East Journal of Engineering
ISSN: 3079-9376

Volume, Issue
Publisher: East Publication & Technology
DOI: https://doi.org/10.63496/eje.Vol1.Iss3.121

Design and Evaluation of a Machine Learning-Based Decision-Making Framework for Robotics

Islam Alharreem[1] , **Haneen Radad**[2]*

[1]Department of Control and Robotics Engineering, Faculty of Engineering and Technology, Al Zaytona University of Science and Technology, Salfit Street, Al-Laban Al-Sharqiya-Salfit, Palestine, islam.alhareem@gmail.com

[2] Department of Control and Robotics Engineering, Faculty of Engineering and Technology, Al Zaytona University of Science and Technology, Salfit Street, Al-Laban Al-Sharqiya-Salfit, Palestine, haneen.slsl.2003@gmail.com.
*Corresponding Author

Received: 01/06/2025, **Revised**: 10/06/2025, **Accepted**: 11/06/2025, **Published**: 16/06/2025

Abstract:

Decision-making systems are of utmost importance for the autonomy of robotics so that robots may negotiate adaptive and complicated environments with very little human interference. This research presents a detailed study of and development of an autonomous robotic decision-making system through generalized AI techniques such as Neural Networks, Support Vector Machines, and Decision Trees. The design of a computer system capable of interpreting a variety of sensory inputs consisting of obstacle distance, battery levels, wheel speed, type of terrain, avenue weather conditions, ambient temperature, and robot tilt constitutes the main objective of this research so that it may dependably decide on the best action to take during a robotic mission. The study uses a carefully chosen dataset with 500 balanced observation points produced from realistic robot operations. Various advanced data preprocessing techniques, such as normalization, noise removal, and feature selection, were performed to improve the quality of the models. Performance-wise, the ANN model proved to be the most accurate (99%), most precise, and had the highest recall, thus clearly outperforming SVMs and DTs. Nevertheless, the SVM model especially finds its use in working with system requirements that need classification, interpretability, and computational efficiency, whereas, in contexts where safety is paramount, a DT model provides clear and transparent decision logic. The study advances the paradigm of AI-driven decision-making to solve intermediate-level decision problems for autonomous, flexible, and efficient robotic operations. More lines of research could go into hybrid AI and real-time implementation in critical sectors like disaster response and hazardous material management.

***Keywords*: Robotic Decision Making, Artificial Neural Networks, Support Vector Machines, Decision Trees, Autonomous Robotics.**

1. Introduction

Decision-making systems are an important component of intelligent robot platforms that enable machines to cope with dynamic and complex environments. In addition, as robotics applications continue to develop rapidly for industrial, medical, logistics, and rescue purposes, machine learning approaches for interpreting sensory data and sensing contextual information have arisen that can help create effective judgments about the appropriate response to identified objectives or mitigate potential risks [1].

A robotic decision-making system is a computational and algorithmic framework that allows a robot to process sensor data, consider a given environment, and adopt the best course of action (Russell & Norvig, 2016). Robotic decision-making systems typically employ diverse approaches in the field of artificial intelligence (AI)—based on rule-based reasoning, artificial neural networks, machine learning (ML), and reinforcement learning (RL)—to support adaptive and context-aware behavior (Russell & Norvig, 2016) [2].

Considering uncertainty (uncertainty-mediated decision making) as an important parameter of robotic performance, safety, and energy efficiency is a key consideration in some systems, such as: In the presence of obstacles or variable weather conditions, a robot can react rapidly and safely to changes in conditions, which can result from situations being simultaneously seen, observed and measured by the interacting robot. For this

25 The author(s). Published by East Publication and Technology. This open-access article is distributed under the terms of the Creative bution 4.0 International License. Which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

reason, developing effective decision models requires the capacity for collecting multiple dimensions of sensory data coupled with advanced analytical techniques that allow for the accurate capture of key behavioral styles and context-relevant cues [3].

In this paper, a structured dataset modeled real-world robotic operations under different environmental and internal conditions is investigated. The dataset contains various variables such as obstacle distance, battery level, wheel speed, terrain type, weather condition, wind speed, ambient temperature, robot tilt, mission type, and mission outcome (success or failure). A total of 100 operational cases are studied, which provide a rich experimental environment for the development of data-driven models for the correlation of environmental variables with decision outcomes.

By analyzing this dataset, machine learning techniques are used to identify variables that influence mission success or failure most, e.g., low battery levels, extreme robot tilt angles or demanding terrain may significantly contribute to failures during operational missions. Based on this analysis, adaptive artificial intelligence systems are trained on the data based on operational experience, which can learn from operational experience over time to improve decisions.

This research seeks to enable such flexibility by introducing a prototype AI-based decision-making model that can map environmental and sensory inputs to mission outcomes. Through proposed model not only increases the robotic efficiency in complex settings, decreases the operational errors but also improves the task execution, which can be with high stakes during rescue or high hazardous materials handling.

Ultimately, this study contributes to a deeper understanding of robotic decision-making mechanisms and establishes a solid, data-driven foundation for the development of more autonomous, reliable, and responsive robotic systems.

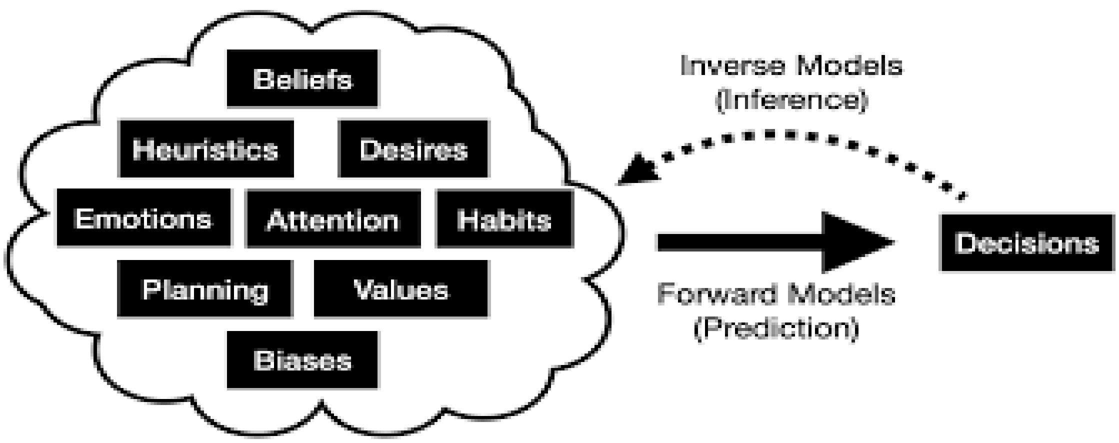

Figure 1: The forward and inverse models of cognitive science [4].

In this study, we discuss cognitive science research on forward models of how people make decisions and inverse models, which humans employ to reason about other agents. Cognitive scientists are increasingly employing computational methods such as probability theory, reinforcement learning, and statistical machine learning to describe forward and inverse models of human decision-making. This opens up prospects for cross-talk and collaboration between cognitive science and control studies.

Engineers designing automated systems – as both forward and inverse models – we will also highlight how recent work in computational cognitive science has emphasized formalisms that will be very familiar to researchers coming from a background of optimization and control. Cognitive scientists increasingly use ideas from probability theory, statistical machine learning, and reinforcement learning in specifying models of human cognition. This creates an opportunity to develop a common language for describing the behavior of both humans and machines and supports easier integration of insights from cognitive science into control[5], [6].

