning (Deep RL). Beyond summarizing their mathematical foundations and learning processes, this study offers new insights into how these models are applied in real-world financial contexts, highlighting their specific advantages and limitations in tasks such as algorithmic trading, risk management, and portfolio optimization. It also examines recent advances and emerging trends in the financial industry alongside critical challenges such as data quality, model interpretability, and computational complexity. These insights can guide future research directions toward developing more efficient, robust, and explainable financial models that address the evolving needs of the financial sector.

Keywords: CNN; deep learning; finance; GRU; LSTM; machine learning; RNN

Citation: Mienye, E.; Jere, N.; Obaido, G.; Mienye, I.D.; Aruleba, K. Deep Learning in Finance: A Survey of Applications and Techniques. *AI* **2024**, *5*, 2066–2091. https://doi.org/10.3390/ai5040101

Academic Editors: Xianrong (Shawn) Zheng and Demos T. Tsahalis

Received: 18 August 2024
Revised: 16 October 2024
Accepted: 23 October 2024
Published: 28 October 2024

1. Introduction

The integration of artificial intelligence (AI), particularly deep learning (DL), into financial systems has significantly transformed the finance industry. Deep learning's ability to process and analyze vast arrays of data has led to breakthroughs in areas such as credit scoring, fraud detection, and algorithmic trading [1,2]. These advancements have improved accuracy and enabled the development of more sophisticated financial tools and services. Despite these advancements, the deployment of deep learning in finance is not without its challenges, such as the interpretability of DL models, their demand for large amounts of data, and the need for high computational power [3,4]. These challenges highlight the need for models that not only excel in performance but also offer transparency, scalability, and compliance with financial regulations.

Recent reviews have explored various aspects of AI in finance; however, they have often focused broadly on machine learning (ML) without delving deep into the specific applications and intricacies of DL models. For example, the reviews by Ahmed et al. [5] and Goodell et al. [6] explored the applications of ML in finance, covering financial fraud, bankruptcy prediction, stock price prediction, and portfolio management. Furthermore, some studies have addressed the technical capabilities and applications of deep learning in specific financial applications. For example, Mienye and Jere [4] reviewed the application of DL architectures in credit card fraud detection, including the challenges encountered in deploying these models in real-world applications. Similarly, Gunnarsson et al. [7] reviewed deep learning with application to credit scoring, suggesting best practices to ensure optimal utilization of DL models. Other deep learning reviews include those applied to stock market prediction [8] and algorithmic trading [9]. However, these studies often overlook the detailed discussion of how advancements in modern deep learning architectures, like Transformers and generative adversarial networks (GANs), can offer novel approaches for overcoming challenges specific to the financial sector, such as non-stationary data and sudden market shifts.

Over the years, there have been numerous advances in deep learning architectures, such as advancements in Transformer architectures, GANs, and deep reinforcement learning models [10]. These methods have played vital roles in various technological advancements and have been applied in different financial applications. For instance, Transformer models have revolutionized natural language processing tasks, which are critical in sentiment analysis and financial document analysis [11]. GANs have been effectively used in generating synthetic financial data, thereby enhancing the robustness of models trained on limited datasets [12]. Deep reinforcement learning, on the other hand, has shown promise in optimizing trading strategies and portfolio management [13]. Although these models have demonstrated significant potential, there remains a pressing need to evaluate their application in complex financial scenarios, where model performance must balance both prediction accuracy and risk management. Due to these recent advances, there is a need for an up-to-date review that consolidates these developments and critically examines their implications and potential in the financial sector.

Therefore, this paper provides a comprehensive review of deep learning and its applications in the financial industry. The study aims to critically assess the inner workings of different DL architectures and their effectiveness and explore the challenges they present in financial contexts. Unlike previous reviews, this paper aims to go beyond simply summarizing established methods by offering insights into how emerging DL architectures can address the unique challenges posed by the financial sector, such as regulatory compliance and real-time data analysis. The goal of this review is to provide a robust analysis that highlights the current state of deep learning in finance and identifies areas where further research and development are needed. This will be beneficial to both deep learning researchers and industry professionals aiming to harness these technologies for financial applications.

The remainder of this paper is organized as follows. Section 2 reviews related works, highlighting their contributions and limitations. Sections 3 and 4 discuss the various deep learning models used in finance and their applications, respectively. Section 5 presents recent advances and emerging trends in deep learning. Section 6 discusses challenges limiting deep learning applications in finance. Section 7 highlights future research directions, while Section 8 concludes the study.

2. Related Works

Recently, there have been significant advancements in the application of deep learning within financial data modeling. For example, Wang et al. [14] highlighted the effectiveness of sequence-to-sequence models for predicting market movements, offering sophisticated tools for algorithmic trading. However, their work does not explore how these models can be tailored to different financial environments, a critical factor for effective deployment. Risk management is another critical area where deep learning has made a substantial impact. Meng et al. [15] explored the application of CNNs in identifying high-risk patterns, thus aiding preemptive measures against financial instability. These models are well-suited for analyzing large volumes of unstructured data, which is common in financial datasets.

Nonetheless, the scalability of CNNs for larger financial institutions and complex datasets remains underexplored.

Credit scoring has also benefited from deep learning. Khandani et al. [16] demonstrated how deep learning could enhance the precision of credit scoring systems, essential for evaluating the creditworthiness of potential borrowers. Similarly, Esenegho et al. [17] employed a deep learning ensemble for credit risk and fraud detection. The study used the long short-term memory (LSTM) network as the base learner in the adaptive boosting (AdaBoost) implementation, achieving excellent performance. Mienye and Sun [18] proposed a deep ensemble learning method for credit card fraud detection using LSTM and gated recurrent units (GRUs) as base learners and a multilayer perceptron (MLP) as the meta-learner, achieving classification performance that outperformed benchmark models. However, the specific challenges these models face, such as overfitting and interpretability, were not deeply examined in these works.

Additionally, Sezer and Ozbayoglu [19] provided a comprehensive review of deep learning applications in finance, covering areas such as fraud detection, algorithmic trading, and portfolio management. Their study illustrates the adaptability of DL models and their potential to transform many aspects of the financial industry. However, their review did not address how the advancements in DL architectures, particularly for Transformers, GANs, and Deep RL, can be integrated into these applications to further enhance performance and scalability. Their review primarily focused on established deep learning techniques like CNN, LSTM, and deep belief networks (DBNs), with limited coverage of the latest advancements such as Transformer models, GANs, and Deep RL, which have shown great promise in recent years. Another notable review by Lim et al. [20] focused on deep learning's role in time series forecasting, such as financial markets. While their analysis was thorough in the context of traditional ML and DL, they did not explore the application of advanced DL techniques in high-frequency trading and its complexities, a growing area of interest.

Moreover, a study by Goodell et al. [6] provided an extensive overview of AI applications in finance, including ML and DL. The study grouped all AI and ML methods together without a specific focus on the unique contributions and challenges of deep learning in finance. As a result, the review did not adequately cover the intricacies of deep learning models nor the specific advancements in architectures like GANs and Deep RL, which are crucial for understanding the current deep learning in finance. Their approach limited the discussion on how GANs, for example, can generate synthetic financial data to improve model training under data scarcity, or how Deep RL models can optimize trading strategies in dynamic market environments.

Recent work by Mienye and Jere [4] concentrated on the challenges and applications of DL in credit card fraud detection. Although the study provided valuable insights into the practical deployment of deep learning models, it was narrowly focused on a specific application, leaving out a broader discussion of other emerging deep learning techniques and their implications for the financial industry as a whole. A more in-depth examination of how these models could be generalized across different financial sectors would have provided a more complete perspective.

Given these limitations in the existing literature, there is a gap in comprehensive reviews that not only cover traditional DL applications in finance but also explore the recent advances in DL architectures, such as Transformers, GANs, and Deep RL. These modern techniques, which offer significant improvements in scalability, interpretability, and real-time decision making, are transforming how financial institutions manage risks, make predictions, and automate processes. These modern techniques are reshaping the financial industry, offering new methods for analyzing complex datasets, predicting market movements, and managing financial risk.

Therefore, this paper aims to bridge this gap by providing an up-to-date review of deep learning applications in the financial industry, focusing on the latest advancements and how they can address current challenges in finance. This approach will provide a deeper understanding of the intricacies of applying DL architectures, particularly in environments that demand explainability and adaptability. This review is timely, given the rapid evolution of DL technologies and their increasing impact on financial systems, and it seeks to guide future research and development in this dynamic field.

3. Deep Learning Architectures

Deep learning models have significantly impacted financial data modeling, offering advanced solutions for analyzing and predicting financial variables. These DL models are discussed in this section.

3.1. Feedforward Neural Networks

Feedforward Neural Networks (FNNs), also known as MLPs, are the most basic type of artificial neural networks. They consist of an input layer, one or more hidden layers, and an output layer, with each node in one layer connected only to nodes in the subsequent layer [21]. This unidirectional flow of data—from the input layer to the output layer—distinguishes FNNs from other types of neural networks that may have connections cycling back to previous layers. In an FNN, each node (or neuron) in a layer computes a weighted sum of its inputs, adds a bias term, and then applies an activation function to produce its output. Mathematically, the output y of a neuron can be expressed as:

$$y = \sigma(Wx + b) \tag{1}$$

where x represents the input vector, W is the weight matrix, b is the bias vector, and σ is the activation function (commonly a ReLU or sigmoid function) [22]. The activation function introduces non-linearity into the model, allowing the network to learn complex patterns within the data [23]. FNNs are very useful in finance for tasks such as credit scoring, bankruptcy prediction, and customer segmentation, where the relationships between input variables and the output can be learned through straightforward mapping. Their simplicity and efficiency make them suitable for these applications, especially when the goal is to classify or predict outcomes based on historical financial data. The training process of an FNN involves adjusting the weights W and biases b to minimize a loss function, typically using a method like stochastic gradient descent (SGD). Algorithm 1 outlines the key steps in the training process of an FNN.

Algorithm 1 Training a Feedforward Neural Network.

