Fusion of Multidisciplinary Research, An International Journal (FMR) , Vol. **3**, Iss. **1**, 2022
https://doi.org/10.63995/RVZF7165

FMR

ARTICLE

Mastering Neural Network Prediction for Enhanced System Reliability

Josef Baumgartner, Alexandra Schneider, Ulugbek Zhenis, Franz Jager, and Josef Winkler*

Institute of Instructional and School Development, Alpen-Adria University Klagenfurt, Klagenfurt, Austria

*Corresponding author: Josefwin4.kl@aau.at

(Received: 31 August 2021; Revised: 29 November 2021; Accepted: 02 January 2022; Published: 17 January 2022)

Abstract

Mastering neural network prediction is crucial for enhancing system reliability across various fields, from healthcare to autonomous driving. Neural networks, with their ability to learn and generalize from vast datasets, offer unparalleled predictive capabilities. By accurately forecasting system behaviors and potential failures, neural networks can significantly improve reliability and efficiency. For instance, in predictive maintenance, neural networks can analyze sensor data to predict equipment failures before they occur, reducing downtime and maintenance costs. Similarly, in healthcare, predictive models can identify patients at risk of developing certain conditions, enabling early interventions and better outcomes. To achieve high reliability, it is essential to address challenges such as overfitting, interpretability, and robustness of neural network models. Techniques like cross-validation, regularization, and the use of explainable AI (XAI) tools are vital in building trustworthy models. Moreover, continuous monitoring and updating of neural networks with new data ensure their predictions remain accurate over time. Integrating neural network predictions into system design and operational strategies can lead to more resilient and adaptive systems. By leveraging the power of neural networks, organizations can anticipate and mitigate risks more effectively, leading to enhanced reliability and performance in critical applications. This advancement underscores the transformative potential of neural networks in fostering more reliable and efficient systems.

Keywords: Interpretability; Neural Networks; Predictive Maintenance; Reliability; Robustness; System Efficiency; Trustworthy Models

Abbreviations: BPWDNN: Back Propagation with Weight Decay neural network, CNN: Convolutional Neural Network, EDT: Electrical Discharge Turning, FNN: Feed-forward Neural Network, GRNN: General Regression Neural Network, GRU: Gated Recurrent Unit, GDBPMNN: Gradient Descent Back Propagation with Momentum neural network, LSTM: Long Short-Term Memory, MLP: Multi-layer Perceptrons, NN: Neural Network, QRNN: Quantile Regression neural network, ReLU: Rectified Linear Unit, RNN: Recurrent Neural Networks, SHLFFNN: Single hidden layer feed-forward neural network

1. Introduction

In the era of modern machine learning, neural network prediction has emerged as a powerful tool for enhancing system reliability across diverse domains. These deep learning models leverage complex architectures and training regimes to capture intricate patterns and relationships within data, enabling accurate forecasting and decision-making [1, 2, 3, 4, 5].

Neural networks, encompassing feed-forward and recurrent architectures, have demonstrated remarkable proficiency in tasks ranging from translation to regression, outperforming traditional linear models. By delving into the fundamentals of these networks, exploring advanced training techniques, and conducting rigorous experiments, researchers and practitioners alike can unlock the full potential of neural network prediction for improved system performance and robustness [6].

2. Neural Network Fundamentals

Neural networks are a powerful computational technique inspired by the human brain's ability to learn and solve complex problems. At their core, neural networks consist of interconnected artificial neurons that process information through a series of weighted connections and activation functions (Fig. 1) [7].

2.1 Key Components of Neural Networks

1. **Neurons**: The fundamental units of a neural network, analogous to biological neurons in the human brain. Each neuron receives input signals, processes them, and produces an output signal.
2. **Synapses/Weights**: Connections between neurons that determine the strength and influence of the input signals on the output. These weights are adjusted during the training process to optimize the network's performance.
3. **Adder**: A computational unit that sums the weighted input signals received by a neuron.
4. **Activation Function**: A mathematical function applied to the summed input signals to introduce non-linearity and determine the neuron's output signal. Common activation functions include linear, sigmoid, rectified linear unit (ReLU), and hyperbolic tangent (tanh).