2. Related Work

By combining machine learning, control theory, and artificial intelligence, autonomous robotics develops intelligent robots capable of operating in changing conditions with limited human assistance. Autonomous robotics is based on machine learning algorithms that enable tasks such as sensing, navigation, and adaptive decision-making. These modern algorithms are used in many structural sectors, such as cooperative trees (DT), random forests (RF), advanced communication tools (SVM), and artificial neural networks (ANN), with a focus on comparative evaluation and examples. This review examines how such algorithms are used in studies [7].

2.1. Review of Major Studies

Cai et al. (2021) used the Denavit_Hartenberg (D-H) parameter approach to developing a kinematic control framework for robotic manipulators, Their methodology used a structured, rule-based design to make joint angle and robotics link transformation calculations easier, Despite not using machine learning straightforward, the study's hierarchical logic closely resembles a decision tree (DT) approach in terms of branching and state evaluation, Their simulation demonstrated how these structured principles enhanced trajectory responsiveness and accuracy, especially in areas with lots of obstacles, The significance of interpretable, lightweight models that work in real-time applications was underlined by the authors [7].

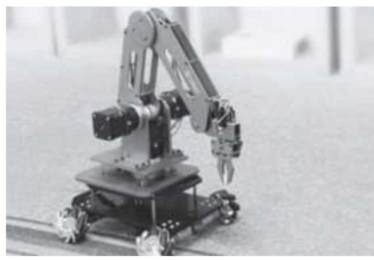

Figure 2: Kinematic model of the manipulator by D-H parameters

Francis et al. (2022) applied all kinds of ML techniques, including ANN and SVM, to analyze gas sludge (GSL) using a mobile robot. The capabilities of models concerning dealing with the noisy nature of volatile chemical data were assessed. In the binary case, whereas SVMs could give a quite accurate classification, ANNs were found to be more flexible and more capable of learning patterns of greater complexity to identify gas sources and intensity. Hence, the research states that, with multi-source cases, supervised learning along with probabilistic mapping greatly improves the accuracy of the localization algorithms. Thus, the results indicate the ANN scored a 91% accuracy among others. The paper details the diagrammatic configurations of the system, as the figures could not be extracted directly [8].

Artificial Neural Networks (ANN) and Support Vector Machines (SVM) have been observed with prominence in autonomous robotics throughout the literature of our present day. Francis et al. (2022) documented the SVM to be effective in classifying noisy sensor readings in the context of gas source localization: It defines its boundary very tightly and can therefore be employed faithfully to localize many gas sources in an area, employing probabilistic mapping techniques. In parallel, Palanivel and Muthulakshmi (2024) incorporated ANN models into a quantum reinforcement learning framework (P-QTFPR-DM) to enhance its real-time decision-making. Their Q-learning algorithm based on ANN converged significantly quicker than conventional models and did so with enhanced navigation performance. This discussion suggests how the two can complement each other: On the one hand, the SVM is particularly suited to well-structured classification tasks such as filtering sensor inputs and environment segmentation. On the other hand, ANN-based approaches are best suited to policy learning and dynamic control where flexibility is a problem. With ANN handling long-term learning and SVM providing firm perception, more and more systems are emerging at the boundary where their synergistic approach toward future robotics lies [9]

Palanivel and Muthulakshmi (2024) proposed a quantum-based learning model, P-QTFPR-DM, based Artificial Neural Networks approach with quantum computing. This model improves learning convergence of reinforcement learning through quantum state-value estimation and fractal-like prioritized experience replay.

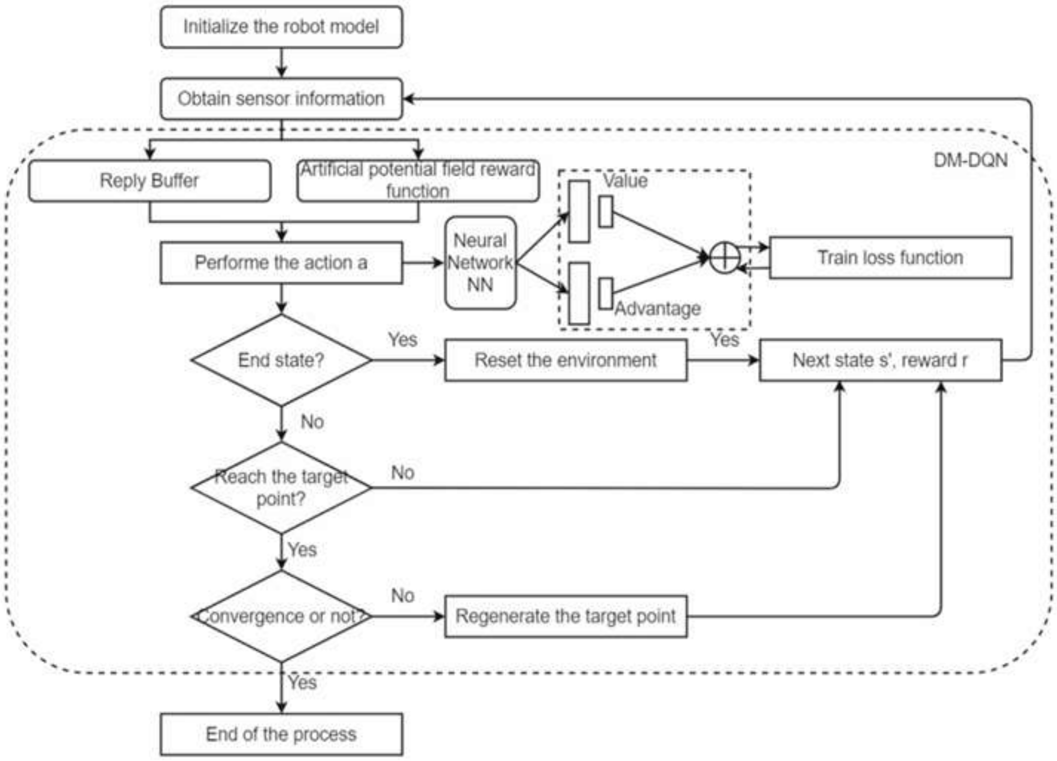

Figure 3: Gas localization configuration using SVM and ANN models [10].

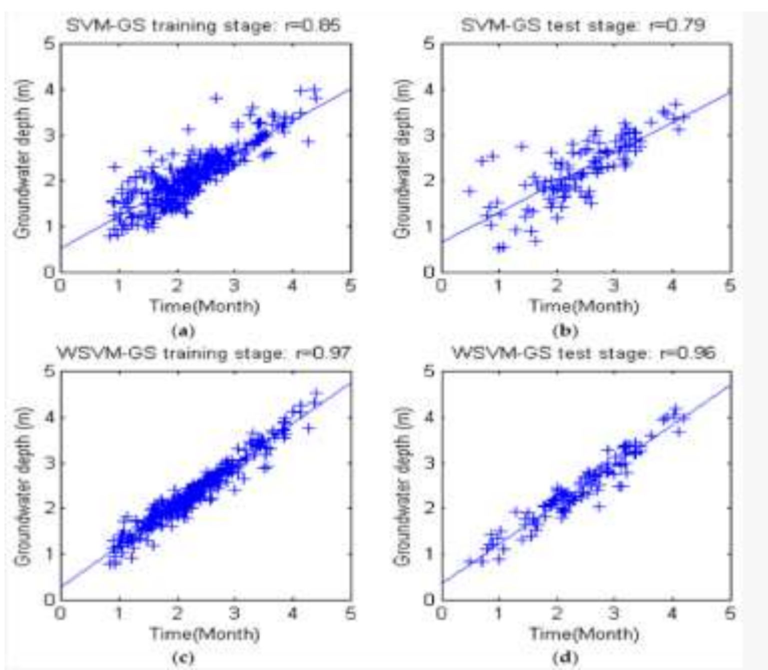

Figure 4: The scatter plot of the simulated data and the actual data of each model (a) the training stage of SVM-GS; (b) the test stage of SVM-GS, (c) the training stage of WSVM-GS; (d) the test stage of WSVM-GS [11].

