5 Advantages of Integrating Neural Networks into Your Digital Twins

The future of work is automating repetitive, time-consuming tasks or activities. Digital transformation solutions and artificial intelligence (AI) provide a pathway to intelligently automating workflows and driving autonomous decision making.

The Digital Twin is a solution that encompasses all the above. The World Economic Forum named the digital twin as a defining solution with the capacity to improve systemic operations in 2015. Since then, the digital twin has evolved from a data exchange and problem-solving virtual mirror to cutting-edge solutions that leverage neural networks for building complex rule-based logic, predicting values or results to optimize operational systems.

The integration of neural networks extends the capabilities and application possibilities of the average digital twin. Getting the best out of your applications of neural networks within your digital twin starts with proper integration and here are some tips to help you with that.

1. Integrating Intelligent Behavior – The digital twin provides enterprises with a digital mirror to monitor and manage real-time operations and processes. Enterprises deploy digital twins for diverse reasons and to achieve multiple goals. Examples include as remote monitoring tool or to predict future performances and the choices needed to actualize optimized performances.

   Applying the digital twin as a prediction or performance optimization solution relies on intelligent objects and the ability to embed these objects with decision logic. For complex digital twin applications, defining custom logic for rules is a difficult, time-consuming task. These difficulties are traced to the process of capturing the diverse factors that influence decision making into a rule.

   Integrating neural networks into digital twin solutions provides you with an easier pathway to building complex logic. With a neural network, the process of building complex logic is by-passed and the neural network uses historical data to make the required decisions. The accuracy levels

of the neural network are also improved through reoccurring trainings using simulated or real-time data.

2. Aggregating Training Data – A machine, system, or model equipped with artificial intelligence capabilities rely on clean and labeled data to improve its decision-making capabilities. Hence, access to clean and labeled data is crucial to the application of AI in achieving the smart factory or automating workflows.

   Leveraging a digital twin within workspaces provide enterprises with the means to capture operational data in real-time. But capturing data is just the first step to gaining insight from data. Putting captured data to work within AI frameworks involves cleaning, labeling, and providing contextual insight from the captured data. Integrating neural networks within a digital twin empowers enterprises with the tools to generate simulated data to train AI algorithms.

3. Automating Workflows – Handling time-consuming, repetitive tasks leads is one of the leading causes of human error across the industrial sector. Thus, one of the major selling points of AI-powered solutions is its ability to support automation by leaving systems to make and take accurate decisions.

   Neural networks enable enterprises that utilize digital twins to automate the development of rules or decision logic within simulation and digital twin models and to monitor performance data to automate real-time decision-making.

4. Leverage Unsupervised Learning – Industry 4.0 focuses on the smart factory's ability to function optimally without human supervision. Thus, AI and its subset, neural networks, have important roles to play in implementing Industry 4.0 business models within the factory floor. Pairing digital twins with neural networks brings unsupervised learning to automated processes.

   With properly trained neural networks, digital twins can make inferences or provide accurate solutions to complex challenges when hidden factors may exist. The neural network is capable of applying reinforcement learning to develop policies that lead to optimized value creation or profit. For example, a neural network can provide time estimates to guide time-sensitive tasks. In a situation where a production job must not be started unless it can be completed within a specified timeline. A neural network will take all factors involved with the production cycle into consideration when predicting estimated completion times. The estimated timeline then provides

the information needed to either start the job or negotiate for an achievable completion time span.

5. Validate and Evaluate AI Algorithms – The digital twin provides enterprises with a virtual recreation of physical systems of facilities. The virtual platform then becomes a safer environment for evaluating the impact of adding new operational plans, schedules, or systems to an existing facility.

   The digital twin can also be used to test, as well as, train AI algorithms to ensure they do not perform optimally with a singular dataset but perform poorly with other dataset options. Here, a training data set, validation data set, and test data set are used to evaluate your AI algorithms before deploying them within your operational systems

Getting Started with Digital Twins and Neural Networks

Simulation and digital twin models provide enhanced analytical capabilities to enterprises interested in optimizing everyday processes. Application opportunities include in planning and scheduling, predictive maintenance, remote monitoring, data-driven plant optimization, validation and testing etc.

The inclusion of neural networks creates a symbiotic relationship between simulation and AI. Neural networks simplify complex logic creation and automate workflows within simulation and digital twin models. Simulation and digital twins provide a fertile platform for developing synthetic training data, training, and evaluating neural networks. You can get started with evaluating and training your neural networks by developing synthetic training data with Simio.