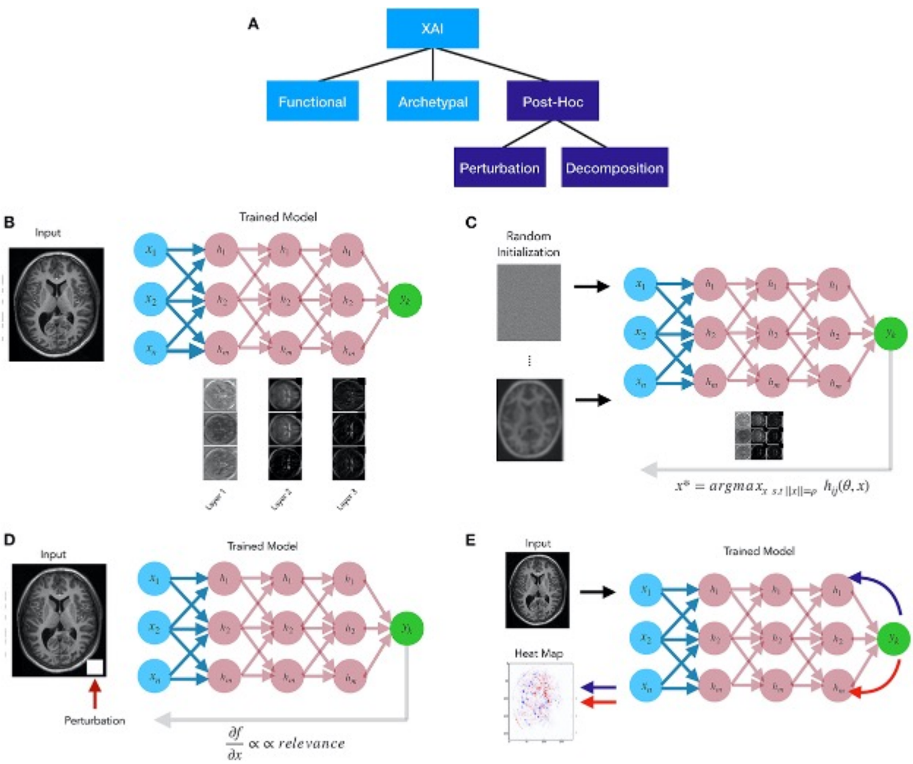

Figure 1. Explainable AI methods taxonomy

2.2 Neural Network Architectures

Neural networks can be classified into three main categories based on their architecture:

- **Single-Layer Feedforward Networks**: These networks have a single layer of neurons that directly map input signals to output signals. They are suitable for linearly separable problems.
- **Multilayer Feedforward Networks**: These networks consist of multiple layers of neurons, including input, hidden, and output layers. The hidden layers enable the network to learn and represent complex non-linear relationships between inputs and outputs.
- **Recurrent Neural Networks (RNNs)**: These networks incorporate feedback loops, allowing them to process sequential data and maintain an internal state. RNNs are particularly useful for tasks involving time series data, such as natural language processing and speech recognition.

2.3 Training Neural Networks

Neural networks can be trained using supervised or unsupervised learning algorithms:

- Supervised Learning: In this approach, the network is provided with labeled training data, consisting of input-output pairs. The weights are adjusted iteratively to minimize the error between the network's predictions and the desired outputs.
- Unsupervised Learning: In this approach, the network is trained on unlabeled data, and it must discover patterns and relationships within the data without explicit guidance.

Neural networks have found widespread applications in various domains, including digital signal processing, pattern recognition, classification, and regression tasks. Their ability to learn complex non-linear relationships from data has made them invaluable tools for enhancing system reliability and performance across diverse industries [7, 8, 9, 10, 11].

Feed-Forward Neural Networks Feed-forward neural networks, also known as multi-layer perceptrons (MLPs), are a fundamental type of neural network architecture that has been widely adopted for various machine learning tasks. These networks are characterized by their unidirectional flow of information, where data moves from the input layer through one or more hidden layers, and finally to the output layer [12, 13, 14].

2.4 Key Features of Feed-Forward Neural Networks

1. **Layers**: Feed-forward neural networks consist of multiple layers, including an input layer, one or more hidden layers, and an output layer. Each layer comprises a set of interconnected neurons or nodes.
2. **Connections**: The neurons in each layer are connected to the neurons in the subsequent layer through weighted connections. These weights determine the strength and influence of the connections between neurons.
3. **Activation Functions**: Each neuron in the hidden and output layers applies an activation function to the weighted sum of its inputs. Common activation functions include sigmoid, rectified linear unit (ReLU), and hyperbolic tangent (tanh). These functions introduce non-linearity, enabling the network to learn complex patterns and relationships.
4. **Forward Propagation**: During the forward propagation phase, the input data is fed into the input layer, and the activations are propagated through the network layer by layer until the output layer produces the final predictions or outputs.

2.5 Training Feed-Forward Neural Networks

Feed-forward neural networks are typically trained using supervised learning algorithms, such as backpropagation. The training process involves the following steps:

1. **Forward Pass**: The input data is propagated through the network, and the output is computed.
2. **Loss Calculation**: The difference between the predicted output and the true output (target) is calculated using a loss function, such as mean squared error (MSE) or cross-entropy loss.
3. **Backpropagation**: The error is propagated backward through the network, and the weights are adjusted using an optimization algorithm (e.g., gradient descent) to minimize the loss.
4. **Iteration**: The process is repeated for multiple epochs (iterations over the entire training dataset) until the network converges to an optimal set of weights that minimizes the loss.

Feed-forward neural networks have been successfully applied to a wide range of tasks, including image recognition, natural language processing, and regression problems. Their ability to learn complex non-linear mappings from input data to desired outputs has made them a powerful tool in various domains, contributing to enhanced system reliability and performance (depicted in Fig. 2) [15, 16, 17].