Input: Training dataset $\{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$
Initialize: Weights W and biases b randomly
for each epoch **do**
  for each training sample $(x^{(i)}, y^{(i)})$ **do**
    Forward Pass:
      Compute the input to each neuron in the hidden layers and the output layer:
        $a^{(l)} = W^{(l)} x^{(l-1)} + b^{(l)}$
      Apply activation function: $h^{(l)} = \sigma(a^{(l)})$
      Compute output of the network: $y^{\text{pred}} = h^{\text{output}}$
    Backpropagation:
      Compute the loss: $L(y^{\text{pred}}, y^{(i)})$
      Compute gradients $\nabla W^{(l)}$ and $\nabla b^{(l)}$ using backpropagation
      Update weights and biases:
        $W^{(l)} \leftarrow W^{(l)} - \eta \nabla W^{(l)}$
        $b^{(l)} \leftarrow b^{(l)} - \eta \nabla b^{(l)}$
  end for
end for

3.2. Simple Recurrent Neural Networks

RNNs are a type of neural network designed to process sequential data by maintaining a hidden state that captures information from previous elements in the sequence [24]. Unlike feedforward neural networks, RNNs have recurrent connections that allow them to remember past inputs, making them effective for tasks involving time-dependent data such as stock price forecasting, natural language processing, and algorithmic trading [25]. The RNN architecture is shown in Figure 1. At each time step t, an RNN takes the current input x_t and the hidden state from the previous time step h_{t-1} to compute the current hidden state h_t. The hidden state is updated using the following equation:

$$h_t = \sigma(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \tag{2}$$

where W_{hx} and W_{hh} are weight matrices, b_h is the bias, and σ is a non-linear activation function, such as the hyperbolic tangent (tanh) or Rectified Linear Unit (ReLU). The hidden state h_t serves as a summary of all past inputs up to time t, allowing the RNN to model temporal dependencies in the data [26]. This capability is crucial in financial applications where the value of an asset is influenced by its past behavior and other sequential factors. Meanwhile, despite their effectiveness, the simple RNN face a significant challenge known as the vanishing gradient problem, which arises during backpropagation. When training on long sequences, the gradients of the loss function with respect to the network parameters can diminish to near zero, hindering the model's ability to learn long-term dependencies [27]. This limitation reduces the RNN's effectiveness in capturing extended patterns in data, which is critical in many financial tasks. However, there are more advanced RNN architectures like LSTM networks and GRU, which are specifically designed to retain information over longer sequences.

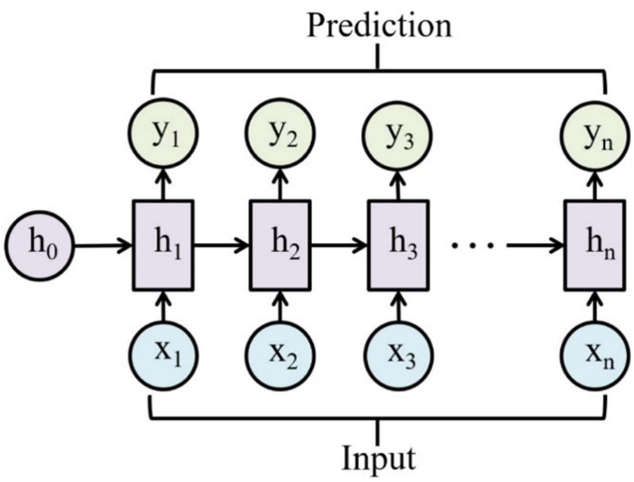

Figure 1. RNN architecture [4].

3.3. Long Short-Term Memory Networks

LSTMs are designed to overcome the vanishing gradient problem that can occur in traditional RNNs by incorporating gates that regulate the flow of information [28]. These gates ensure that the network can maintain long-term dependencies in the data, which is critical for applications like sequential prediction, where context from far back in the sequence is important. The LSTM unit, shown in Figure 2, includes input, forget, and output gates, together with a cell state that carries information across time steps [17]. The equations governing these components are:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i), \tag{3}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f), \tag{4}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o), \tag{5}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{xg}\mathbf{x}_t + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_g), \quad (6)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \quad (7)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (8)$$

where $\mathbf{i}_t$, $\mathbf{f}_t$, and $\mathbf{o}_t$ are the input, forget, and output gates, respectively, and $\mathbf{c}_t$ represents the cell state at time t. The term σ denotes the sigmoid activation function, which squashes the input values to a range between 0 and 1, determining the extent to which information should be allowed through each gate. The cell state $\mathbf{c}_t$ acts as the memory of the network, carrying forward relevant information across time steps. The operator $\odot$ represents the element-wise (Hadamard) product, which controls how much of the past information is retained in the cell state [18]. This gating mechanism allows LSTMs to effectively manage long-term dependencies in sequential data, making them highly effective for financial time series analysis, anomaly detection in transaction data, and complex decision-making processes.

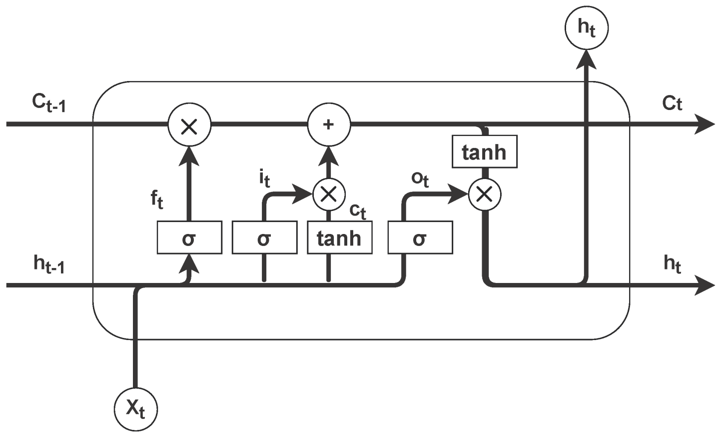

Figure 2. LSTM architecture [29].

3.4. Gated Recurrent Units

GRUs simplify the LSTM design by combining the forget and input gates into a single update gate and by merging the cell state and hidden state into one [26]. This reduction in complexity can lead to faster training times without a significant drop in performance, making GRUs a popular choice for tasks where efficiency is crucial. The architecture of a GRU is shown in Figure 3. Meanwhile, the GRU uses the following set of update equations:

$$\mathbf{z}_t = \sigma(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{W}_{hz}\mathbf{h}_{t-1} + \mathbf{b}_z), \quad (9)$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_{xr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1} + \mathbf{b}_r), \quad (10)$$

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{r}_t \odot (\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h)), \quad (11)$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}}_t, \quad (12)$$

where $\mathbf{z}_t$ is the update gate, determining how much of the previous hidden state $\mathbf{h}_{t-1}$ is carried forward to the current hidden state $\mathbf{h}_t$, the reset gate $\mathbf{r}_t$ controls how much of the previous hidden state contributes to the candidate hidden state $\tilde{\mathbf{h}}_t$, the term σ represents the sigmoid activation function, which outputs values between 0 and 1, controlling the influence of previous states, the candidate hidden state $\tilde{\mathbf{h}}_t$ is a potential new state influenced by the reset gate and the input at time t, while the final hidden state $\mathbf{h}_t$ is a combination of the previous hidden state and the candidate hidden state, modulated by the update gate, and the operator $\odot$ denotes the element-wise product [26].

GRUs have proven effective in various financial applications, including predictive analytics for stock prices, loan default likelihood, and identifying patterns in high-frequency trading data [30]. Their simplified architecture makes them suitable for scenarios where computational efficiency is crucial without compromising accuracy. For instance, in stock price prediction, GRUs can capture and model temporal dependencies within financial time series data, allowing for more accurate forecasts. The reset and update gates help the model maintain relevant historical information while discarding noise, which is valuable in volatile markets where only certain past events may be predictive of future price movements. With respect to credit risk assessment, GRUs can be used to predict loan defaults by analyzing the sequential behavior of borrowers, such as payment histories and transaction patterns. The efficiency of GRUs allows them to be trained on large datasets, ensuring that they can process and learn from vast amounts of borrower information quickly. This capability is essential for real-time risk assessment, where decisions must be made rapidly based on the latest available data.

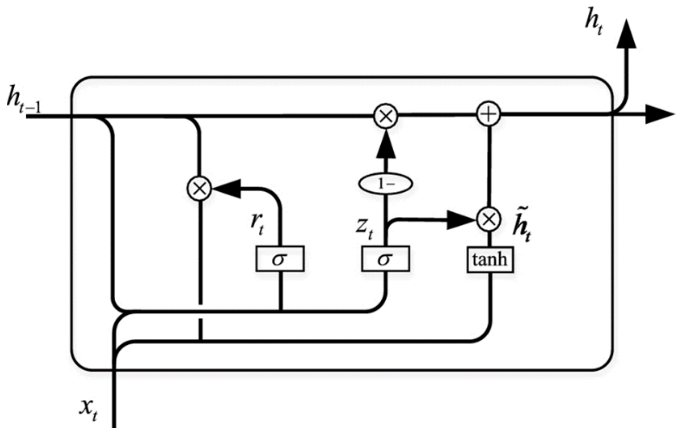

Figure 3. GRU architecture [4].

3.5. Convolutional Neural Networks

CNNs are a class of deep learning models originally designed for processing grid-like data, such as images and videos. The architecture of CNNs is particularly effective for tasks that involve identifying spatial hierarchies in data through the application of convolutional layers. Although CNNs are most commonly associated with image processing, their powerful pattern recognition capabilities have made them highly suitable for various financial applications, especially in areas like anomaly detection, fraud detection, and financial time series analysis. The CNN architecture is shown in Figure 4.

The fundamental building block of a CNN is the convolutional layer, where the convolution operation is performed. The convolution operation involves sliding a filter (also known as a kernel) across the input data to produce feature maps [31]. The mathematical operation for a convolution in one dimension is defined as:

$$z = \sigma\left(\sum_{i=1}^{k} W_i \cdot x_{i:i+n-1} + b\right) \tag{13}$$

where W_i represents the weight of the filter, $x_{i:i+n-1}$ is the segment of the input data over which the filter is applied, b is the bias term, and σ denotes the activation function, typically a ReLU [31]. The result of the convolution operation is the feature map z, which highlights specific patterns in the input data based on the learned filter weights. Meanwhile, the pooling layers perform down-sampling operations to reduce the dimensionality of the feature maps, thereby focusing on the most critical features and reducing the computational load [31]. The pooling operation is usually defined as:

$$z_{pool} = \max(x_{i:i+n-1}) \tag{14}$$

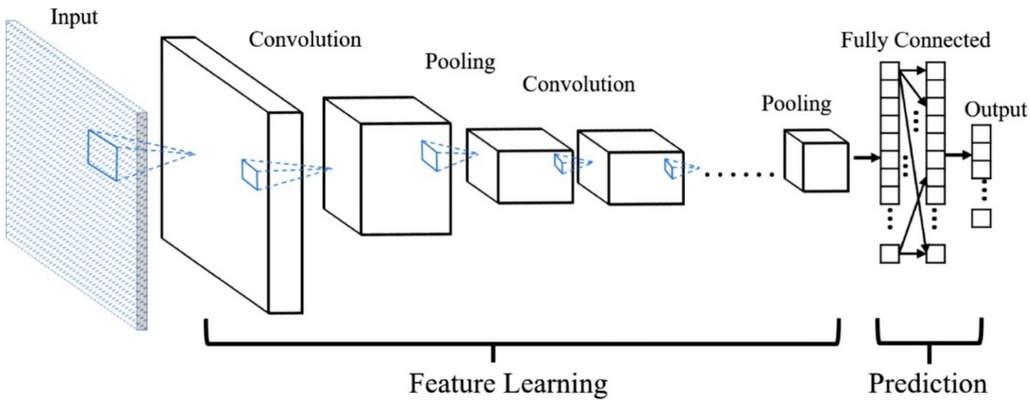

Figure 4. CNN architecture [32].