Concerning ANNs, a quantum circuit handling state-action exploration through entangled superposition comprises ANNs. Their simulations demonstrated a 30% improvement in learning time with a success rate of

95% for reaching targets. This architecture illustrates the possibility that quantum-powered powered scalable, and efficient robotic learning systems can be achieved by quantum ANN topologies.

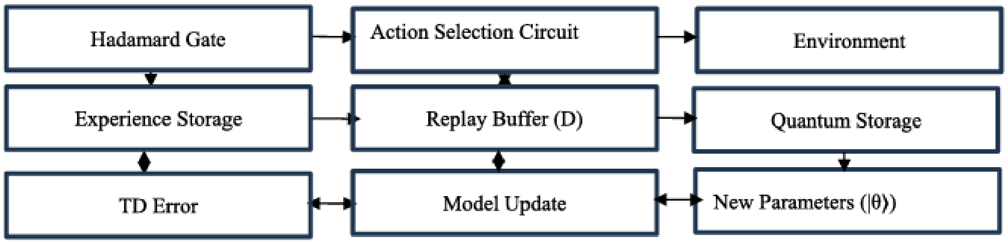

Figure 5: Quantum ANN reinforcement model architecture.

The DM-SPP-4 algorithm, which uses grid-based heuristics and deterministic logic, was presented by Dhouib (2024) for mobile robot path planning. The algorithm calculates optimal pathways faster than traditional techniques like A* or Dijkstra by using a contingency matrix. When tested on 41x41 grid maps, DM-SPP-4 outperformed baseline models in static obstacle situations, producing valid pathways in less than 0.05 seconds. Despite not being a learning-based model, its rule-based structure mimics the planning logic of a decision tree. Path overlays in several simulated scenarios were used to visualize the system's performance.

The A* (A-star) algorithm is a famous algorithm in the realms of robot path planning and graph search. It is deemed the fastest algorithm for determining the shortest path between two nodes in an environment.

The DM-SPP-4 algorithm is faster than techniques like A* and Dijkstra because it uses a contingency matrix instead of relying solely on heuristics, making it more efficient in certain cases .

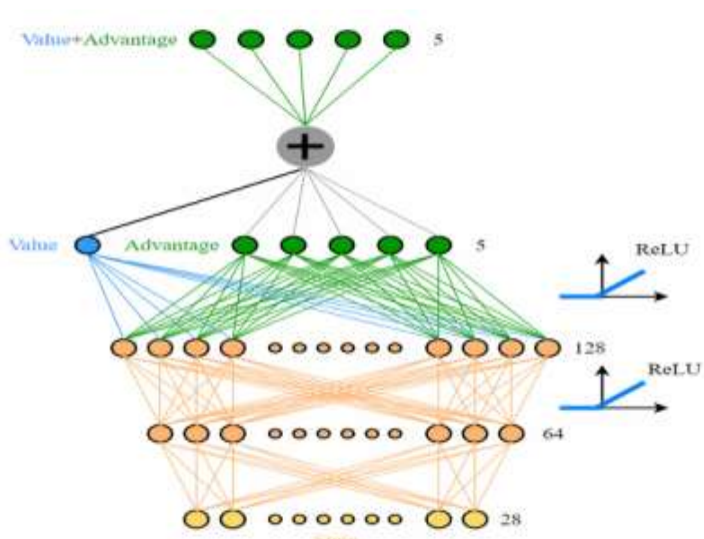

Figure 6: DM-SPP-4 path planning result on an easy map [12].

2.2. Comparative Analysis of Algorithms

When applied to autonomous robotic systems, each algorithm's unique capabilities are shown throughout the examined literature. According to Cai et al. (2021), decision trees (DT) provide deterministic control and transparency that are beneficial for rule-based robotic manipulation. Francis et al. (2022) showed that Random Forests (RF) and Support Vector Machines (SVM) are efficient in handling multi-source contexts and sensor uncertainty. Particularly, SVMs are renowned for their accuracy in applications involving binary classification.

ANN, on the other hand, is more versatile and performs better in learning-based navigation (Palanivel & Muthulakshmi, 2024). Both Francis and Palanivel's work emphasizes the complementary nature of ANN and SVM, which points to the benefits of hybrid systems that combine robust perception (SVM) and real-time adaptation ANN. This emerging direction is key to enabling robots to operate effectively in uncertain and dynamic environments.

Table 1: A comparative of related studies

Study	Algorithms	Dataset Type	Key Features	Performance Metric	Main Contribution
Cai et al.[2021]	Decision Tree (DT)	Robotic arm sensor data	Joint angle estimation, movement modeling	-	Used D-H model with DT logic for manipulator control
Francis et al. [2022]	Random Forest (RF), SVM	Gas sensor - localization logs	Sensor fusion, classification, probabilistic mapping	Accuracy: 91%	Applied ML models to enhance gas localization in multi-source environments
Palanivel _ Muthulakshmi. [2024]	ANN	Q-learning simulation data	Quantum-enhanced ANN for decision-making	Accuracy: 95%	Used QNN for fast navigation in complex environments
Francis _Palanivel	ANN vs. SVM	Comparative (literature-based)	Policy learning vs. binary classification	F1-score/Precision	Highlighted the complementarity between ANN adaptability and SVM precision

3. Methodology

3.1. Overview

The purpose of this work is to develop a system that allows robots to make decisions on their own in dynamic and complex environments with minimal human help. According to the approach discussed, the system is designed by using its sensory data to process and decide on reactions transparently. Machine learning models are used to help the decision process improve and adapt as time goes on. Autonomous decision-making has been crafted to incorporate sensor data with machine learning mechanisms to influence the robot's actions. The robot obtains environmental information through diverse sensors (e.g., proximity sensors, temperature sensors, battery level sensors, and so on). Machine learning algorithms analyze it in real time to make decisions to fulfill certain predefined mission objectives, like rescue operations, exploration, transportation, etc. [13].

3.2. Machine Learning Techniques

The heart of our system relies on three key machine-learning models: ANN, SVM, and DT. They select the model due to its success in decision problems, high ability to cope with dynamics, and an uncertain environment. Following is an introduction to each of them:

3.2.1. Artificial Neural Networks

Artificial Neural Networks (ANNs) are a family of machine learning algorithms inspired by the structure and functioning of the human brain to identify patterns and make predictions. At the core of the ANNs are neurons that are organized in layers. Each neuron takes the input, does some weight multiplication, passes through an activation function, and emits the result.

ANNs are considered a state-of-the-art computer-aided solution to very complex types of information processing, having multi-layer networks and activities of a human-like mind in the direction of pattern matching and information extraction. The mathematical equation defining the output of neurons describes how inputs from different sensors, like images, sound, and the environment, are passed on to give very specific response cases. In this equation $x_i$represents the sensory inputs, while the values of the weights w_i Identify the relative importance of affecting the final output caused by the input. The bias term b Adjusts and improves the model output. The activation function f Implements converting the weighted inputs into actual usable outputs. This kind of strategy in robotics improves the operational accuracy of the robots to some extent in making decisions involving autonomous navigation or interactive decisions with the environment. Further, robots are made to be more flexible and intelligent to rapidly respond to changing conditions; this goes a long way in helping the robot's ability to adapt to different environments [14]

The output y of a neuron is defined as:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \quad (1)$$

where:

- x_i Represents the sensory inputs (e.g., data from sensors such as images, sounds, or environmental readings).