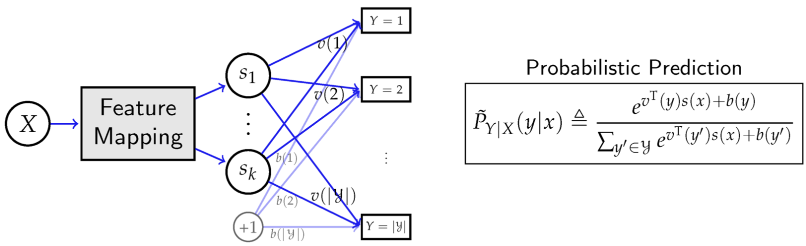

Figure 2. A deep neural network that uses data X to predict Y.

Recurrent Neural Networks Recurrent Neural Networks (RNNs) are a specialized type of neural network architecture designed to process sequential data, such as text, speech, and time series. Unlike feedforward neural networks, which process inputs independently, RNNs have an internal memory that allows them to remember and utilize information from previous inputs. This capability makes RNNs particularly well-suited for tasks involving sequential data, where the order and context of the inputs are crucial [18].

2.6 Key Characteristics of RNNs

1. **Sequential Processing**: RNNs process data step-by-step, maintaining a hidden state that acts as a memory. This hidden state is updated at each time step using the current input and the previous hidden state, allowing the network to capture information from past inputs.
2. **Weight Sharing**: RNNs use the same set of weights across all time steps, enabling them to share information throughout the sequence. This weight sharing mechanism enhances training efficiency and allows RNNs to generalize well to sequences of varying lengths.
3. **Feedback Loop**: RNNs incorporate a feedback loop, where the output from the previous step is fed as input to the current step. This feedback loop allows RNNs to remember and utilize information from previous inputs, making them suitable for tasks that require context and memory.

2.7 Advantages and Applications of RNNs

- **Handling Sequential Data**: RNNs excel at processing sequential data, such as natural language, speech, and time series data, where the order and context of the inputs are crucial.

- **Variable Input Length**: RNNs can process inputs of any length, making them versatile for tasks with varying sequence lengths.
- **Efficient Training**: By sharing weights across time steps, RNNs can be trained more efficiently compared to architectures that require separate weights for each time step.
- **Applications**: RNNs have been successfully applied in various domains, including speech recognition, language modeling, text generation, machine translation, and sequence analysis in biology.

2.8 Challenges and Variations

While RNNs offer significant advantages, they also face challenges such as:

- **Vanishing and Exploding Gradients**: During training, RNNs can suffer from vanishing or exploding gradient problems, which can hinder their ability to learn long-term dependencies effectively.
- **Computational Complexity**: RNNs can be computationally slower than other neural network architectures, especially for long sequences.

To address these challenges, variations of RNNs have been developed, including:

- **Long Short-Term Memory (LSTM)**: LSTMs are a type of RNN that uses a gating mechanism to control the flow of information, helping to mitigate the vanishing gradient problem and capture long-term dependencies more effectively.
- **Gated Recurrent Unit (GRU)**: GRUs are another variation of RNNs that use a simpler gating mechanism compared to LSTMs, offering a balance between performance and computational complexity.

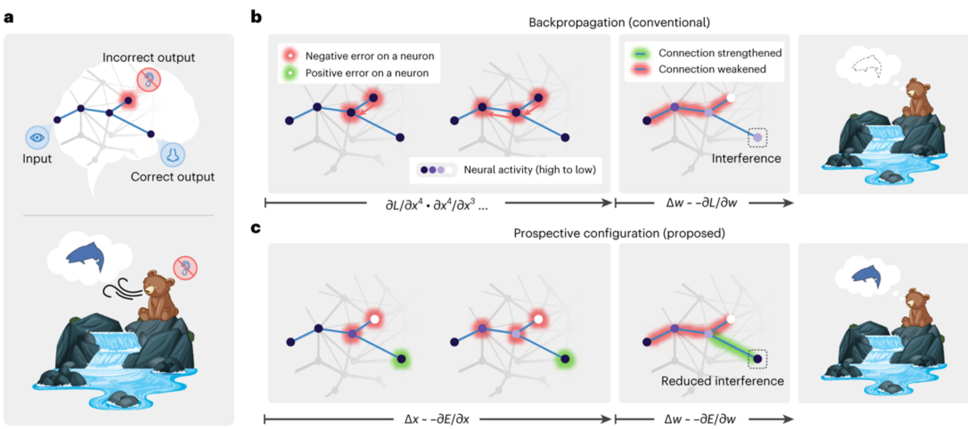

Figure 3. Task inducing interference during learning.

Neural network prediction using RNNs has proven to be a powerful approach for enhancing system reliability in various domains, particularly those involving sequential data (Fig. 3). By leveraging their ability to capture context and long-term dependencies, RNNs can provide accurate predictions and improve decision-making processes, ultimately contributing to enhanced system performance and robustness [19, 20].