In the case of max pooling, the operation selects the maximum value from a segment of the feature map. This helps in making the model more robust to small changes in the input, which is crucial in financial applications where noise and minor fluctuations are common in the data [33]. Furthermore, the fully connected layer is typically added after several convolutional and pooling layers. These layers integrate the features extracted by the convolutional layers to perform tasks such as classification (e.g., determining whether a transaction is fraudulent) or regression (e.g., predicting the likelihood of default). The output from the fully connected layer can be given by:

$$y = \sigma(W_{fc} \cdot z_{flattened} + b_{fc}) \tag{15}$$

where W_{fc} and b_{fc} are the weights and biases of the fully connected layer, and $z_{flattened}$ represents the flattened output from the previous layer, which is a one-dimensional vector. The applications of CNNs in the financial domain extend beyond anomaly detection and fraud detection. CNNs have been employed in algorithmic trading to analyze market data, including order book data, where the model learns to predict price movements by identifying patterns in the order flow [31]. CNNs are also used in credit scoring systems, where they can process large volumes of borrower data to identify risk factors and make accurate predictions about creditworthiness [34].

3.6. Transformers and Attention Mechanisms

Transformers are a powerful class of DL models that have significantly impacted various domains, mostly natural language processing (NLP), due to their ability to efficiently handle large sets of time-dependent data. The core innovation of Transformers is in their use of attention mechanisms, which allow the model to dynamically weigh the importance of different parts of the input data, enabling the capture of long-range dependencies without the need for sequential processing, as required by RNNs [35]. The attention mechanism, a fundamental component of Transformers, can be mathematically described as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{16}$$

where Q (queries), K (keys), and V (values) are linear projections of the input data, with $Q = W_q X$, $K = W_k X$, and $V = W_v X$, where W_q, W_k, and W_v are learned weight matrices. The dot product QK^T calculates the similarity between queries and keys, and the softmax function normalizes these into attention weights, allowing the model to focus on the most relevant parts of the input [35]. Algorithm 2 summarizes the key steps in the Transformer model.

Furthermore, a major advantage of this architecture is its parallelizability, making it highly scalable and efficient for processing large datasets, which is advantageous in financial applications such as market movement prediction, risk assessment, and sentiment analysis. For example, in market movement prediction, Transformers can analyze sequences of historical price data, economic indicators, and textual data from news articles or social media [36]. The flexibility of Transformers also extends to other financial tasks such as sentiment analysis, where they analyze textual data to gauge market sentiment, and portfolio optimization, where their ability to capture dependencies across multiple time periods ensures more informed investment strategies.

Algorithm 2 Training a Transformer Model.

1: **Input:** Sequence of data points $X = \{x_1, x_2, \ldots, x_T\}$
2: **Initialize:** Parameters W_q, W_k, W_v, and other model weights
3: **for each** layer in the Transformer **do**
4: Compute queries: $Q = W_q X$
5: Compute keys: $K = W_k X$
6: Compute values: $V = W_v X$
7: Compute attention scores: $A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)$
8: Compute weighted values: $Z = AV$
9: Pass Z through feedforward network layers
10: **end for**
11: **Output:** Predicted output based on the final Transformer layer

3.7. Generative Adversarial Networks

GANs are a class of DL models that consist of two neural networks: a generator and a discriminator. These networks are trained simultaneously in a competitive setting, where the generator aims to create realistic data instances while the discriminator attempts to distinguish between real data (from the actual dataset) and fake data (produced by the generator) [37]. The adversarial nature of this training process forces the generator to produce increasingly realistic data over time. Meanwhile, the generator takes a random noise vector z from a latent space (usually sampled from a standard normal distribution) and transforms it into a data instance $G(z)$ that resembles the real data. The goal of the generator is to fool the discriminator by producing data that is indistinguishable from the real data. Conversely, the discriminator receives both real data x and generated data $G(z)$. It outputs a probability $D(x)$ or $D(G(z))$, indicating whether the input data are real or fake [37]. The discriminator is trained to correctly classify the real data as "real" and the generated data as "fake".

The objective of the GAN is expressed as a minimax game, where the generator and discriminator are pitted against each other. The generator tries to minimize the probability that the discriminator correctly classifies its outputs as fake, while the discriminator tries to maximize this probability. The overall objective function $V(D, G)$ is given by:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{17}$$

where $p_{data}(x)$ is the distribution of the real data and $p_z(z)$ is the distribution of the noise input to the generator. The steps in Algorithm 3 describe the training procedure of a GAN.

GANs have proven to be highly effective in various financial applications due to their ability to generate realistic synthetic data and model complex distributions. One prominent application is in the generation of synthetic financial datasets [38]. Financial data, especially in areas like credit scoring or market transactions, are often scarce or imbalanced. GANs can generate additional synthetic data points that resemble the original dataset, which can be used to augment training datasets, thus reducing the risk of overfitting and improving model robustness. For example, in credit risk modeling, GANs can generate synthetic borrower profiles that maintain the statistical properties of the original dataset. Similarly, in portfolio management, GANs have been used to simulate various market scenarios, helping investors to assess the robustness of different investment strategies under different market conditions [39].

Algorithm 3 Training a Generative Adversarial Network.

Initialize generator G and discriminator D with random weights.
while not converged **do**
  for each training step **do**
    Sample a minibatch of m noise samples $\{z^{(1)}, \ldots, z^{(m)}\}$ from the noise prior $p_z(z)$.
    Sample a minibatch of m real data samples $\{x^{(1)}, \ldots, x^{(m)}\}$ from the data distribution $p_{data}(x)$.
    Compute the discriminator loss:

$$L_D = -\frac{1}{m}\sum_{i=1}^{m}\left[\log D(x^{(i)}) + \log(1 - D(G(z^{(i)})))\right]$$

    Update the discriminator by performing a gradient ascent step on L_D.
    Compute the generator loss:

$$L_G = -\frac{1}{m}\sum_{i=1}^{m}\log(D(G(z^{(i)})))$$

    Update the generator by performing a gradient descent step on L_G.
  end for
end while

3.8. Deep Reinforcement Learning

Deep reinforcement learning (Deep RL) combines the principles of reinforcement learning with deep learning to tackle complex decision-making problems, particularly in environments with large state or action spaces [40]. In Deep RL, DNNs are employed as function approximators to represent the policy or value functions, enabling the agent to learn optimal strategies even in high-dimensional spaces, such as those found in financial markets. Deep RL has been effective in optimizing trading strategies, portfolio management, and risk management [41]. Unlike traditional RL, where simpler function approximators like tables or linear models might be used, Deep RL leverages the representational power of DNNs to model complex relationships within data. Meanwhile, Deep RL problems are often modeled as Markov decision processes (MDPs), defined by the tuple (S, A, P, R, γ), where:

- S represents the set of possible states the agent can be in, such as different market conditions;
- A represents the set of possible actions the agent can take, such as buying, selling, or holding assets;
- P is the state transition probability, which defines the probability of moving from one state to another given an action;
- R is the reward function, which assigns a reward to each state–action pair, reflecting the profitability of an action in a given state;
- γ is the discount factor, which determines the importance of future rewards.

The agent's goal is to learn a policy π that maximizes the expected cumulative reward, defined as:

$$G_t = \mathbb{E}\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right], \tag{18}$$

where G_t is the return at time step t and π is the policy that maps states to actions. The policy π or value function $V(s)$ is typically represented by a deep neural network, which is trained using algorithms like Deep Q-Networks (DQN), Proximal Policy Optimization (PPO), or Actor-Critic methods. Deep RL is particularly well-suited for solving complex decision-making problems, such as real-time trading and portfolio optimization, where decisions must be made under uncertainty and dynamic market conditions. Financial markets are characterized by high volatility and the need for immediate responses to changes, making traditional rule-based models ineffective. Deep RL's ability to learn from historical data, simulate multiple trading scenarios, and adapt to evolving market dynamics allows it to outperform conventional models. Moreover, Deep RL can be employed to optimize multi-period investment strategies, continuously adjusting asset allocations to maximize long-term returns while managing risk, which is critical in financial contexts. These attributes make Deep RL a powerful tool for addressing key financial challenges.

3.9. Deep Belief Networks

Deep Belief Networks (DBNs) are a class of generative DL models composed of multiple layers of stochastic, latent variables. These layers are typically Restricted Boltzmann Machines (RBMs), where each layer serves as a feature detector for the layer above it [42]. DBNs are trained in a layer-wise manner, where each RBM is trained to model the data distribution of the inputs it receives. Once trained, these layers can be stacked to form a deep network that captures complex patterns in the data. The architecture of a DBN begins with a visible layer, which directly interacts with the input data, and is followed by one or more hidden layers that learn hierarchical representations of the data. The main advantage of DBNs is in their ability to pre-train each layer as an RBM before fine-tuning the entire network using backpropagation [43]. This pre-training assists in overcoming issues such as poor initialization and vanishing gradients, which can hinder the performance of deep neural networks. Meanwhile, an RBM is a type of Markov Random Field that consists of a visible layer $\mathbf{v}$ and a hidden layer $\mathbf{h}$, where the joint distribution $P(\mathbf{v}, \mathbf{h})$ is defined as:

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})), \tag{19}$$

where $E(\mathbf{v}, \mathbf{h})$ is the energy function, and Z is the partition function. The energy function for an RBM is typically defined as:

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^T \mathbf{W} \mathbf{h} - \mathbf{b}^T \mathbf{v} - \mathbf{c}^T \mathbf{h}, \tag{20}$$

where $\mathbf{W}$ is the weight matrix between the visible and hidden layers, $\mathbf{b}$ is the bias vector for the visible layer, and $\mathbf{c}$ is the bias vector for the hidden layer [44]. DBNs have been successfully applied in various financial applications, particularly in tasks that require the modeling of complex, high-dimensional data distributions. For example, DBNs can be used for credit risk assessment, where they model the underlying patterns in borrower behavior and financial histories. By learning a hierarchical representation of the data, DBNs can capture subtle patterns that traditional models might miss, leading to more accurate predictions of creditworthiness. Despite their powerful modeling capabilities, DBNs are not without challenges. One of the main difficulties is in the training process, which can be computationally expensive, especially when dealing with very deep networks. Additionally, while DBNs can model complex data distributions, they may still suffer from issues related to scalability and overfitting, particularly when applied to large financial datasets [45].

3.10. Comparison of Deep Learning Models

Selecting the appropriate DL model for financial applications depends on several factors, such as the nature of the data, the task at hand, and the desired trade-off between accuracy, interpretability, and computational efficiency. While some models, like FNNs, excel in static tasks like credit scoring, others, like RNNs and their variants (LSTMs and GRUs), are more suited to sequential data, such as stock price forecasting. Models like CNNs and GANs are better suited to specialized tasks like anomaly detection and synthetic data generation, respectively, [46].

