Learning When a Robot Should Ask for Help: A Decision Tree Approach

Your Name

June 28, 2025

Abstract

This project examines how a robot can learn to determine when to request assistance based on situational features. We trained decision tree classifiers using various-sized subsets of a 100-example dataset containing binary attributes such as familiarity, urgency, and safety. As described in the assignment objectives, our goal was to assess how accuracy improves with training size. Trees were trained on subsets of 2, 5, 10, 20, and 50 examples, and tested on the remaining data. We found that accuracy generally increases with more training data, confirming the value of increased sample sizes to enhance training for autonomous agents. We visualized the learned trees and examined how the decision logic evolves with increasing training size.

1 Learning Algorithm

We used the decision tree classifier from `scikit-learn`, a supervised learning model that recursively partitions feature space based on impurity metrics. In this case, we used the Gini impurity to determine optimal splits. Each internal node of the tree represents a decision based on one of the binary input features:

- Familiar Situation
- In a Hurry
- Long Delay
- Safe Situation
- Confident
- Safety Critical Decision
- Asked Before

The classifier was trained on increasingly large subsets of the dataset, and its accuracy was tested on the remaining data. Each decision tree was visualized and analyzed to observe how its structure changed with training size.

2 Results

Figure 1 illustrates the accuracy of the decision tree classifier on test data. Based on the number of samples, the data suggests that fewer samples do not benefit the decision tree model as there is not enough information for it to learn from. However, as the sample size increased, the accuracy of the decision tree outcomes also increased. This factor enables the robot to have a significantly increased chance (94% based on 50 samples vs 59% for the 2 samples) of making the correct decision given its current state. NOTE: The following figures are provided with the source code submission.

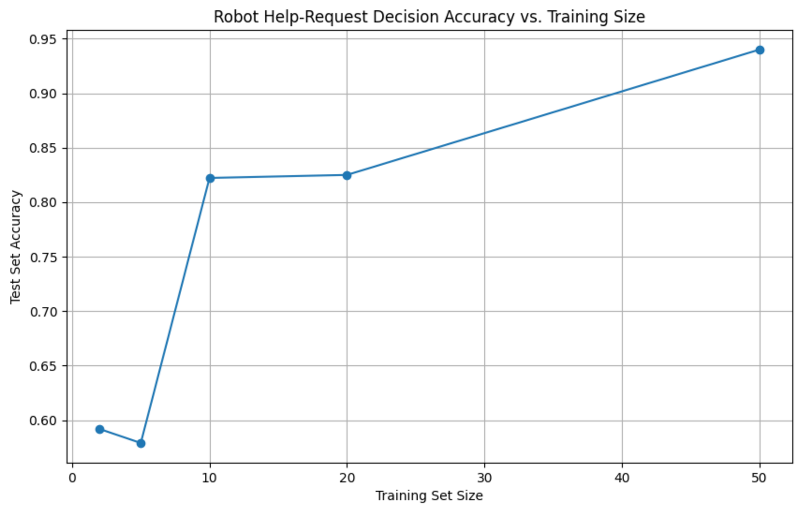

Figure 1: Test accuracy vs. training set size.

Figures 2 to 4 illustrate the decision trees trained on each subset size. As training size increases, the tree becomes more complex and better reflects generalizable decision boundaries.

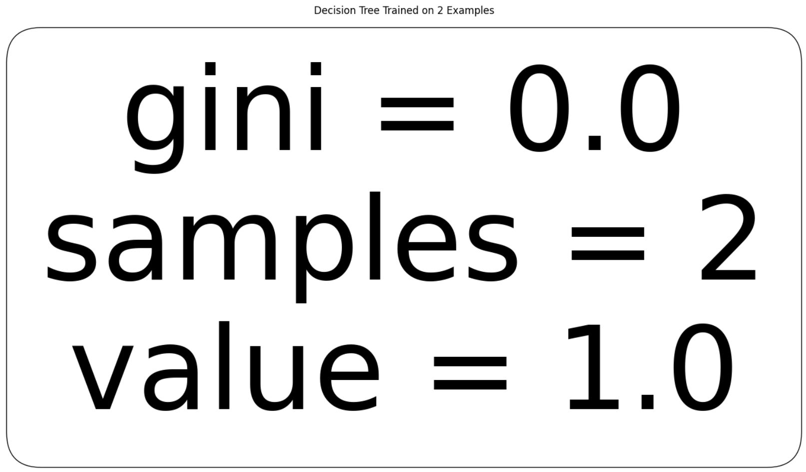

Figure 2: Decision tree trained on 2 examples.