3. Training Regimes

Effective training regimes are crucial for harnessing the full potential of neural networks for prediction tasks and enhancing system reliability (Fig. 4). The following key points highlight the importance and considerations of training neural networks:

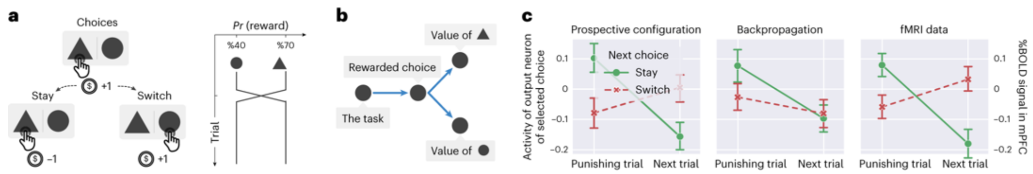

Figure 4. Reinforcement learning task

1. **Improved Accuracy**: Training a neural network model before using it for prediction can significantly improve its accuracy and performance. The training process allows the model to learn from data and adjust its internal parameters, enabling it to capture complex patterns and relationships more effectively.
2. **Customization and Scalability**: Neural networks offer flexibility in terms of customization and scalability. By training the network with specific data and requirements, it can be tailored to meet the unique needs of different prediction tasks and systems, ensuring optimal reliability and performance.
3. **Transfer Learning**: Pre-trained neural networks can serve as a starting point for new prediction tasks through transfer learning. This approach leverages the knowledge and representations learned from previous training, accelerating the training process and improving performance, especially when working with limited data.
4. **Suitability for Reliability Prediction**: Neural networks have proven to be effective for software reliability prediction, often outperforming analytical models. They require only failure history as input and can automatically develop an internal model of the failure process, adjusting the model complexity to match the data.
5. **Network Architecture Selection**: Choosing a suitable network architecture is a critical step in developing a neural network for reliability prediction. Algorithms like cascade-correlation can be employed to specify an appropriate architecture based on the problem and data characteristics.
6. **Training Data Selection**: The choice of training data plays a crucial role in the performance of the neural network. Two common approaches are generalization training, where the network learns from a representative subset of data, and prediction training, where the network is trained on the entire dataset.
7. **Training Algorithms**: Supervised learning algorithms, such as backpropagation, are commonly used to train neural networks for reliability prediction tasks. These algorithms iteratively adjust the network's weights and biases to minimize the error between the predicted and actual outputs.
8. **Adaptability and Complexity**: Neural networks can develop models of varying complexity at different phases of testing, adapting to the data and requirements. This flexibility contrasts with analytical models, which have a fixed complexity, making neural networks a more versatile and powerful approach for reliability prediction.
9. **Minimal Assumptions and Automatic Adaptation**: One of the main advantages of the neural network approach is that it is a "black-box" method, requiring minimal assumptions about the underlying data and processes. Additionally, neural networks can automatically adapt their model complexity to the data, eliminating the need for manual intervention or assumptions.

By carefully considering and implementing appropriate training regimes, neural networks can be effectively leveraged for accurate and reliable prediction tasks, contributing to enhanced system reliability across various domains [9, 11, 21].

4. Prediction Experiment

To assess the predictive capabilities of neural networks and their potential for enhancing system reliability, several studies have been conducted across various domains (Fig. 5). Here are some notable prediction experiments:

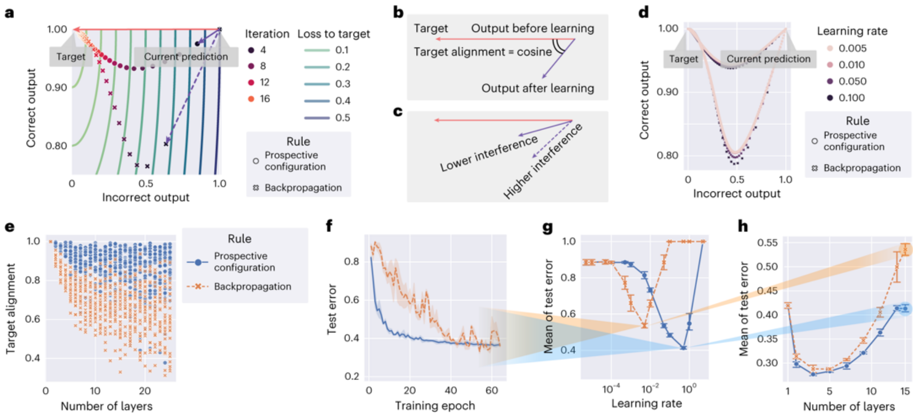

Figure 5. Simulation of the network

4.1 Predicting COVID-19 Patient Outcomes

- **Objective**: Develop a neural network classification model to predict survival or death of hospitalized COVID-19 patients based on initial laboratory findings, comorbidities, and demographics.
- **Dataset**: 634 hospitalized COVID-19 patients (68.77% male, 31.23% female), with 71 deaths.
- **Input Features**: 12 laboratory blood test results, 6 comorbidities, age, and gender.
- **Methodology**: 11 neural network models with different threshold values (0 to 1 in 0.1 steps).
- **Results**:
  - **Optimal Model**: Accuracy of 87.78%, precision of 96.37%, sensitivity of 90.07%, specificity of 62.16% (threshold = 0.7).
  - **Hypothesis Testing**: Mean values of age, erythrocytes, hemoglobin, hematocrit, leukocytes, neutrophil granulocytes, CRP, and D-dimer were significantly higher in patients who died.
  - **95% Confidence Intervals**: Calculated for mean values of all laboratory features and age for survivors and non-survivors.