For example, FNNs are simple and computationally efficient but may struggle with tasks that require capturing sequential dependencies or handling unstructured data. In contrast, RNNs, LSTMs, and GRUs excel at tasks involving time-series data, though they may require more computational resources and are susceptible to issues like the vanishing gradient problem. CNNs, although primarily designed for image processing, have found their place in financial applications such as pattern recognition. Transformers offer further advancements, particularly for tasks that require attention to specific parts of the input data, such as financial sentiment analysis from text. Deep reinforcement learning excels in dynamic decision-making tasks, such as portfolio management and trading strategy optimization, but demands significant computational power and extensive training data [47]. Table 1 below summarizes the key advantages and disadvantages of the various DL models covered in this review, focusing on their application in the financial domain. This comparison can help practitioners and researchers choose the most appropriate architecture based on their specific needs and constraints.

Table 1. Comparison of DL models in financial applications.

Model	Advantages	Disadvantages	Suitable Applications
FNN	Simple, computationally efficient	Limited ability to handle sequential or unstructured data	Credit scoring, classification tasks
RNN	Handles sequential data, captures temporal dependencies	Susceptible to vanishing gradient problem	Time-series analysis, stock price prediction
LSTM	Overcomes vanishing gradient problem, handles long-term dependencies	Higher computational cost compared to RNN	Financial time-series analysis, anomaly detection
GRU	Simplified structure, faster training time than LSTM	May not capture long-term dependencies as well as LSTM	Predictive analytics, high-frequency trading
CNN	Effective for pattern recognition, robust to noise	Not designed for sequential data	Fraud detection, anomaly detection, pattern recognition
Transformer	Captures long-range dependencies, efficient parallel processing	Requires large datasets, computationally expensive	Sentiment analysis, market trend prediction
GAN	Generates realistic synthetic data, handles complex data distributions	Training instability, computational complexity	Synthetic financial data generation, stress testing, portfolio optimization
Deep RL	Optimizes dynamic decision making, learns from interaction with environment	Requires significant computational resources and training data	Trading strategy optimization, portfolio management

4. Applications of Deep Learning in Finance

Deep learning has found numerous applications in finance, providing innovative solutions across various aspects of the financial sector. This section explores some of the key applications where deep learning has made a significant impact.

4.1. Algorithmic Trading

Algorithmic trading involves the use of computer algorithms to execute trading orders based on predefined criteria. Deep learning models, specifically those utilizing RNNs like LSTM and GRU, have been increasingly applied to enhance algorithmic trading strategies. These models are capable of analyzing vast amounts of historical and real-time data to identify patterns and predict market movements with greater accuracy. The ability to process and interpret sequential data allows these models to adapt to changing market conditions, offering traders a competitive edge in executing trades efficiently.

Recent studies have demonstrated the effectiveness of deep learning models in algorithmic trading. For instance, Ozbayoglu et al. [48] applied an LSTM-based model to forecast stock prices and subsequently generate trading signals. The proposed LSTM obtained a classification accuracy of 91.5%, which outperformed traditional moving average strategies. Similarly, Wang et al. [14] employed a sequence-to-sequence model, which is an extension of RNNs, to predict market trends and optimize trading algorithms. Their model achieved a prediction accuracy of 85%, leading to more profitable trades. These applications of LSTM and sequence-to-sequence models demonstrate how deep learning can process temporal data and provide dynamic trading strategies in real time.

In addition, Sirignano and Cont [49] proposed a universal deep learning (DL) model using convolutional neural networks (CNNs) to predict price changes in limit order books. Their approach demonstrated an accuracy improvement of 5–10% over traditional methods, significantly enhancing trading strategy performance. Moreover, the model's ability to generalize across different market conditions indicates its robustness and adaptability, making it a valuable tool in volatile trading environments. In real-world applications, banks such as JPMorgan and Goldman Sachs have adopted similar deep learning models to optimize high-frequency trading, where speed and accuracy in prediction provide significant profit margins [50]. The adoption of DL models in these institutions illustrates their capacity to handle massive data streams and react to sudden price movements more efficiently than traditional algorithms.

The use of reinforcement learning (RL) in algorithmic trading has also gained traction. For instance, Huang et al. [51] developed a deep reinforcement learning model that learns optimal trading strategies by interacting with the market environment. Their model outperformed conventional strategies by achieving a higher cumulative return and a precision of 92%, which indicates the potential of RL in creating autonomous trading agents that can continuously adapt to market dynamics without the need for manual intervention. In particular, RL-based models have been successfully implemented by hedge funds like Renaissance Technologies and Citadel, where trading bots autonomously adapt to real-time changes in market conditions and identify profitable opportunities. These real-world applications show how RL is reshaping algorithmic trading by introducing agents that self-learn and adapt to complex market behaviors, minimizing human intervention while maximizing returns.

4.2. Risk Management and Credit Scoring

Risk management and credit scoring are essential components of the financial industry, as accurate assessment of risk is crucial for maintaining financial stability and optimizing decision-making processes. In risk management, institutions must continuously evaluate various types of risk, such as market risk, credit risk, and operational risk, to avoid potential losses. Traditional methods, often based on statistical techniques, tend to oversimplify these risks by assuming linear relationships within the data. However, deep learning models, such as feedforward neural networks and autoencoders, have emerged as powerful tools capable of modeling complex, non-linear relationships within financial datasets. These models can automatically identify intricate patterns in large, high-dimensional data, offering more granular risk assessments compared to conventional models.

In credit scoring, deep learning models have proven effective in analyzing borrower profiles and predicting default risks by capturing subtle interactions between variables that traditional models might overlook. For example, autoencoders have been applied to detect anomalies in transaction data, providing early warnings for potential risks [52]. The unsupervised nature of autoencoders allows these models to identify irregular patterns that do not fit within the expected data distribution, making them particularly useful in high-stakes scenarios where preemptive action is needed. Deep learning's ability to work with both labeled and unlabeled data are especially critical in environments where labeled datasets are scarce, yet there is still a need to make accurate, data-driven decisions. Furthermore, the real-time processing capabilities of DL models make them suitable for high-frequency risk evaluations, providing financial institutions with the tools to adjust their strategies in response to market fluctuations.

For example, Khandani et al. [16] utilized a deep learning model to predict consumer credit risk, achieving a 15% improvement in predictive accuracy over logistic regression models. Their model was effective in identifying high-risk borrowers, with an AUC of 0.89, thereby enabling financial institutions to take preemptive measures to minimize losses. Real-world implementations of deep learning in credit risk assessments have been adopted by institutions like FICO, where deep learning models are increasingly being used to develop more robust credit scores that account for non-traditional variables, such as transactional behavior or social media data, resulting in more inclusive credit evaluation processes [53]. These models not only provide more accurate predictions but also help reduce biases inherent in traditional scoring methods, particularly in underserved markets where traditional credit data might be limited.

Xiao et al. [54] proposed a deep neural network (DNN) for credit scoring that outperformed traditional scoring methods such as FICO scores by up to 20% in terms of predictive accuracy, achieving an AUC of 0.92. The DNN model was able to account for non-linear interactions between variables, providing a more detailed understanding of creditworthiness. This has significant implications for improving access to credit and reducing the likelihood of defaults, especially in underserved markets. Furthermore, companies like Zest AI have applied DL models to make credit decisions that are faster and more accurate while reducing human bias. By integrating deep learning algorithms into the credit scoring process, financial institutions can refine their risk models and make fairer, more inclusive lending decisions. Yang et al. [55] developed a DL model that predicts market risks by analyzing historical data and market indicators. Their model achieved a lower prediction error rate compared to conventional risk assessment models, enabling better-informed decision making regarding asset allocation and risk mitigation. In risk management, leading hedge funds and asset management firms, such as BlackRock, have started leveraging deep learning models to manage portfolio risks in real time. The ability to anticipate market shifts and adjust asset allocations accordingly helps reduce potential losses and optimize long-term returns, highlighting DL's practical application in modern financial systems.

4.3. Fraud Detection

Fraud detection is a critical area where deep learning has been effectively applied to identify and prevent unauthorized transactions, money laundering, and other financial crimes. Traditional rule-based systems and statistical models often struggle to detect sophisticated fraud patterns, particularly when dealing with large volumes of data and rapidly evolving fraud techniques. In contrast, deep learning models, such as CNNs, RNNs, and autoencoders, have proven highly effective in analyzing complex, high-dimensional transaction data and detecting subtle, non-linear patterns indicative of fraudulent activity.

CNN and other deep learning architectures have been utilized to analyze transaction data, detect anomalies, and identify fraudulent patterns in real time [56]. These models can process large volumes of structured, unstructured, and semi-structured data, which is crucial in uncovering intricate patterns that might be missed by traditional methods. For instance, CNNs are particularly robust at detecting spatial patterns within transactional data, while RNNs, such as LSTMs, can capture temporal dependencies, allowing them to track the evolution of fraud over time. Moreover, the real-time processing capabilities of deep learning models enable financial institutions to detect and block fraudulent transactions before they can cause significant harm. Payment processors like PayPal and Visa have adopted deep-learning-based fraud detection systems that analyze millions of transactions per second, reducing false positives and improving overall detection accuracy [57].

Jurgovsky et al. [58] employed LSTM networks to detect fraudulent credit card transactions. Their model achieved an F1-score of 0.93, significantly outperforming traditional machine learning models such as random forests and logistic regression, which had F1-scores of 0.85. The LSTM's ability to capture temporal dependencies in transaction sequences was crucial in identifying suspicious patterns that evolve over time. These real-world applications highlight the growing importance of DL models in adapting to evolving fraud patterns, where new types of fraud can emerge quickly and overwhelm static rule-based systems. By leveraging historical data, DL models can learn from previous fraud cases and anticipate future fraud patterns, helping institutions stay ahead of increasingly sophisticated fraud techniques. Another study by Gandhar et al. [59] used a deep learning model to detect anomalies in financial transactions. The model was able to reduce false positives, which is critical in minimizing the impact of fraud detection on legitimate transactions.

Reducing false positives is important in minimizing customer friction. For example, large financial institutions have adopted anomaly detection models based on deep learning to ensure that genuine transactions are not unnecessarily blocked, improving both customer satisfaction and fraud detection efficiency.

In addition to supervised learning approaches, unsupervised deep learning methods have also been explored for fraud detection. For example, Raj and Kumar [60] developed an unsupervised deep learning model using autoencoders to detect fraudulent patterns in banking transactions without the need for labeled data. Their model achieved a high detection rate of 95% with minimal manual intervention, making it a scalable solution for large financial institutions. Unsupervised models provide a cost-effective solution for institutions with limited access to labeled fraud data, allowing them to detect anomalies based on learned representations of legitimate transactions. Meanwhile, deep reinforcement learning has shown promise in dynamic fraud detection systems. Qayoom et al. [61] proposed a deep Q-learning model for real-time fraud detection that adapts to changing fraud patterns. The model achieved a high detection rate, demonstrating the potential of reinforcement learning in continuously evolving financial environments. Reinforcement learning is useful in environments where fraud patterns shift rapidly, such as in cryptocurrency transactions, where fraud dynamics are less predictable. As digital payments continue to grow, adaptive systems like these are becoming critical for fraud management.