- w_i : The weights applied to each input are learned during the training phase.

- b Is the bias term adjusting the activation function for better adaptability?

- f Is the activation function (commonly Sigmoid, ReLU, or Tanh) which introduces non-linearity into the model and helps in solving complex decision-making tasks.

ANNs are particularly well-suited for autonomous systems due to their ability to process unstructured data and improve decision-making over time through training.

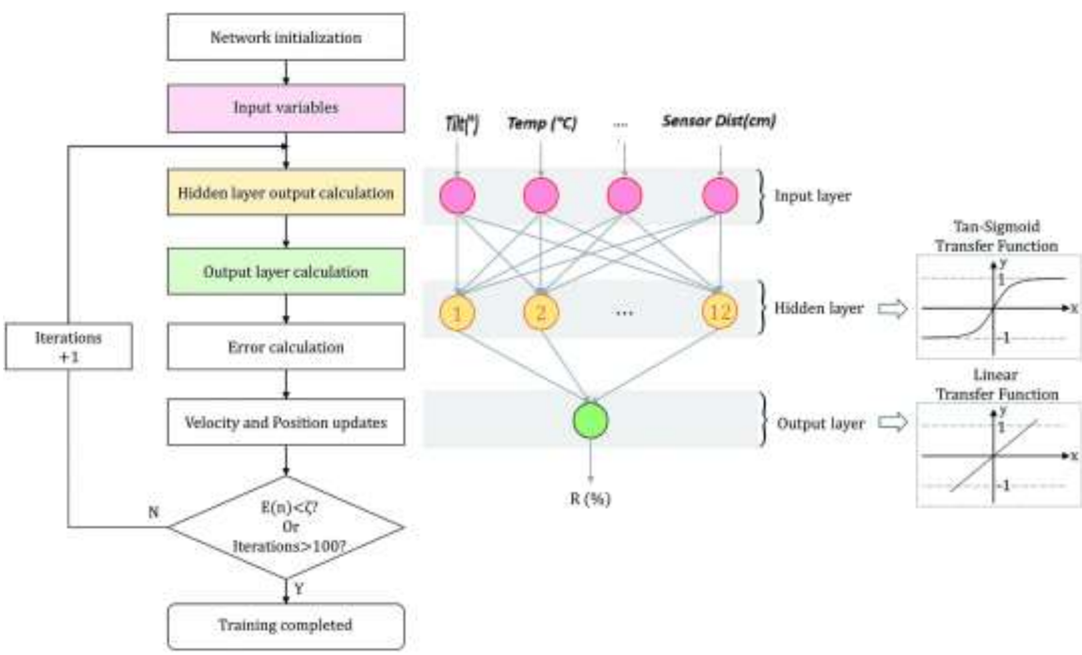

Figure 7: The optimal ANN structure and a flow chart of the training process

The ANN mechanism used in robotic decision-making is illustrated in the diagram. The activity begins with some input variables, i.e., tilt angle, temperature, and sensor distance, followed by moving through a hidden layer with the Tan-Sigmoid activation function, which helps identify patterns in processing it. The final choice is made through the output layer, which uses a linear activation function. The network then goes through a training session in iterations with error computation and update, and can finally be used in making real-time decisions when a threshold acceptable error level is achieved or the maximum number of iterations is completed.

3.2.2. Support Vector Machines

Support Vector Machines are supervised learning algorithms designed primarily for robust classification and regression tasks. The primary objective of an SVM is to establish an optimal hyperplane that distinctly separates data points of different classes, maximizing the margin between the closest data points (support vectors).

SVM can be expanded in nonlinear classification with the help of the kernel trick, which maps the data into a larger space to help identify a hyperplane for division. Among its key strengths, SVM can solve complex issues since it can separate them into a new format.

SVM supports a lot of different applications like tagging texts, identifying patterns, and catching outliers. It is strong in handling huge amounts of information, which makes it the first choice for topics like computer vision and the analysis of medical data. Having many data points or messy data may make SVM have issues, but it also works very well with different kernel functions [3].

The mathematical representation is:

$$\mathbf{w}^T\mathbf{x} + b = 0 \qquad (2)$$

Where:

- w : is the weight vector.

- $\mathbf{x}$: feature vector representing environmental states.

- b : is a coefficient that indicates the constant part of the model.

We try to increase the distance between the two sets of lines by calculating:

$$\rho = \frac{2}{||w||} \qquad (3)$$

Where $||w||$ Is the norm of the weight vector w. This equation indicates the distance between the decision boundary and the closest data point from each class.

The classification rule follows:

$$y = \text{sign}(\mathbf{w}^T\mathbf{x} + b) \qquad (4)$$

Where:

y is the class label (+1 or 1)

SVM was selected because of its strict mathematical foundations, robustness toward noisy data, and high efficiency in defining boundaries among classes, even in complex non-linear arrangements. The SVM technique seemed to be an appropriate choice.

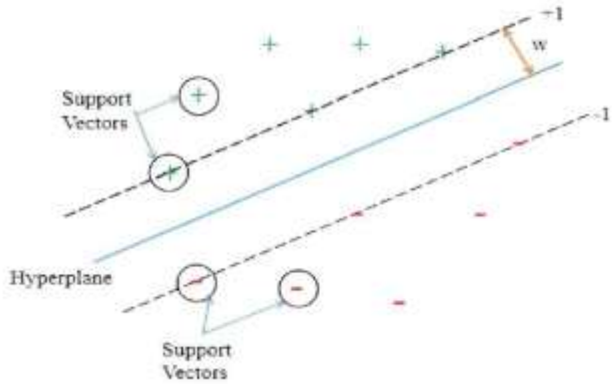

Figure 8: Schematic of the SVM.

In Figure 8, the SVM concept is broadly explained. A hyperplane separates the input space representing two classes, +1 and -1. The hyperplane could either be a line (in two dimensions) or a surface (in higher-dimensional spaces) according to the dimensionality of the input space. The points near the hyperplane defining its optimal position and orientation are called support vectors. The margin (the distance to the nearest support vectors) is thereby maximized by minimizing the norm of the weight vector (||w||). This leads to better classification accuracy and, therefore, better generalization of the model.

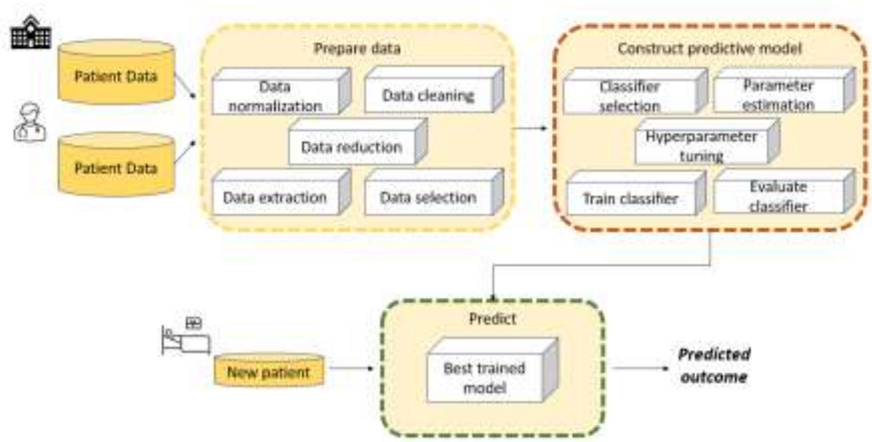

Figure 9: The typical workflow for developing a ML model using SVM for classification or regression tasks.