4.2 Predicting Electrical Discharge Turning (EDT) Process Responses

- **Objective**: Apply five different neural network (NN) models to predict material removal rate (MRR) and overcut (OC) responses in the EDT process.
- **NN Models**: Feed-forward neural network (FNN), convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM) based RNN, and general regression neural network (GRNN).

- **Input Parameters**: Magnetic field, pulse current, pulse duration, and angular velocity.
- **Results**:
  - **LSTM**: Most accurate predictions for MRR and OC, highest R-squared and adjusted R-squared, lowest RMSE and relative RMSE.
  - **CNN and RNN**: Comparable prediction accuracy.
  - **FNN**: Moderately satisfactory results.
  - **GRNN**: Average prediction results but extremely robust and repetitive performance.

4.3 Predicting Antitubercular Activity of Oxazolines and Oxazoles Derivatives

- **Objective**: Evaluate five neural network models for predicting antitubercular activity of Oxazolines and Oxazoles derivatives.
- **NN Models**: Single hidden layer feed-forward neural network (SHLFFNN), Gradient Descent Back Propagation neural network (GDBPNN), Gradient Descent Back Propagation with Momentum neural network (GDBPMNN), Back Propagation with Weight Decay neural network (BPWDNN), and Quantile Regression neural network (QRNN).
- **Results**:
  - **QRNN**: Outperformed other models, lowest RMSE, highest R^2 values.
  - **GDBPMNN**: Good performance, comparable to QRNN.
  - **SHLFFNN and GDBPNN**: Largest RMSE and error spread.
  - **Statistical Analysis**: No significant performance difference between QRNN, GDBPMNN, and BPWDNN; other model comparisons showed significant differences.

These prediction experiments demonstrate the effectiveness of neural networks in various domains, including healthcare, manufacturing, and drug discovery. By leveraging different architectures and training regimes, neural networks can provide accurate predictions and insights, contributing to enhanced system reliability and performance [22, 23].

5. Results and Analysis

The results and analysis of various prediction experiments involving neural networks have shed light on their effectiveness and potential for enhancing system reliability across diverse domains as explained in Table 1 [13].

5.1 Predicting COVID-19 Patient Outcomes

- The optimal neural network model achieved an accuracy of 87.78%, precision of 96.37%, sensitivity of 90.07%, and specificity of 62.16% in predicting survival or death of hospitalized COVID-19 patients.
- Hypothesis testing revealed significant differences in the mean values of age, erythrocytes, hemoglobin, hematocrit, leukocytes, neutrophil granulocytes, CRP, and D-dimer between survivors and non-survivors.
- The study calculated 95% confidence intervals for the mean values of all laboratory features and age, providing a measure of uncertainty and statistical significance.

Table 1. Predicting Electrical Discharge Turning (EDT) Process Responses

Neural Network Model	Material Removal Rate (MRR)	Overcut (OC)
LSTM	Most accurate predictions	Most accurate predictions
CNN, RNN	Comparable prediction accuracy	Comparable prediction accuracy
FNN	Moderately satisfactory results	Moderately satisfactory results
GRNN	Average prediction results but extremely robust and repetitive performance	Average prediction results but extremely robust and repetitive performance

5.2 Predicting Antitubercular Activity of Oxazolines and Oxazoles Derivatives

- The Quantile Regression neural network (QRNN) outperformed other models, exhibiting the lowest RMSE and highest $R\hat{2}$ values.
- The Gradient Descent Back Propagation with Momentum neural network (GDBPMNN) demonstrated good performance, comparable to QRNN.
- Single hidden layer feed-forward neural network (SHLFFNN) and Gradient Descent Back Propagation neural network (GDBPNN) showed the largest RMSE and error spread.
- Statistical analysis revealed no significant performance difference between QRNN, GDBPMNN, and Back Propagation with Weight Decay neural network (BPWDNN), while other model comparisons showed significant differences.

These experiments highlight the versatility and effectiveness of neural networks in tackling diverse prediction tasks, ranging from healthcare to manufacturing and drug discovery. By leveraging different architectures and training regimes, neural networks can provide accurate predictions, contributing to enhanced system reliability and performance across various domains [24, 25, 26].