4.4. Market Forecasting

Market forecasting involves predicting future market movements and trends based on historical data. Deep learning models, such as Transformers and attention mechanisms, have shown considerable promise in improving the accuracy of market forecasts. These models can analyze vast datasets, including price movements, economic indicators, and market sentiment, to generate predictions that inform investment strategies. Unlike traditional statistical models, deep learning architectures can capture complex temporal dependencies and non-linear relationships across multiple data sources, providing a more comprehensive view of market dynamics. The ability of deep learning models to capture these intricate relationships within the data allows for more precise and timely forecasts, which are essential for making informed investment decisions in dynamic market environments.

In recent years, Transformers have been increasingly used in market forecasting. For instance, Zeng et al. [62] applied a Transformer-based model to forecast stock prices, achieving a higher prediction than traditional RNN-based models. Transformers have an edge over RNNs because they can handle longer sequences without the risk of vanishing gradients, allowing them to focus on the most relevant past events for improved market predictions. This feature is particularly advantageous in volatile markets where key events significantly impact future trends. For example, in high-frequency trading, Transformers can process large volumes of data in real time, helping traders identify short-term opportunities. The attention mechanism inherent in Transformers allowed the model to focus on the most relevant historical data points, thereby improving the quality of predictions. This approach is useful in financial markets, where certain events or periods may have a disproportionate impact on future prices.

Another approach to market forecasting involves the use of ensemble deep learning methods. Li et al. [63] combined multiple DL models, including CNNs and LSTMs, to forecast market indices. The ensemble approach mitigates the weaknesses of individual models by leveraging their complementary strengths, leading to more robust and reliable market predictions. Ensemble methods are increasingly being adopted by hedge funds and asset management firms, as they provide the flexibility to incorporate various data sources and models, thereby enhancing the accuracy of market predictions [64]. This method is particularly useful for complex market environments where no single model can account for all variables.

Sentiment analysis combined with deep learning has also been explored for market forecasting. Lin et al. [65] developed a hybrid model that integrates sentiment analysis from social media with LSTM networks to predict stock market movements. Their model outperformed traditional sentiment analysis methods, highlighting the value of incorporating unstructured data into financial forecasts. In particular, unstructured data from social media platforms such as Twitter or financial news can provide real-time sentiment analysis, which can be critical for predicting market movements triggered by breaking news or public sentiment. Financial institutions like Bloomberg have been utilizing sentiment analysis combined with deep learning to improve their trading algorithms by gauging market sentiment alongside traditional financial metrics [66].

Additionally, the application of generative models in market forecasting has gained attention. Vuletic et al. [67] used GANs to simulate future market scenarios based on historical data. Their model provided valuable insights into potential market trends, achieving superior performance compared to traditional forecasting models. The use of GANs in market forecasting, though still in its early stages, has shown significant potential for stress-testing various investment strategies under different market conditions. Leading investment firms have begun exploring GAN-based models for market simulations to improve risk management and asset allocation. This helps traders prepare for low-probability, high-impact events such as market crashes or sudden volatility spikes, where traditional models often fail.

4.5. Portfolio Management

Portfolio management requires the optimization of asset allocation to achieve desired financial objectives, such as maximizing returns or minimizing risk. Deep learning models have been applied to develop more sophisticated portfolio management strategies that account for a broader range of variables and market conditions. Unlike traditional optimization techniques, which may rely on static assumptions or limited variables, deep learning approaches are capable of processing vast amounts of data, including historical market behavior, investor preferences, and economic indicators, to deliver more dynamic and adaptive portfolio strategies. Techniques such as reinforcement learning and GANs have been explored to optimize portfolio allocations dynamically, considering factors such as market volatility, investor preferences, and risk tolerance. Ye et al. [68] developed a reinforcement-learning-based model for portfolio management that adapts to changing market conditions by learning from historical data. The reinforcement-learning approach allows the model to continuously adjust the portfolio in response to market changes, thereby providing a more dynamic and responsive investment strategy.

Another application of deep learning in portfolio management involves the use of GANs. Jiang et al. [69] applied a GAN-based model to optimize cryptocurrency portfolios, which are characterized by high volatility and non-linear market behavior. Their model outperformed traditional portfolio optimization methods. The advantage of using GANs in portfolio management is in their ability to generate synthetic market scenarios, providing investors with the ability to simulate and test different strategies under a variety of potential market conditions. This is valuable in volatile asset classes such as cryptocurrencies, where sudden market shifts are common. The use of GANs allows for the generation of synthetic market scenarios, enabling the model to explore a wider range of potential market conditions and optimize the portfolio accordingly.

Deep learning models have also been used to personalize portfolio management strategies. Shi et al. [70] proposed a deep-learning-based framework that tailors investment strategies to individual investor preferences and risk tolerance. Their model integrates reinforcement learning with DL to optimize asset allocation in real time. This approach demonstrates the potential of DL in creating more personalized and effective investment solutions.

Furthermore, DL models have been employed to enhance portfolio diversification strategies. Zhang et al. [71] utilized a deep learning model to analyze the correlation structure of assets in a portfolio, improving diversification by identifying non-obvious correlations that traditional methods might miss. By identifying previously overlooked correlations among assets, deep learning models can help portfolio managers reduce overall portfolio risk while maximizing returns. Additionally, the integration of deep learning with traditional quantitative models has been explored to enhance portfolio management. Lin et al. [72] proposed a hybrid approach that combines deep learning with factor models to optimize asset allocation. This hybrid approach employs the strengths of both deep learning and traditional financial theories, providing a balanced and effective portfolio management strategy. The hybridization of deep learning with traditional quantitative models offers the advantage of combining the adaptability and precision of DL models with the stability and theoretical rigor of traditional financial frameworks, making it a promising approach for institutions aiming to enhance the accuracy of their asset allocation strategies.

4.6. Customer Segmentation

Customer segmentation is a critical task in the financial industry, enabling personalized marketing, targeted product offerings, and improved customer service. It involves dividing a broad customer base into smaller, more manageable groups based on shared characteristics. These characteristics can include behavioral patterns, financial habits, and transaction histories. Customer segmentation has traditionally relied on clustering techniques like k-means [73], hierarchical clustering [74], and Gaussian mixture models [75], which are effective but often limited by their reliance on predefined features and linear relationships. However, the application of deep learning techniques has provided a more sophisticated approach to customer segmentation by automatically learning complex, non-linear patterns and latent features within the data. This allows for more granular segmentation that can reveal previously hidden relationships between customers, enabling more accurate predictions of customer behavior.

Wang [76] proposed a novel unsupervised deep learning approach for customer segmentation. A Modified Social Spider Optimization algorithm is employed for feature selection, identifying relevant customer behaviors. These selected features are then used to cluster customers using a Self-Organizing Neural Network. Finally, a DNN classifies customers based on these clusters. The proposed model achieves high segmentation accuracy (98.67%), outperforming traditional methods. The integration of unsupervised learning for feature selection and clustering, followed by supervised classification, demonstrates the versatility of deep learning in customer segmentation. Financial institutions have employed deep-learning-based segmentation models to personalize banking services and improve customer retention, illustrating how AI-driven segmentation can have tangible benefits in practice.

Mousaeirad [77] proposed a novel customer segmentation approach using a neural embedding framework called Customer2Vec. The approach leverages feature engineering to identify important customer characteristics and combines supervised and unsupervised learning techniques to embed customers into a vector space. This allows for a better understanding of customer similarities and improves the quality of segmentation, as demonstrated in a banking sector case study. The use of embeddings for customer segmentation introduces a new level of precision, enabling institutions to track changing customer behavior in real time.

4.7. Financial Document Analysis and Information Extraction

Financial document analysis and information extraction are vital components in the financial industry, facilitating tasks such as risk assessment, compliance monitoring, and decision making. The process involves automatically identifying, extracting, and interpreting relevant data from vast amounts of unstructured financial documents, including invoices, contracts, reports, and transaction records. Traditionally, rule-based systems and natural language processing (NLP) techniques, such as named entity recognition (NER) [78], have been employed for these tasks. However, these methods often struggle with the complexity and variability of financial documents, especially when dealing with ambiguous language, varying formats, and domain-specific terminologies.

Recent advancements in machine learning, particularly in deep learning, have significantly enhanced the capabilities of financial document analysis. Models, such as Bidirectional Encoder Representations from Transformers (BERT) and its domain-specific variants such as FinBERT, have revolutionized the extraction of information from financial texts by capturing contextual meanings and domain-specific nuances. These models excel in tasks such as sentiment analysis, entity recognition, and document classification, thereby improving the accuracy and efficiency of information extraction in financial contexts. FinBERT, in particular, has been adopted by financial firms such as Bloomberg and Thomson Reuters for automating financial news and report analysis [79], demonstrating the practical utility of domain-specific models in real-world settings. These institutions use FinBERT to automate tasks like market sentiment analysis, which significantly reduces the time required for analysts to process large volumes of unstructured data.

Melus [80] introduced an automated framework that combines a fine-tuned BERT with Optical Character Recognition (OCR) for analyzing scanned financial documents. The OCR component converts scanned images into machine-readable text, which is then processed by the BERT model to extract relevant information. This approach is particularly useful in scenarios where documents are not originally in digital format, ensuring that even legacy paper documents can be included in automated workflows. The integration of OCR with deep learning models ensures financial institutions can automate their compliance and reporting processes, saving time and reducing human error. In addition, Yang et al. [81] developed FinBERT, a pre-trained language model specifically tailored for financial sentiment analysis. By fine-tuning BERT on a large corpus of financial texts, FinBERT outperforms general-purpose models in tasks such as sentiment classification and named entity recognition within financial documents, making it a powerful tool for tasks like market sentiment analysis and risk assessment.

Another significant contribution by Montariol et al. [82] proposed a multi-task learning framework using BERT for financial annual report feature extraction. This model simultaneously performs multiple related tasks, such as sentiment detection, objectivity extraction, and Environmental, Social, and Governance classification, by leveraging the shared representations learned by BERT. The multi-task approach not only improves the performance of individual tasks but also enhances the model's ability to generalize across different types of feature extraction in financial reports.

Similarly, Moirangthem and Lee [83] explored the use of GRUs for financial text classification, utilizing a hierarchical structure to better capture the document's contextual information. The GRUs in their model effectively processed sequential data, enabling the extraction of meaningful sentence-level and document-level representations. Through incorporating a hierarchical attention mechanism, the model could assign varying levels of importance to different sentences and words, leading to improved accuracy in classifying financial texts. This approach demonstrated the effectiveness of GRUs in handling complex financial documents, such as earnings reports and financial news articles, by emphasizing the most relevant content within the text.

5. Recent Advances and Emerging Trends

The field of deep learning in finance is rapidly evolving, driven by both technological advancements and the growing availability of data. This section explores recent developments and emerging trends that are shaping the future of this domain.

1. Explainable AI and Model Transparency: One of the most significant recent advances in the field has been the development of Explainable AI (XAI) techniques. These methods aim to make the decision-making processes of DL models more transparent and understandable to human users. This is important in finance, where stakeholders need to trust and comprehend the reasoning behind model predictions, especially in high-stakes environments such as credit scoring, fraud detection, and trading.