First comes the preprocessing of patient data, which includes normalization, cleaning, extraction, reduction, and feature selection. This is then followed by the creation of a predictive model by the selection of an appropriate classifier and estimation of parameters, hyperparameter tuning, training of the classifier, and evaluation of its performance. The final model is utilized for the new patient data to predict the outcomes. SVM was originally developed for binary classification based on statistical learning theory but has subsequently and successfully been reformulated for use in many classification problems. The present diagram indicates some of the main phases involved, particularly data preprocessing and careful model training, hyperparameter optimization, and robust evaluation, which constitute an exhaustive and reliable predictive capability in diagnosis and regression.

3.2.3. Decision Trees

Decision Trees provide clear and interpretable mechanisms in decision-making, a requirement for cases where actions have to be justified for the decision. Each tree node refers to a decision that can be based on certain sensor data, and the leaf nodes correlate with the best actions that can be performed, such as moving ahead or turning left.[18] The decision trees are mainly grown by Information Gain (IG) criteria calculated through:

$$IG(D,A) = Entropy(D) - \sum \frac{|D_v|}{|D|} \times Entropy(D_v) \quad (5)$$

Where:
- $Entropy(D)$: a measure of dataset uncertainty.
- D_v a subset of data for each attribute value v.

In this framework, the decision tree analyzes the sensory data from the surrounding environment and generates an optimal path for the robot. In this sense, the decision tree enhances the understanding of how the robot would seek an effective solution to the gathered sensory data, which can include temperature, distance, or battery level. Next, it helps determine how the robot should act, say, by moving forward or stopping. It is based on optimal features for splitting the data, using Information Gain and Gini Index, etc., to ensure the data is accurately distributed for optimum separation of categories. These parameters are established by equations on how to compute different variance indicators from different categories concerning searching for the best data splitting points.

Thus, the decision tree starts with the root node of the data, which will then be decided by taking a specific feature and dividing the data into sub-groups. Then, the procedure takes care of the whole internal nodes, where further decisions would be made based on the rest of the features until it reaches the leaf nodes, where the final classification or action will be picked from. For autonomous robotic systems, a decision tree is a perfect model where one needs to make decisions based on sensory data easy to interpret.

One of the widely used techniques in data mining is systems that create classifiers. In data mining, classification algorithms are capable of handling a vast volume of information. It can be used to make assumptions regarding categorical class names, to classify knowledge based on training sets and class labels, and to classify newly obtainable data. Classification algorithms in machine learning contain several algorithms, and in this work, the paper focused on the decision tree algorithm in general. Fig.10 illustrates the structure of DT.

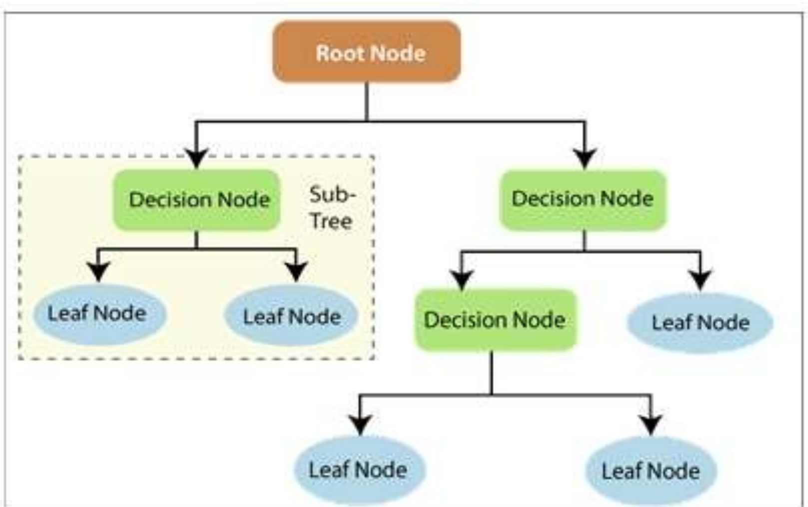

Figure 10: Decision Tree Flowchart

Decision trees are one of the powerful methods commonly used in various fields, such as machine learning, image processing, and the identification of patterns. DT is a successive model that unites a series of basic tests efficiently and cohesively, where a numeric feature is compared to a threshold value in each test. Figure 11 presents an applied example of decision tree usage for predicting mission outcomes for robots based on environmental sensor inputs, almost in real time. At the root node, the framework would test the most critical sensor data, whose values were compared with their threshold limits: distance measurements, battery levels, and ambient temperature. Once decisions have been made at this root node, the tree splits into decision paths, each representing different scenarios. For instance, if the battery level is below 49%, the tree considers this parameter as critical in deciding on mission feasibility and then on task selection. Distance measurements from proximity sensors less than or equal to threshold values will immediately determine subsequent navigation decisions, with choices such as obstacle avoidance or continuing forward being made. Leaf nodes would then display the designated values of mission success (+1), partial success, or mission failure (1). This interpretative decision-making capability provides a structured methodology for the robot to effectively alter its actions autonomously in dynamic and complex environments.

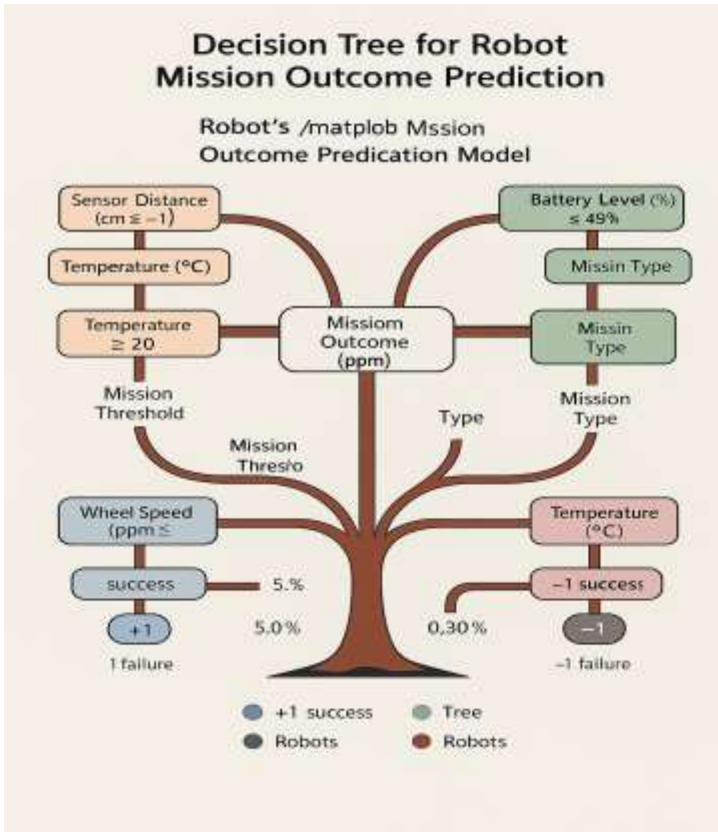

Figure 11: Decision Tree Example for Robot Mission Outcome Prediction.