6. Advantages of Neural Networks

Neural networks offer numerous advantages that make them invaluable tools for enhancing system reliability across various domains (Fig. 6):

- **Accurate Classification and Pattern Recognition**: Neural networks excel at accurately classifying inputs into different categories by learning the distinguishing features and patterns, often outperforming traditional methods [16]. This capability is crucial for tasks such as image recognition, speech recognition, and anomaly detection, enabling reliable decision-making and automation.
- **Non-linear Modeling and Numerical Prediction**: Neural networks can effectively learn non-linear relationships within data and make accurate predictions of numerical values. They leverage techniques like ensembles and transfer learning to improve their predictive capabilities, making them suitable for applications like sales forecasting, stock market predictions, and customer churn forecasting [13, 14].
- **Data Generation and Synthesis**: Neural networks can generate new data samples, such as images, text, or sounds, by learning the underlying probability distribution of the outputs [8]. This ability has applications in fields like computer vision, natural language processing, and creative industries, enabling the generation of synthetic data for training or content creation.
- **Reinforcement Learning and Decision-Making**: Neural networks are crucial for training age-

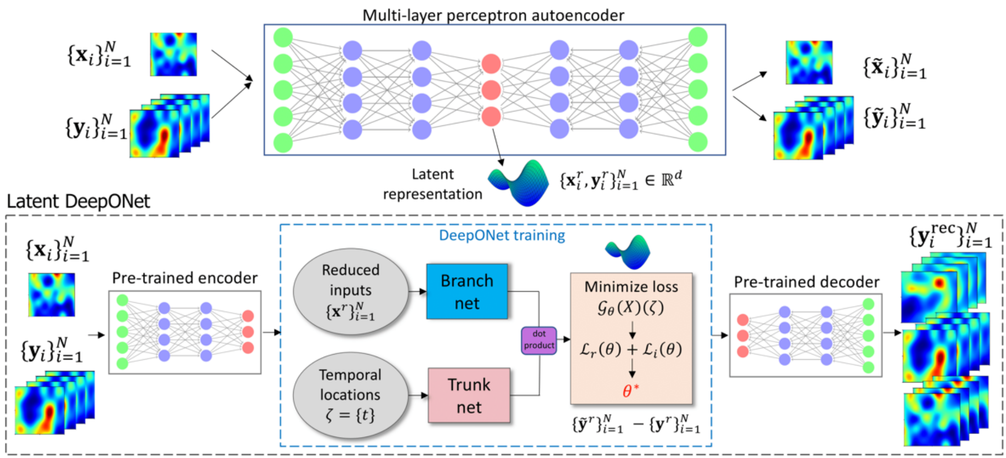

Figure 6. Latent DeepONet (L-DeepONet) framework for learning deep neural operators on latent spaces

nts to perform actions that maximize a reward in complex environments, enabling efficient decision-making [11]. This capability is valuable in domains such as robotics, gaming, and autonomous systems, where reliable decision-making is essential for system performance and safety.

- **Automated Feature Engineering and Non-linear Feature Extraction**: Neural networks bring the advantages of automated feature engineering and the ability to capture non-linear features, which linear regression models cannot [9]. This makes them more powerful and accurate for various tasks, including pattern recognition, automation, personalization, anomaly detection, and natural language understanding.
- **Cost Reduction and Efficiency**: By leveraging neural networks for tasks such as automation, personalization, and anomaly detection, businesses can reduce operational costs and improve efficiency, ultimately contributing to enhanced system reliability and profitability.
- **Solving Complex Problems**: Neural networks can be used to solve problems that the human brain excels at, such as recognizing sounds, pictures, or text, and can be employed to extract features for clustering and classification [17]. This capability makes them valuable for applications in domains like healthcare, finance, and security, where accurate decision-making is critical for system reliability.

Neural network prediction leverages these advantages, enabling accurate forecasting, decision making, and automation across various industries, ultimately contributing to enhanced system reliability and performance.

7. Challenges and Future Directions

Despite the remarkable success of neural networks in various prediction tasks, several challenges remain that need to be addressed to further enhance their reliability and performance. Additionally, ongoing research is exploring new directions to push the boundaries of what is achievable with neural network prediction (Fig. 7).

7.1 Interpretability and Explainability

One of the major challenges with neural networks is their lack of interpretability and explainability. These models often operate as "black boxes," making it difficult to understand how they arrive at their

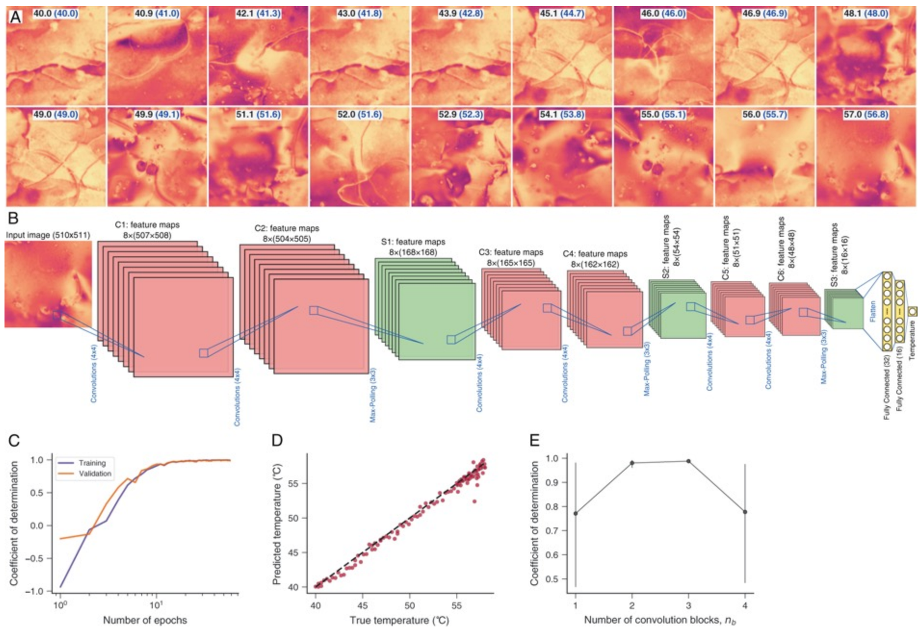

Figure 7. Learning physical properties of liquid crystals with deep convolutional neural networks