Techniques such as SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) are increasingly being adopted to enhance model transparency [84].

2. Transfer Learning and Pretrained Models: Transfer learning is a powerful technique in deep learning, allowing models trained on one task to be repurposed for another related task [85]. This approach is beneficial in financial applications, where labeled data are often scarce or expensive to obtain. By leveraging pre-trained models on large datasets (e.g., language models for sentiment analysis), financial institutions can achieve high performance with limited data. This trend has also led to the development of financial-specific pre-trained models, which can be fine-tuned for specific applications such as market prediction or risk assessment [86].
3. Federated Learning and Data Privacy: With increasing concerns about data privacy, federated learning has gained traction as a solution that allows for collaborative model training without the need to share raw data across institutions. In federated learning, models are trained across decentralized devices or servers, where data remain local, and only the model updates are shared. This approach is advantageous in finance since data privacy is paramount, enabling institutions to benefit from collective learning while maintaining data security and compliance with regulations like GDPR [87].
4. Reinforcement Learning in Financial Markets: Reinforcement learning has seen a surge of interest as a method for optimizing decision-making processes in financial markets. Unlike supervised learning, where models learn from labeled data, RL involves learning from the environment through trial and error, making it highly suitable for dynamic environments like trading. RL models are being used to develop autonomous trading agents, optimize portfolio management strategies, and improve algorithmic trading systems by adapting to changing market conditions [88].
5. Quantum Computing and Quantum Machine Learning: Quantum computing, though still in its infancy, is an emerging trend that holds the potential to revolutionize DL and financial modeling. Quantum ML leverages quantum computer's ability to process information at speeds far beyond classical computers, offering the promise of solving complex optimization problems in finance more efficiently. While practical applications are still limited, ongoing research and development in quantum algorithms for financial modeling suggest that this technology could become a significant technology in the future of finance [89].
6. Ethical AI and Fairness: As AI technologies become more embedded in financial systems, there is a growing emphasis on ensuring that these systems operate fairly and ethically. Recent advances have focused on developing methods to detect and mitigate biases in AI models, ensuring that financial services are accessible and equitable. This trend is driving the adoption of fairness-aware machine learning techniques and the integration of ethical considerations into the AI development lifecycle. The financial industry is increasingly prioritizing these concerns to maintain public trust and comply with evolving regulatory standards [90].

6. Challenges and Limitations

Applying deep learning in the financial sector presents several notable challenges that can hinder effectiveness and practical deployment, and they are discussed in this section.

6.1. Data Quality and Availability

One of the fundamental challenges in applying deep learning to financial data are the quality and availability of the data itself. Financial datasets often contain a high degree of noise and are subject to issues such as missing values, outliers, and inconsistencies. Moreover, the sensitive nature of financial information means that many data are not publicly available, and where they are available, they often come with stringent usage restrictions [67]. These factors can significantly impede the training and validation of robust DL models, which require large, diverse, and representative datasets to function optimally.

Another issue is the dynamic nature of financial markets. Financial data are often non-stationary, meaning that the underlying distribution can change over time. This presents a challenge for deep learning models, which tend to perform best when trained on stationary data [91]. Addressing these issues often requires sophisticated data preprocessing techniques, such as resampling, normalization, and dealing with missing values, but even these solutions are not always sufficient to guarantee data quality. Furthermore, the availability of high-frequency data are limited, which can constrain model development, especially for more complex tasks, like real-time trading or high-frequency fraud detection.

6.2. Overfitting and Model Interpretability

Deep learning models, mainly those with many layers and parameters, are prone to overfitting, especially when trained on financial data that inherently exhibits high volatility and is non-stationary. Overfitting leads to models that perform well on training data but fail to generalize to unseen data. This is concerning in financial markets, where past performance is not always indicative of future outcomes, making overfitting a significant risk when designing predictive models for stock prices, credit risk, or portfolio management.

Additionally, the "black-box" nature of many DL models poses significant challenges in terms of interpretability [84,92]. Financial stakeholders typically require clear explanations for decisions made by automated systems, particularly in scenarios involving investments, lending, and risk management, where accountability is crucial. Explainability is especially important when regulatory requirements demand transparency, such as in credit scoring decisions. In response, research into XAI techniques has gained momentum, with methods like SHAP and LIME being employed to make DL models more interpretable. However, the complexity of these models still limits their deployment in high-stakes financial applications.

6.3. Computational Complexity

The training of DL models often requires substantial computational resources, which can be a barrier, especially for smaller institutions or startups. Deep learning algorithms, particularly those involving large neural networks with many layers, demand considerable processing power, memory, and storage capacity, which may not be feasible for all financial firms [93]. This complexity also extends to the time required to train these models, which can lead to long development cycles and increased costs. Additionally, many DL applications require continuous retraining to remain effective in dynamic financial environments, further increasing the computational demands.

Furthermore, the energy consumption associated with training and maintaining these models can be considerable, adding to operational costs and environmental impact. Recent research has highlighted the significant carbon footprint associated with training large-scale deep learning models. This has led to a growing interest in developing more energy-efficient algorithms and hardware, such as Tensor Processing Units (TPUs) and more optimized DL architectures that can perform well with less computational overhead. However, balancing the need for computational efficiency with model performance remains a critical challenge in the deployment of DL in finance.

6.4. Ethical and Regulatory Concerns

Deep learning applications in finance must navigate a complex landscape of ethical and regulatory issues, particularly when it comes to ensuring transparency and accountability. One of the primary challenges is the "black-box" nature of many deep learning models, which can obscure the decision-making process for both the developers and users of the system [90]. This lack of transparency is especially problematic in the financial industry, where decisions can directly affect individuals' financial well-being. Automated credit scoring, loan approvals, and investment decisions made by opaque algorithms can have serious consequences, including legal challenges if a model's reasoning cannot be adequately explained or defended.

Regulatory frameworks, such as the General Data Protection Regulation (GDPR) in Europe, have introduced specific requirements related to the use of AI in decision-making processes, particularly with respect to data privacy and algorithmic transparency. Under the GDPR, financial institutions using AI systems must be able to explain the logic behind automated decisions and provide mechanisms for users to contest outcomes. Furthermore, these regulations require that any personal data used in AI models is handled in a secure and privacy-compliant manner [94]. Financial institutions need to incorporate XAI tools that provide post hoc explanations of model decisions to meet these regulatory demands, but this remains an ongoing challenge as models become more complex and difficult to interpret.

Additionally, ethical concerns are increasingly coming to the forefront, particularly as the use of DL models in finance continues to expand. There is a growing risk that automated decision-making systems could perpetuate or even exacerbate existing societal inequalities, especially when they are trained on biased or incomplete datasets. Financial services such as lending and insurance can inadvertently reinforce biases if not carefully designed and monitored. As a result, financial institutions and regulators are increasingly focused on creating ethical frameworks that address these concerns, ensuring that AI systems are used responsibly and equitably. This includes the formation of AI ethics boards and guidelines that promote fairness, transparency, and accountability in all AI-driven financial decisions.

6.5. Bias and Fairness

Bias and fairness in ML models are critical challenges, especially in finance, where algorithmic decisions can have profound societal impacts. The primary source of bias in financial AI systems often stems from historical data, which may reflect systemic inequalities. When DL models are trained on these biased datasets, they can perpetuate or even amplify these inequalities, leading to unfair outcomes such as biased credit scoring, loan approvals, or investment advice. For example, if a model's training data are skewed toward certain demographic groups, it may make decisions that disproportionately favor or disadvantage specific populations [95]. This not only affects the fairness and accuracy of financial services but can also violate legal frameworks designed to protect against discrimination.

To address these issues, fairness-aware algorithms are being developed that aim to detect and mitigate bias during the training process. These algorithms seek to enforce fairness metrics, such as demographic parity or equalized odds, ensuring that the model does not unfairly penalize or benefit any particular group. In addition to these algorithmic solutions, XAI techniques, such as SHAP and LIME, are being employed to provide transparency into how deep learning models make decisions. These tools allow stakeholders to understand the relative importance of various features (such as income or geographic location) in the decision-making process, which helps in identifying and mitigating potential biases before they become systemic issues.

However, achieving true fairness in DL models remains an ongoing challenge. Bias in financial AI systems is not only a technical issue but also a societal and ethical concern. Financial institutions must balance accuracy with fairness while adhering to legal frameworks such as the Equal Credit Opportunity Act (ECOA) in the U.S., which prohibits discrimination based on race, gender, or other protected characteristics in lending practices [96]. As models evolve and become more complex, there is a growing recognition that fairness cannot be fully achieved through technical fixes alone. Continuous monitoring, auditing, and collaboration between AI developers, financial institutions, and regulators are essential to ensuring that AI-driven financial systems remain fair and equitable, and that they promote inclusivity in financial services.

7. Future Research Directions

The ongoing evolution of deep learning in the financial industry presents numerous avenues for future research. One critical area is enhancing the interpretability and explainability of DL models. As these models grow in complexity, there is a pressing need to develop new techniques that can provide transparent and actionable insights, especially in financial contexts where understanding the rationale behind model predictions is crucial. While XAI has made significant strides, further research could focus on integrating domain-specific knowledge and creating more intuitive visualization tools that bridge the gap between technical complexity and practical usability, a challenge that remains largely unmet.

In addition to interpretability, improving data quality and addressing data scarcity continue to be significant challenges. Financial data often suffer from issues such as noise, missing values, and non-stationarity, which can undermine the reliability of DL models. Research into more sophisticated data preprocessing methods and techniques for augmenting scarce datasets, such as the generation of synthetic data using GANs, could greatly enhance the robustness of financial models. Furthermore, exploring federated learning as a solution to data scarcity and privacy concerns offers a promising research direction, especially in contexts where data sharing is restricted by regulatory or competitive concerns.

Addressing bias and ensuring fairness in AI systems is another critical area that requires further investigation. Bias in financial models can lead to unfair and discriminatory outcomes, which is concerning in areas such as credit scoring and lending. While some progress has been made in developing fairness-aware algorithms, there remains a substantial gap in research on how to systematically detect and mitigate bias throughout the model lifecycle, especially in the dynamic environments typical of financial markets. Future research could focus on creating comprehensive frameworks that monitor and correct biases as models are deployed in real-world scenarios, which is an essential step toward ensuring that financial AI systems operate ethically.

Reinforcement learning in financial applications, though promising, is still in its early stages and presents numerous opportunities for further research. While RL has shown potential in optimizing trading strategies and portfolio management, more work is needed to refine these approaches in order to develop risk-sensitive RL models and hybrid systems that combine RL with traditional financial techniques. This research could lead to more robust and adaptable financial decision-making systems.

8. Conclusions

Deep learning algorithms have become integral to numerous financial applications, including credit scoring, fraud detection, algorithmic trading, and market forecasting. This paper provides a concise yet comprehensive overview of key DL models, such as CNNs, LSTMs, GANs, and Deep RL. The study also addresses critical challenges associated with deploying these models in finance, including data quality, model interpretability, and computational demands. This paper is essential for researchers and practitioners looking to understand the current field of deep learning in finance and the potential future directions in this rapidly evolving field.