3.2.4. Data Description

Based on the main data picked up by advanced sensors, the robots can direct their actions independently. It is important to base actions on the data coming from the sensors while monitoring the environment. The kind of data gathered for this purpose is often the distance between the robot and the object. Using proximity sensors, the data helps the robot decide whether to halt, avoid objects in the way, turn around, or continue going in its intended path. Temperature data is also another kind of data. The sensor in the robot checks the temperature of the surrounding environment, which reports the values in degrees Celsius. This information is very valuable when handling reprogramming and debugging if the robot gets too hot. The battery sensors give out battery percentage data in real-time, informing me about the remaining battery in the robot. It helps with sending the robot back to charge when necessary or with making energy-saving efforts during the activity. In addition, wheel speed sensors monitor the number of rotations the robot makes in minutes (RPM). Having the data from the sensor allows the robot to change its speed to fit different situations while performing a task. Information is collected live as the robot goes about its operations in a complicated environment. With this data, machine learning chooses the best response to any given state in the environment for the object.

In the preparation of the data for feeding, some preprocessing steps are involved, such as normalization, noise reduction, and feature selection. Data normalization ensures that for every feature, the values are within the same range, thereby aiding in training accuracy and computational speed; noise reduction takes off erroneous values and outliers that otherwise may be against the analysis; and feature selection gets down to only the relevant features, such as distance, temperature, and battery level, by excluding other irrelevant variables that could only interfere with the analysis. This data maintains the robot's ability to work autonomously in adapting to different environments and carrying out tasks effectively and accurately.

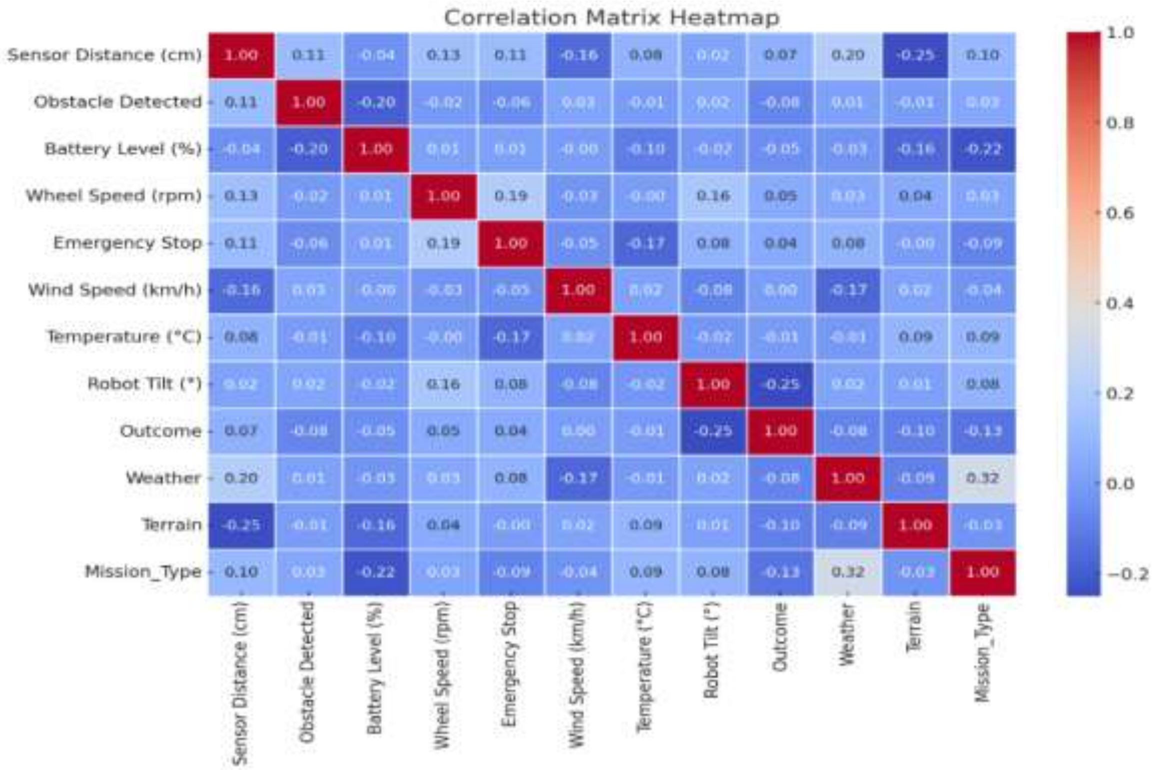

Figure 12: Correlation matrix heatmap for decision making system in robots.

3.2.5. Working structure

The process starts by using data from the robot’s sensor to observe its surroundings and changes in actions as time goes on. Preprocessing is done to enhance the quality of the data so it can be used in decision-making analysis. The process of selecting models introduces various architectures, including single models such as ANN, SVM, and DT, and hybrid models that intend to boost how robots make decisions.

When the model is chosen, it is trained on the data, considering dependencies over time and the robot’s contacts with what surrounds it. At this point, validation data is used to enhance the training and make the model perform better.

The optimization process starts with the choice of an optimization method based on the properties of the robot environment and its decision-making tasks. After training, the models are fully tested on the testing dataset to measure model performance in real-world scenarios.

Finally, a decision point is attained through a comparison of the performance of the models, choosing the one with the highest performance to ensure effective sound decision-making in robotics. Figure 13 shows the interdependent processes and decision-making routines that constitute the robotic decision-making system, based on sensor data and robot interaction with the environment.

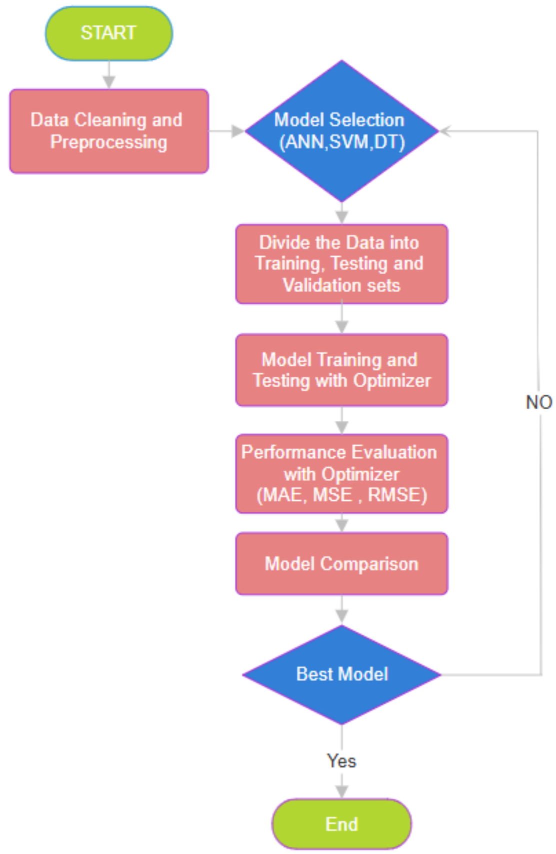

Figure 13: Workflow and model selection flowchart.

4. Result Discussion

4.1. Data Description

We used this carefully engineered and curated dataset with various AI-driven classifiers to model and predict robotic mission outcomes. The dataset consists of 500 observations, of which 250 each represent samples of successful and failed missions. This helps maintain class balance and avoids biased training. Each record contains both intrinsic robotic sensory data and extrinsic environmental information. This dual perspective allowed us to look concretely at modeling decision-making behavior under uncertainty.

Included features thus ranged from continuous sensory signals, such as Sensor Distance (cm), Battery Level (%), Wheel Speed (rpm), Wind Speed (km/h), Temperature (°C), and Robot Tilt (°), to higher-level categorical variables such as Terrain Type and Mission Type. Terrain Type was encoded as a single feature with three values (0 = Rough, 1 = Flat, 2 = Sandy) and Mission Type with three values (0 = Rescue, 1 = Delivery, 2 = Exploration). The continuous variables were standardized with a standard scaler to keep the features comparable and to facilitate convergence during training.