predictions. This lack of transparency can be a significant barrier in high-stakes applications, such as medical diagnosis or autonomous systems, where understanding the decision-making process is crucial for trust and accountability. Researchers are actively working on developing techniques to make neural networks more interpretable and explainable, such as:

- **Attention Mechanisms**: These mechanisms allow neural networks to focus on specific parts of the input data, providing insights into the decision-making process.
- **Saliency Maps**: These visual representations highlight the regions of the input data that contribute most to the network's predictions, aiding in understanding the model's behavior.
- **Concept Activation Vectors (CAVs)**: CAVs are learned representations that capture high-level concepts within the network, enabling interpretability by associating predictions with human-understandable concepts.

7.2 Robustness and Adversarial Attacks

Another challenge is ensuring the robustness of neural networks against adversarial attacks. These attacks involve introducing imperceptible perturbations to the input data, which can cause the network to make incorrect predictions. Adversarial attacks pose a significant threat to the reliability of neural networks in critical applications, such as cybersecurity and autonomous systems. Researchers are exploring various techniques to enhance the robustness of neural networks, including:

- **Adversarial Training**: This involves training the network on adversarial examples, making it more resilient to such attacks.
- **Defensive Distillation**: This technique aims to smooth the network's decision boundaries, reducing the effectiveness of adversarial perturbations.
- **Certified Robustness**: This approach provides provable guarantees of robustness against adversarial attacks within a specific threat model.

7.3 Efficient and Scalable Training

As neural networks become larger and more complex, training them efficiently and at scale becomes increasingly challenging. This is particularly relevant in domains like natural language processing and computer vision, where models can have billions of parameters. Researchers are exploring various techniques to address this challenge, such as:

- **Distributed Training**: This involves parallelizing the training process across multiple computational resources, such as GPUs or TPUs, to accelerate the training process.
- **Model Compression**: This involves reducing the size of the neural network model, either by pruning redundant connections or by quantizing the weights, without significantly impacting its performance.
- **Federated Learning**: This approach enables training neural networks on decentralized data, such as data from multiple devices or organizations, without sharing the raw data, addressing privacy and scalability concerns.

As research in these areas progresses, neural network prediction is poised to become even more reliable, interpretable, and efficient, further enhancing its potential for applications across diverse domains and contributing to improved system reliability and performance.

8. Conclusion

The rapid advancements in neural network prediction have paved the way for remarkable enhancements in system reliability across diverse domains. By leveraging the power of deep learning architectures, neural networks have demonstrated an unparalleled ability to capture intricate patterns, learn complex non-linear relationships, and provide accurate predictions. These capabilities have proven invaluable in applications ranging from healthcare and manufacturing to drug discovery and autonomous systems. As we continue to explore the depths of neural network prediction, ongoing research efforts are focused on addressing key challenges, such as interpretability, robustness against adversarial attacks, and efficient scalable training. By overcoming these hurdles, neural networks will emerge as even more reliable and trustworthy tools, further solidifying their role in enhancing system performance and reliability. The future holds immense promise for neural network prediction, paving the way for groundbreaking advancements across various industries and applications.

References

[1] Jianzhou Wang, Feng Liu, Yiliao Song, and Jing Zhao. "A novel model: Dynamic choice artificial neural network (DCANN) for an electricity price forecasting system". In: *r2* 48 (2016), pp. 281–297.
[2] Gerald Warnecke and R Kluge. "Control of tolerances in turning by predictive control with neural networks". In: *Journal of Intelligent Manufacturing* 9 (1998), pp. 281–287.
[3] Jason Deutsch and David He. "Using deep learning-based approach to predict remaining useful life of rotating components". In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48.1 (2017), pp. 11–20.
[4] Kamran Javed. "A robust and reliable data-driven prognostics approach based on extreme learning machine and fuzzy clustering". PhD thesis. Université de Franche-Comté, 2014.
[5] Nesim Yilmaz, Tuncer Demir, Safak Kaplan, and Sevilin Demirci. "Demystifying Big Data Analytics in Cloud Computing". In: *Fusion of Multidisciplinary Research, An International Journal (FMR)* 1.1 (2020), pp. 25–36.

[6] Shuochao Yao, Yiran Zhao, Aston Zhang, Lu Su, and Tarek Abdelzaher. "Deepiot: Compressing deep neural network structures for sensing systems with a compressor-critic framework". In: *Proceedings of the 15th ACM conference on embedded network sensor systems.* 2017, pp. 1–14.