Author Contributions: Conceptualization, E.M., N.J., G.O., I.D.M. and K.A.; methodology, E.M., N.J., G.O., I.D.M. and K.A.; validation, I.D.M. and G.O.; investigation, E.M. and I.D.M.; writing—original draft preparation, E.M., N.J., G.O., I.D.M. and K.A.; writing—review and editing, E.M., N.J., G.O., I.D.M. and K.A.; visualization, I.D.M.; supervision, N.J. and I.D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AdaBoost	Adaptive Boosting
CNN	Convolutional Neural Network
DNN	Deep Neural Network
DL	Deep Learning
FNN	Feedforward Neural Network
GAN	Generative Adversarial Network
GDPR	General Data Protection Regulation
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
ML	Machine Learning
NLP	Natural Language Processing
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
RL	Reinforcement Learning
Tanh	Hyperbolic Tangent
XAI	Explainable Artificial Intelligence
XGBoost	Extreme Gradient Boosting

References

1. Talaat, F.M.; Aljadani, A.; Badawy, M.; Elhosseini, M. Toward interpretable credit scoring: Integrating explainable artificial intelligence with deep learning for credit card default prediction. *Neural Comput. Appl.* **2024**, *36*, 4847–4865. [CrossRef]
2. Huang, Y.; Wan, X.; Zhang, L.; Lu, X. A novel deep reinforcement learning framework with BiLSTM-Attention networks for algorithmic trading. *Expert Syst. Appl.* **2024**, *240*, 122581. [CrossRef]
3. Vaca, C.; Astorgano, M.; López-Rivero, A.J.; Tejerina, F.; Sahelices, B. Interpretability of deep learning models in analysis of Spanish financial text. *Neural Comput. Appl.* **2024**, *36*, 7509–7527. [CrossRef]
4. Mienye, I.D.; Jere, N. Deep Learning for Credit Card Fraud Detection: A Review of Algorithms, Challenges, and Solutions. *IEEE Access* **2024**, *12*, 96893–96910. [CrossRef]
5. Ahmed, S.; Alshater, M.M.; El Ammari, A.; Hammami, H. Artificial intelligence and machine learning in finance: A bibliometric review. *Res. Int. Bus. Financ.* **2022**, *61*, 101646. [CrossRef]
6. Goodell, J.W.; Kumar, S.; Lim, W.M.; Pattnaik, D. Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis. *J. Behav. Exp. Financ.* **2021**, *32*, 100577. [CrossRef]
7. Gunnarsson, B.R.; Vanden Broucke, S.; Baesens, B.; Óskarsdóttir, M.; Lemahieu, W. Deep learning for credit scoring: Do or don't? *Eur. J. Oper. Res.* **2021**, *295*, 292–305. [CrossRef]
8. Sonkavde, G.; Dharrao, D.S.; Bongale, A.M.; Deokate, S.T.; Doreswamy, D.; Bhat, S.K. Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis and discussion of implications. *Int. J. Financ. Stud.* **2023**, *11*, 94. [CrossRef]
9. Pricope, T.V. Deep reinforcement learning in quantitative algorithmic trading: A review. *arXiv* **2021**, arXiv:2106.00123.
10. Bengesi, S.; El-Sayed, H.; Sarker, M.K.; Houkpati, Y.; Irungu, J.; Oladunni, T. Advancements in Generative AI: A Comprehensive Review of GANs, GPT, Autoencoders, Diffusion Model, and Transformers. *IEEE Access* **2024**, *12*, 69812–69837. [CrossRef]
11. Khan, W.; Daud, A.; Khan, K.; Muhammad, S.; Haq, R. Exploring the frontiers of deep learning and natural language processing: A comprehensive overview of key challenges and emerging trends. *Nat. Lang. Process. J.* **2023**, *4*, 100026. [CrossRef]
12. Assefa, S.A.; Dervovic, D.; Mahfouz, M.; Tillman, R.E.; Reddy, P.; Veloso, M. Generating synthetic data in finance: Opportunities, challenges and pitfalls. In Proceedings of the First ACM International Conference on AI in Finance, New York, NY, USA, 15–16 October 2020; pp. 1–8.
13. Soleymani, F.; Paquet, E. Financial portfolio optimization with online deep reinforcement learning and restricted stacked autoencoder—DeepBreath. *Expert Syst. Appl.* **2020**, *156*, 113456. [CrossRef]
14. Wang, J.; Sun, T.; Liu, B.; Cao, Y.; Zhu, H. CLVSA: A convolutional LSTM based variational sequence-to-sequence model with attention for predicting trends of financial markets. *arXiv* **2021**, arXiv:2104.04041.
15. Meng, B.; Sun, J.; Shi, B. A novel URP-CNN model for bond credit risk evaluation of Chinese listed companies. *Expert Syst. Appl.* **2024**, *255*, 124861. [CrossRef]
16. Khandani, A.E.; Kim, A.J.; Lo, A.W. Consumer credit-risk models via machine-learning algorithms. *J. Bank. Financ.* **2010**, *34*, 2767–2787. [CrossRef]

17. Esenogho, E.; Mienye, I.D.; Swart, T.G.; Aruleba, K.; Obaido, G. A neural network ensemble with feature engineering for improved credit card fraud detection. *IEEE Access* **2022**, *10*, 16400–16407. [CrossRef]
18. Mienye, I.D.; Sun, Y. A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection. *IEEE Access* **2023**, *11*, 30628–30638. [CrossRef]
19. Sezer, O.B.; Ozbayoglu, M. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Appl. Soft Comput.* **2020**, *90*, 106181. [CrossRef]
20. Lim, B.; Zohren, S. Time-series forecasting with deep learning: A survey. *Philos. Trans. R. Soc.* **2021**, *379*, 20200209. [CrossRef]
21. Ojha, V.K.; Abraham, A.; Snášel, V. Metaheuristic design of feedforward neural networks: A review of two decades of research. *Eng. Appl. Artif. Intell.* **2017**, *60*, 97–116. [CrossRef]
22. Alemu, H.Z.; Wu, W.; Zhao, J. Feedforward neural networks with a hidden layer regularization method. *Symmetry* **2018**, *10*, 525. [CrossRef]
23. Nwankpa, C.; Ijomah, W.; Gachagan, A.; Marshall, S. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv* **2018**, arXiv:1811.03378.
24. Sherstinsky, A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys. Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
25. Kombrink, S.; Mikolov, T.; Karafiát, M.; Burget, L. Recurrent neural network based language model. *Interspeech* **2010**, *2*, 1045–1048.
26. Mienye, I.D.; Swart, T.G.; Obaido, G. Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. *Information* **2024**, *15*, 517. [CrossRef]
27. Salehinejad, H.; Sankar, S.; Barfett, J.; Colak, E.; Valaee, S. Recent advances in recurrent neural networks. *arXiv* **2017**, arXiv:1801.01078.
28. Oruh, J.; Viriri, S.; Adegun, A. Long short-term memory recurrent neural network for automatic speech recognition. *IEEE Access* **2022**, *10*, 30069–30079. [CrossRef]
29. Oliveira, P.; Fernandes, B.; Analide, C.; Novais, P. Forecasting energy consumption of wastewater treatment plants with a transfer learning approach for sustainable cities. *Electronics* **2021**, *10*, 1149. [CrossRef]
30. Dutta, A.; Kumar, S.; Basu, M. A gated recurrent unit approach to bitcoin price prediction. *J. Risk Financ. Manag.* **2020**, *13*, 23. [CrossRef]
31. Obaido, G.; Mienye, I.D.; Egbelowo, O.F.; Emmanuel, I.D.; Ogunleye, A.; Ogbuokiri, B.; Mienye, P.; Aruleba, K. Supervised machine learning in drug discovery and development: Algorithms, applications, challenges, and prospects. *Mach. Learn. Appl.* **2024**, *17*, 100576. [CrossRef]
32. González-Rodríguez, L.; Plasencia-Salgueiro, A. Uncertainty-Aware autonomous mobile robot navigation with deep reinforcement learning. In *Deep Learning for Unmanned Systems*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 225–257.
33. Abou Houran, M.; Bukhari, S.M.S.; Zafar, M.H.; Mansoor, M.; Chen, W. COA-CNN-LSTM: Coati optimization algorithm-based hybrid deep learning model for PV/wind power forecasting in smart grid applications. *Appl. Energy* **2023**, *349*, 121638. [CrossRef]
34. Zhu, B.; Yang, W.; Wang, H.; Yuan, Y. A hybrid deep learning model for consumer credit scoring. In Proceedings of the 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 26–28 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 205–208.
35. Khan, S.; Naseer, M.; Hayat, M.; Zamir, S.W.; Khan, F.S.; Shah, M. Transformers in vision: A survey. *Acm Comput. Surv. (Csur)* **2022**, *54*, 1–41. [CrossRef]
36. Zhang, Q.; Qin, C.; Zhang, Y.; Bao, F.; Zhang, C.; Liu, P. Transformer-based attention network for stock movement prediction. *Expert Syst. Appl.* **2022**, *202*, 117239.
37. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. Acm* **2020**, *63*, 139–144. [CrossRef]
38. Ramzan, F.; Sartori, C.; Consoli, S.; Reforgiato Recupero, D. Generative Adversarial Networks for Synthetic Data Generation in Finance: Evaluating Statistical Similarities and Quality Assessment. *AI* **2024**, *5*, 667–685. [CrossRef]
39. Pun, C.S.; Wang, L.; Wong, H.Y. Financial thought experiment: A GAN-based approach to vast robust portfolio selection. In Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI'20), Yokohama, Japan, 7–15 January 2020.
40. Nguyen, T.T.; Nguyen, N.D.; Nahavandi, S. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Trans. Cybern.* **2020**, *50*, 3826–3839. [CrossRef]
41. Wang, Z.; Huang, B.; Tu, S.; Zhang, K.; Xu, L. DeepTrader: A deep reinforcement learning approach for risk-return balanced portfolio management with market conditions Embedding. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, Canada, 2–9 February 2021; Volume 35, pp. 643–650.
42. Salakhutdinov, R. Learning deep generative models. *Annu. Rev. Stat. Its Appl.* **2015**, *2*, 361–385. [CrossRef]
43. Yu, D.; Deng, L.; Dahl, G. Roles of pre-training and fine-tuning in context-dependent DBN-HMMs for real-world speech recognition. In Proceedings of the Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning. sn, San Diego, CA, USA, 15 December 2010.
44. Hua, Y.; Guo, J.; Zhao, H. Deep belief networks and deep learning. In Proceedings of the 2015 International Conference on Intelligent Computing and Internet of Things, Harbin, China, 17–18 January 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–4.