The data were preprocessed to remove outliers and noise and processed for missing values with domain-sensitive imputation. Such a preprocessing provides a consistent platform for machine learning model learning. The design of the dataset is representative of real-world operating conditions in robotics and includes rich mixtures of mechanical, physical, and contextual parameters. Such a structure makes it conducive to building data-driven and interpretable decision systems.

Table 2 shows how the robotic mission dataset is put together. The table shows that predictive modeling takes into account various inputs, among them Sensor Distance, Battery Level, Wheel Speed, Mission Type, and Outcome. The data was well-balanced and free of noise, so training a classifier would be accurate.

Table 2: The robotic mission dataset description

Attribute	Type	Example Values	Categorical Variable
Sensor Distance (cm)	Continuous	-1	Terrain Type
Battery Level (%)	Continuous	90	
Wheel Speed (rpm)	Continuous	120	
Wind Speed (km/h)	Continuous	8.5	Mission Type
Temperature (°C)	Continuous	23.4	
Robot Tilt (°)	Continuous	-10	
Terrain Type	Continuous	Delivery	Delivery
Mission Type	Continuous	Flat	Delivery

4.2. Experimental Setup

The experimental environment attempted to quantify and compare the three most important classification models' predictability, i.e., Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Decision Trees (DT). The formatted data were first divided into training (80%) and test (20%) subsets randomly for model generalizability. A preprocessing pipeline with feature scaling and categorical encoding was applied uniformly across all models to be fair.

The ANN model architecture consisted of three hidden layers with 128, 64, and 32 neurons, respectively, each with ReLU activation. Early stopping conditions, along with a maximum of 3000 epochs of training, were implemented. The optimizer used was the Adam optimizer, and the cross-entropy loss function was used, with batch normalization being used to make the learning stable. This arrangement was used owing to its effectiveness in high-dimensional nonlinear classification problems.

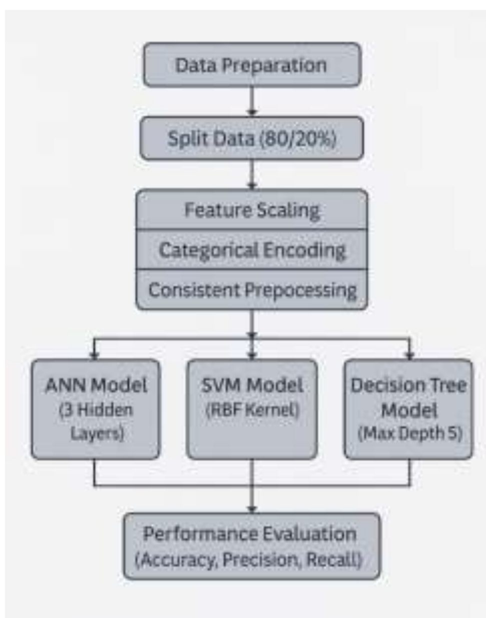

Figure 14: Workflow of the experimental setup detailing the processing steps, Model architectures, and evaluation Strategy for robotic mission classification.

The SVM model utilized a Radial Basis Function (RBF) kernel with regularization parameters C = 1.0 and gamma = 'scale'. SVM is known for its robustness in small to medium-sized datasets and its ability to find optimal separating hyperplanes between classes, particularly in non-linear settings.

The Decision Tree classifier was implemented with a maximum depth of 5 to avoid overfitting and to provide interpretability. The Gini impurity metric was used to split nodes, and the model emphasized clarity of rule extraction over accuracy.

All models were trained and tested on the same hardware and software environment to ensure comparability. Performance metrics were evaluated using accuracy, precision, recall, F1-score, and confusion matrices.

4.3. Numerical Results and Analysis

When trained and tested, the Artificial Neural Network model was far superior to the rest of the classifiers. The model generated a classification accuracy of 99% for the test set. The ANN model was not only superior in overall accuracy but also in terms of precision-recall balance. For the success class, the ANN had 100% precision and 98.5% recall, while for the failure class, it achieved 97.2% precision and 100% recall. These are extremely high values and show that the ANN model successfully picked up linear as well as non-linear patterns in the feature space. The Support Vector Machine model also achieved 93% accuracy. Its precision and recall values were more than 90% in both classes.

The SVM model illustrated a clear ability to separate operating patterns from environmental factors that were related to success or failure. It is most appropriate for boundary-based learning, and hence it is perfect for high-risk robotic tasks where the reduction of false positives is paramount.

The Decision Tree model, while being less accurate overall at 76%, provided useful interpretability. It was observed that the DT classifier overpredicted the failure class with good recall (88.6%) but poor precision (60.8%). This is common in models tuned for binary thresholds without regularization.

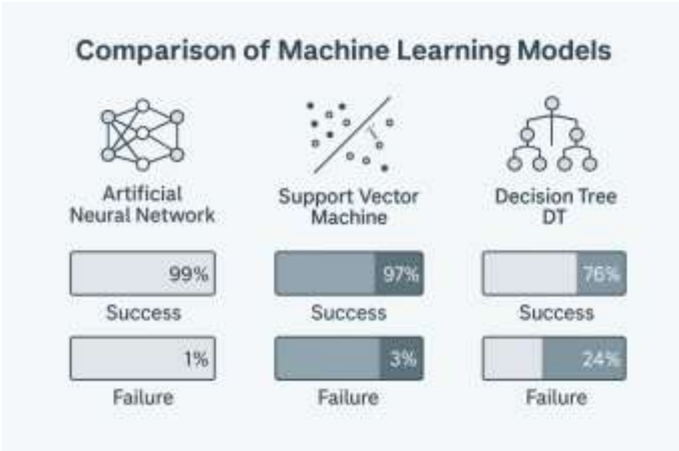

Figure 15: A comparison of the ml models' performance.

These results tell or affirm, surely, the supremacy of neural architectures like ANN in capturing complex robot-environment interactions. Despite SVM standing fine at the balance point between accuracy and computational cost, the DT remains suitable for purposes of rule-based inference and for depicting decision logic visually. Also, the comparison points to the utmost importance of proper feature engineering and scaling in achieving good classification.

4.4. Comparison Between Best and Worst Models

It was found that ANN achieved better results than SVM and DT. By using complex architecture, the Rover could detect how important things, such as the Rover's speed, the state of the battery, and the terrain, influenced the mission. The ability of the model to generalize is clear from the fact that it can nearly perfectly classify missions as either successful or unsuccessful.

However, the outcome of this work suggests that the Decision Tree performed the least successfully. Due to dividing the tree only once and not much, it could not find more complex patterns in the data. However, since it is easy to understand and safety systems, readability matters more than the ability to make predictions.

The SVM model provided a balanced prediction that was effective and interpretable. Compared to ANN, SVM does not require much tuning or excessive computational power. It also worked well using the normalized feature set and exhibited resistance to overfitting. SVM's kernel trick enabled it to project data into higher dimensions where data separation was simpler.

In general, the ANN model ranked top among the three to employ in real-time robot mission decisions due to its high accuracy as well as the ability to learn. However, where quicker inference or explainable reasoning is essential, the SVM or DT models may remain practically applicable, dependent on system constraints.