[7] Thomas G Calderon and John J Cheh. "A roadmap for future neural networks research in auditing and risk assessment". In: *International Journal of Accounting Information Systems* 3.4 (2002), pp. 203–236.

[8] Yang Liu, Nam Dinh, Yohei Sato, and Bojan Niceno. "Data-driven modeling for boiling heat transfer: using deep neural networks and high-fidelity simulation results". In: *Applied Thermal Engineering* 144 (2018), pp. 305–320.

[9] Jason Brownlee. *Long short-term memory networks with python: develop sequence prediction models with deep learning.* Machine Learning Mastery, 2017.

[10] Friedrich Stephanie and Louisa Karl. "Incorporating Renewable Energy Systems for a New Era of Grid Stability". In: *Fusion of Multidisciplinary Research, An International Journal (FMR)* 1.1 (2020), pp. 37–49.

[11] Monica Lam. "Neural network techniques for financial performance prediction: integrating fundamental and technical analysis". In: *Decision support systems* 37.4 (2004), pp. 567–581.

[12] Christine K Lee, Ira Hofer, Eilon Gabel, Pierre Baldi, and Maxime Cannesson. "Development and validation of a deep neural network model for prediction of postoperative in-hospital mortality". In: *Anesthesiology* 129.4 (2018), pp. 649–662.

[13] Rashid Mijumbi, Sidhant Hasija, Steven Davy, Alan Davy, Brendan Jennings, and Raouf Boutaba. "Topology-aware prediction of virtual network function resource requirements". In: *IEEE Transactions on Network and Service Management* 14.1 (2017), pp. 106–120.

[14] ChoHee Kim, Donghyun Gwan, and Minho Sena Nam. "Beyond the Atmosphere: The Revolution in Hypersonic Flight". In: *Fusion of Multidisciplinary Research, An International Journal (FMR)* 2.1 (2021), pp. 152–163.

[15] Zhengliang Xiang, Yuequan Bao, Zhiyi Tang, and Hui Li. "Deep reinforcement learning-based sampling method for structural reliability assessment". In: *Reliability Engineering & System Safety* 199 (2020), p. 106901.

[16] Kenneth O Stanley, Jeff Clune, Joel Lehman, and Risto Miikkulainen. "Designing neural networks through neuroevolution". In: *Nature Machine Intelligence* 1.1 (2019), pp. 24–35.

[17] Abbas Sheikhtaheri, Farahnaz Sadoughi, and Zahra Hashemi Dehaghi. "Developing and using expert systems and neural networks in medicine: a review on benefits and challenges". In: *Journal of medical systems* 38 (2014), pp. 1–6.

[18] Stylianos Ioannis Mimilakis, Konstantinos Drossos, Tuomas Virtanen, and Gerald Schuller. "Deep neural networks for dynamic range compression in mastering applications". In: *Audio Engineering Society Convention 140.* Audio Engineering Society. 2016.

[19] Long Wang, Zijun Zhang, Huan Long, Jia Xu, and Ruihua Liu. "Wind turbine gearbox failure identification with deep neural networks". In: *IEEE Transactions on Industrial Informatics* 13.3 (2016), pp. 1360–1368.

[20] Manuel André, João Margarida, Heitor Garcia, and Augusto Dante. "Complexities of Blockchain Technology and Distributed Ledger Technologies: A Detailed Inspection". In: *Fusion of Multidisciplinary Research, An International Journal (FMR)* 2.1 (2021), pp. 164–177.

[21] Hao Peng and Xiaoli Bai. "Artificial neural network–based machine learning approach to improve orbit prediction accuracy". In: *Journal of Spacecraft and Rockets* 55.5 (2018), pp. 1248–1260.

[22] Simon Haykin. "Neural networks expand SP's horizons". In: *IEEE Signal Processing Magazine* 13.2 (1996), pp. 24–49.

[23] Wei Jiang and Hans D Schotten. "Neural network-based fading channel prediction: A comprehensive overview". In: *IEEE Access* 7 (2019), pp. 118112–118124.

[24] Adrichem De Jong, Mark Jansen, Jeffrey Van Dijk, and Johannes Meyer. “Analysis of Innovative Practices in Advanced Materials and Structural Engineering”. In: *Fusion of Multidisciplinary Research, An International Journal (FMR)* 2.1 (2021), pp. 178–188.
[25] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Abubakar Malah Umar, Okafor Uchenwa Linus, Humaira Arshad, Abdullahi Aminu Kazaure, Usman Gana, and Muhammad Ubale Kiru. “Comprehensive review of artificial neural network applications to pattern recognition”. In: *IEEE access* 7 (2019), pp. 158820–158846.
[26] Alberto Bosio, Paolo Bernardi, Annachiara Ruospo, and Ernesto Sanchez. “A reliability analysis of a deep neural network”. In: *2019 IEEE Latin American Test Symposium (LATS)*. IEEE. 2019, pp. 1–6.