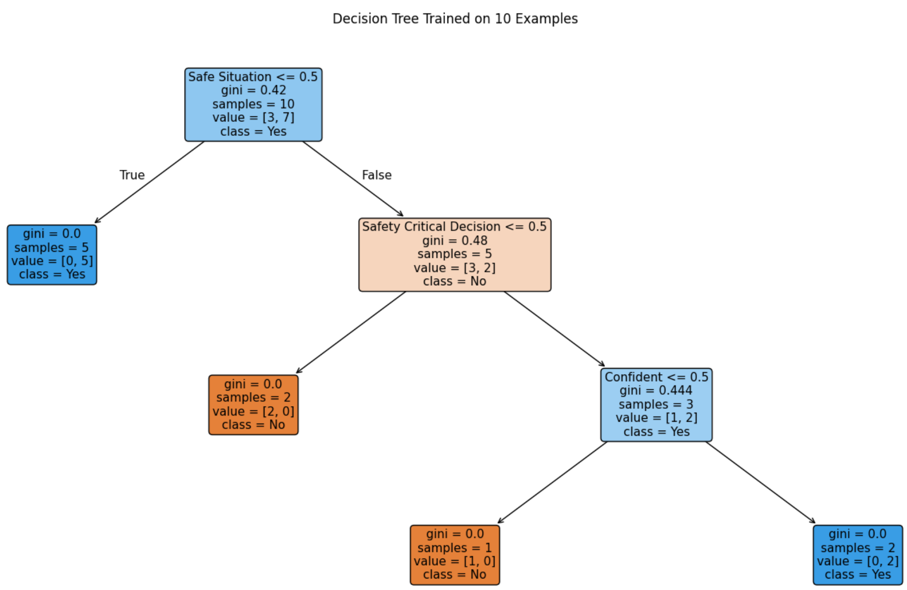

Figure 3: Decision tree trained on 10 examples.

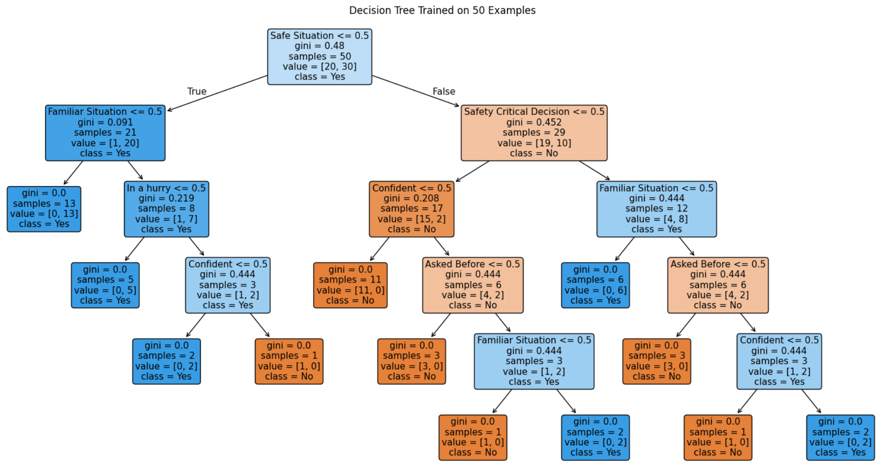

Figure 4: Decision tree trained on 50 examples.

3 AI Tools Used

The following AI tools and libraries were used:

- **ChatGPT**: Used for debugging help and figure generation.
- **Python (NumPy, Pandas)**: For data preprocessing and handling.
- **scikit-learn**: For building and evaluating decision tree models.
- **Matplotlib**: For plotting accuracy graphs and visualizing decision trees.

These tools accelerated development, provided explanations for algorithm behavior, and helped visualize complex results in a human-readable format.

4 Discussion/Conclusions

This project demonstrated that decision trees can effectively model when a robot should request help. The experiments show that even with small training sets, decision trees can generalize meaningful decision-making logic. However, small trees can be subject to bias and often lead down a branch that could result in poor outcomes; accuracy and structure improve significantly with larger datasets. The 50-sample decision tree was hard to visualize legibly for this paper due to its depth. What was observed was that if the robot was in a safe situation, it was most likely familiar with that state, in a hurry, and confident, the robot would most likely still ask for some help. If it was not in a safe situation, and it led to a safety-critical but familiar situation, it was more likely that it would not ask for help.