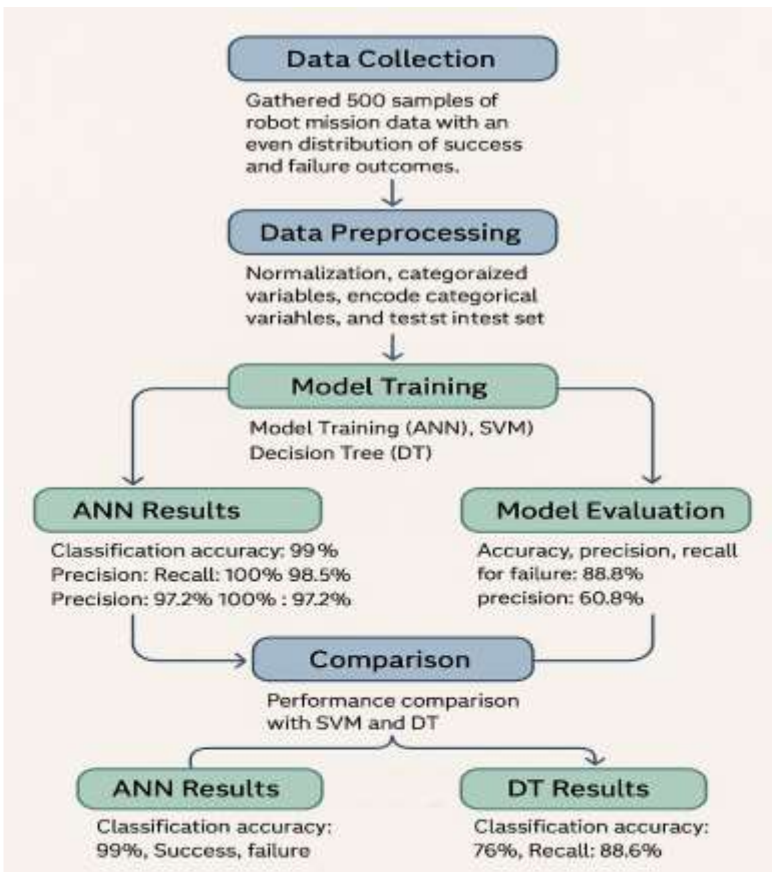

Figure 17: Grayscale workflow illustrating the complete classification pipeline from data preparation to model comparison in robotic mission outcome prediction.

5. Conclusion

Decision-making systems in robotics constitute the very essence of providing autonomy and flexibility in the arena of complex and dynamic settings. The study has, therefore, expanded upon its basic scope to cover the design and development of an intelligent model based on higher AI techniques, namely ANN, SVM, and DT, to enable robotic systems to make decisions with multiple data points and changing environmental conditions taken into consideration. The simulation results indicated that the ANN model showed excellent accuracy (99%), indicating its excellent ability to manage non-linear and complex data, while the SVM and DT models were noted for providing logically sound and transparent decision explanations, hence contributing to a compromise between performance and precision.

On the other hand, despite the clear superiority of the ANN model, many areas are ripe for hybrid artificial intelligence techniques, where the learning strength of ANN could be coupled with the classification strengths of SVM and the transparency of DT to pave the way for future applications in critical areas such as disaster response and recovery, hazardous material handling, and medical robotics.

This study is a critical step toward developing more autonomous and adaptive robots capable of operating in diverse environments. Yet, there remains a big hurdle to deploying such systems in real-time at a large level, especially in high-stakes environments. Therefore, the study proposes continued exploration of hybrid models and their deployment in real-world scenarios with an emphasis on the continued learning and self-improvement of robots from real-world experiences. This research also opens up avenues for more studies on artificial intelligence applications in industrial robotics, health, and security systems, resulting in the development of new solutions for complex societal problems.

Acknowledgment

Above, this research could not have been realized without the definitely instrumental support and invaluable guidance of several persons and institutions. For the broad areas of knowledge, timely mentorship, and husbanding criticism applied toward steering and molding this work in terms of content, profundity, and quality, I am deeply indebted to my supervisors.

Being a student of Robotics Engineering at Alzaytona University (ZUST), I sincerely thank the Department of Control and Robotics Engineering for giving me the best of facilities, access to a few specialized datasets, and a very intellectually stimulating research environment, through which practically this study could be realized.

I would also like to mention the global scientific society on artificial intelligence, machine learning, and autonomous robotics- they stand as the bedrock: the research, development, and implementation of decision-making systems in robotics have been heavily based on theirs.

Finally, I am extremely grateful to my family for their undeterred encouragement, patience, and sacrifices that enabled me to face all challenges and withstand the way through this entire research process.

References

[1] T. Mota, M. Sridharan, and A. Leonardis, "Integrated Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics," *SN Comput. Sci.*, vol. 2, no. 4, pp. 1–18, 2021.

[2] F. Logic, "Artificial Intelligence Syllabus – Spring DRAf," *Artif. Intell.*, pp. 1–2, 2007.

[3] D. Salman and M. Kusaf, "Short-term unit commitment by using machine learning to cover the uncertainty of wind power forecasting," *Sustain.*, vol. 13, no. 24, 2021.

[4] M. K. Ho and T. L. Griffiths, "Cognitive Science as a Source of Forward and Inverse Models of Human Decisions for Robotics and Control," *Annu. Rev. Control. Robot. Auton. Syst.*, vol. 5, pp. 33–53, 2022.

[5] D. Salman, C. Direkoglu, M. Kusaf, and M. Fahrioglu, "Hybrid deep learning models for time series forecasting of solar power," *Neural Comput. Appl.*, vol. 36, no. 16, pp. 9095–9112, 2024.

[6] D. Salman, Y. K. Elmi, A. S. Mohamed, and Y. H. Mohamed, "Forecasting Maximum Power Point in Solar Panels Using CNN-GRU," *SSRG Int. J. Electr. Electron. Eng.*, vol. 11, no. 7, pp. 215–227, 2024.

[7] D. Salman, C. Direkoglu, N. Altanneh, and A. Ahmed, "Hybrid Wavelet-LSTM-Transformer Model for Fault Forecasting in Power Grids," *SSRG Int. J. Electr. Electron. Eng.*, vol. 11, no. 12, pp. 314–326, 2024.

[8] "M. Francis, S. Kumar, and J. Lee, 'Machine learning techniques for gas sludge analysis and localization using mobile robots: A comparative study of ANN and SVM,' Journal of Environmental Robotics, vol. 15, no. 3, pp. 245–261, 2022."

[9] D. Salman, "Optimal Power Systems Planning for IEEE-14 Bus Test System Application," pp. 290–295, 2022.

[10] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, 2016.

[11] S. Samantaray and A. Sahoo, "Groundwater level prediction using an improved ELM model integrated with hybrid particle swarm optimisation and grey wolf optimisation," *Groundw. Sustain. Dev.*, vol. 26, no. November 2023, p. 101178, 2024.

[12] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Frcitas, "Dueling Network Architectures for Deep Reinforcement Learning," *33rd Int. Conf. Mach. Learn. ICML 2016*, vol. 4, no. 9, pp. 2939–2947, 2016.

[13] D. Salman, A. A. Siyad, M. Kusaf, and Y. Elmi, "Day Ahead Unit Commitment with High Penetration of Renewable Energy Sources and Electric Vehicle Charging Stations," vol. 72, no. 6, pp. 361–379, 2024.

[14] D. Salman, A. F. Ali, S. A. Ali, and A. S. Mohamed, "Enhancing Power Grid Stability through Reactive Power Demand Forecasting Using Deep Learning," *SSRG Int. J. Electr. Electron. Eng.*, vol. 11, no. 12, pp. 170–185, 2024.