45. Ahmed, S.F.; Alam, M.S.B.; Hassan, M.; Rozbu, M.R.; Ishtiak, T.; Rafa, N.; Mofijur, M.; Shawkat Ali, A.; Gandomi, A.H. Deep learning modelling techniques: Current progress, applications, advantages, and challenges. *Artif. Intell. Rev.* **2023**, *56*, 13521–13617. [CrossRef]
46. Mienye, I.D.; Swart, T.G. A Hybrid Deep Learning Approach with Generative Adversarial Network for Credit Card Fraud Detection. *Technologies* **2024**, *12*, 186. [CrossRef]
47. Salamai, A.A. Curriculum learning empowered reinforcement learning for graph-based portfolio management: Performance optimization and comprehensive analysis. *Neural Netw.* **2024**, *179*, 106537. [CrossRef]
48. Ozbayoglu, A.M.; Gudelek, M.U.; Sezer, O.B. Deep learning for financial applications: A survey. *Appl. Soft Comput.* **2020**, *93*, 106384. [CrossRef]
49. Sirignano, J.; Cont, R. Universal features of price formation in financial markets: Perspectives from deep learning. *Quant. Financ.* **2019**, *19*, 1449–1459. [CrossRef]
50. Balaji, K. Harnessing AI for Financial Innovations: Pioneering the Future of Financial Services. In *Modern Management Science Practices in the Age of AI*; IGI Global: Hershey, PA, USA, 2024; pp. 91–122.
51. Huang, Y.; Zhou, X. Financial trading as a game: A deep reinforcement learning approach. *J. Financ. Data Sci.* **2018**, *1*, 10–24.
52. Chang, C.H. Managing credit card fraud risks by autoencoders. In *Advances in Pacific Basin Business, Economics and Finance*; Emerald Publishing Limited: Bingley, UK, 2021; Vol. 9, pp. 225–235.
53. Gambacorta, L.; Huang, Y.; Qiu, H.; Wang, J. How do machine learning and non-traditional data affect credit scoring? New evidence from a Chinese fintech firm. *J. Financ. Stab.* **2024**, *73*, 101284. [CrossRef]
54. Xiao, J.; Zhong, Y.; Jia, Y.; Wang, Y.; Li, R.; Jiang, X.; Wang, S. A novel deep ensemble model for imbalanced credit scoring in internet finance. *Int. J. Forecast.* **2024**, *40*, 348–372. [CrossRef]
55. Yang, T.; Li, A.; Xu, J.; Su, G.; Wang, J. Deep learning model-driven financial risk prediction and analysis. *Appl. Comput. Eng.* **2024**, *67*, 54–60. [CrossRef]
56. Mienye, I.D.; Sun, Y. A machine learning method with hybrid feature selection for improved credit card fraud detection. *Appl. Sci.* **2023**, *13*, 7254. [CrossRef]
57. Abi Din, Z.; Venugopalan, H.; Lin, H.; Wushensky, A.; Liu, S.; King, S.T. Doing good by fighting fraud: Ethical anti-fraud systems for mobile payments. In Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP), Francisco, CA, USA, 24–27 May 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1623–1640.
58. Jurgovsky, J.; Granitzer, M.; Ziegler, K.; Calabretto, S.; Portier, P.E.; He-Guelton, L.; Caelen, O. Sequence classification for credit-card fraud detection. *Expert Syst. Appl.* **2018**, *100*, 234–245. [CrossRef]
59. Gandhar, A.; Gupta, K.; Pandey, A.K.; Raj, D. Fraud Detection Using Machine Learning and Deep Learning. *Comput. Sci.* **2024**, *5*, 1–10. [CrossRef]
60. Rai, A.K.; Dwivedi, R.K. Fraud detection in credit card data using unsupervised machine learning based scheme. In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2–4 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 421–426.
61. Qayoom, A.; Khuhro, M.A.; Kumar, K.; Waqas, M.; Saeed, U.; ur Rehman, S.; Wu, Y.; Wang, S. A novel approach for credit card fraud transaction detection using deep reinforcement learning scheme. *Peerj Comput. Sci.* **2024**, *10*, e1998. [CrossRef]
62. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are transformers effective for time series forecasting? In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 11121–11128. [CrossRef]
63. Li, Y.; Pan, Y. A novel ensemble deep learning model for stock prediction based on stock prices and news. *Int. J. Data Sci. Anal.* **2022**, *13*, 139–149. [CrossRef] [PubMed]
64. Mienye, I.D.; Jere, N. A Survey of Decision Trees: Concepts, Algorithms, and Applications. *IEEE Access* **2024**, *12*, 86716–86727. [CrossRef]
65. Lin, Y.L.; Lai, C.J.; Pai, P.F. Using deep learning techniques in forecasting stock markets by hybrid data with multilingual sentiment analysis. *Electronics* **2022**, *11*, 3513. [CrossRef]
66. Du, K.; Xing, F.; Mao, R.; Cambria, E. Financial sentiment analysis: Techniques and applications. *Acm Comput. Surv.* **2024**, *56*, 1–42. [CrossRef]
67. Vuletić, M.; Prenzel, F.; Cucuringu, M. Fin-gan: Forecasting and classifying financial time series via generative adversarial networks. *Quant. Financ.* **2024**, *24*, 175–199. [CrossRef]
68. Ye, Y.; Pei, H.; Wang, B.; Chen, P.Y.; Zhu, Y.; Xiao, J.; Li, B. Reinforcement-learning based portfolio management with augmented asset movement prediction states. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 19–23 October 2020; Volume 34, pp. 1112–1119.
69. Jiang, Z.; Liang, J. Cryptocurrency portfolio management with deep reinforcement learning. In Proceedings of the IEEE International Conference on Data Mining Workshops, Orleans, LA, USA, 18–21 November 2017; pp. 18–25.
70. Shi, S.; Li, J.; Li, G.; Pan, P.; Liu, K. Xpm: An explainable deep reinforcement learning framework for portfolio management. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Queensland, Australia, 1–5 November 2021; pp. 1661–1670.
71. Zhang, Z.; Zohren, S.; Roberts, S. Deep learning for portfolio optimization. *arXiv* **2020**, arXiv:2005.13665. [CrossRef]
72. Lin, Y.C.; Chen, C.T.; Sang, C.Y.; Huang, S.H. Multiagent-based deep reinforcement learning for risk-shifting portfolio management. *Appl. Soft Comput.* **2022**, *123*, 108894. [CrossRef]

73. Kansal, T.; Bahuguna, S.; Singh, V.; Choudhury, T. Customer segmentation using K-means clustering. In Proceedings of the 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Belagavi, India, 21–22 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 135–139.
74. Abdulhafedh, A. Incorporating k-means, hierarchical clustering and pca in customer segmentation. *J. City Dev.* **2021**, *3*, 12–30.
75. Chang, W.; Kim, S.; Chae, H. A regularized spatial market segmentation method with Dirichlet process—Gaussian mixture prior. *Spat. Stat.* **2020**, *35*, 100402. [CrossRef]
76. Wang, C. Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach. *Inf. Process. Manag.* **2022**, *59*, 103085. [CrossRef]
77. Mousaeirad, S. Intelligent vector-based customer segmentation in the banking industry. *arXiv* **2020**, arXiv:2012.11876.
78. Gunter, D.; Puac-Polanco, P.; Miguel, O.; Thornhill, R.E.; Yu, A.Y.; Liu, Z.A.; Mamdani, M.; Pou-Prom, C.; Aviv, R.I. Rule-based natural language processing for automation of stroke data extraction: A validation study. *Neuroradiology* **2022**, *64*, 2357–2362. [CrossRef] [PubMed]
79. Sinha, A.; Kedas, S.; Kumar, R.; Malo, P. SEntFiN 1.0: Entity-aware sentiment analysis for financial news. *J. Assoc. Inf. Sci. Technol.* **2022**, *73*, 1314–1335. [CrossRef]
80. Melus, D. Enhancing Quality of Optical Character Recognition for Financial Document Processing. Master's Thesis, Masaryk University, Brno, Czech, 2023.
81. Yang, Y.; Uy, M.C.S.; Huang, A. Finbert: A pretrained language model for financial communications. *arXiv* **2020**, arXiv:2006.08097.
82. Montariol, S.; Martinc, M.; Pelicon, A.; Pollak, S.; Koloski, B.; Lončarski, I.; Valentinčič, A.; Šuštar, K.S.; Ichev, R.; Žnidaršič, M. Multi-task Learning for Features Extraction in Financial Annual Reports. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Vilnius, Lithuania, 8–12 September 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 7–24.
83. Moirangthem, D.S.; Lee, M. Hierarchical and lateral multiple timescales gated recurrent units with pre-trained encoder for long text classification. *Expert Syst. Appl.* **2021**, *165*, 113898. [CrossRef]
84. Mienye, I.D.; Jere, N. Optimized ensemble learning approach with explainable AI for improved heart disease prediction. *Information* **2024**, *15*, 394. [CrossRef]
85. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
86. Zhang, Y.; Zhang, H. FinBERT–MRC: Financial named entity recognition using BERT under the machine reading comprehension paradigm. *Neural Process. Lett.* **2023**, *55*, 7393–7413. [CrossRef]
87. Yang, Q.; Liu, Y.; Chen, T.; Tong, Y. Federated machine learning: Concept and applications. *Acm Trans. Intell. Syst. Technol. (Tist)* **2019**, *10*, 1–19. [CrossRef]
88. Lussange, J.; Vrizzi, S.; Palminteri, S.; Gutkin, B. Mesoscale effects of trader learning behaviors in financial markets: A multi-agent reinforcement learning study. *PloS One* **2024**, *19*, e0301141. [CrossRef] [PubMed]
89. Kavitha, S.; Kaulgud, N. Quantum machine learning for support vector machine classification. *Evol. Intell.* **2024**, *17*, 819–828. [CrossRef]
90. Mienye, I.D.; Obaido, G.; Jere, N.; Mienye, E.; Aruleba, K.; Emmanuel, I.D.; Ogbuokiri, B. A Survey of Explainable Artificial Intelligence in Healthcare: Concepts, Applications, and Challenges. *Infor. Med. Unlocked* **2024**, *51*, 101587. [CrossRef]
91. Santoro, D.; Ciano, T.; Ferrara, M. A comparison between machine and deep learning models on high stationarity data. *Sci. Rep.* **2024**, *14*, 19409. [CrossRef] [PubMed]
92. Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [CrossRef]
93. Ahmed, N.; Wahed, M. The De-democratization of AI: Deep learning and the compute divide in artificial intelligence research. *arXiv* **2020**, arXiv:2010.15581.
94. Politou, E.; Alepis, E.; Patsakis, C. Profiling tax and financial behaviour with Big Data under the GDPR. *Comput. Law Secur. Rev.* **2019**, *35*, 306–329. [CrossRef]
95. Mienye, I.D.; Swart, T.G.; Obaido, G. Fairness Metrics in AI Healthcare Applications: A Review. In Proceedings of the 2024 IEEE International Conference on Information Reuse and Integration for Data Science (IRI), San Jose, CA, USA, 7–9 August 2024, IEEE: Piscataway, NJ, USA, 2024; pp. 284–289.
96. Wu, J.J.X. Algorithmic Fairness in Consumer Credit Underwriting: Towards a Harm-Based Framework for AI Fair Lending. *Berkeley Bus. LJ* **2024**, *21*, 65. [CrossRef]