The Halt to the Halting Problem: Utilizing Artificial Intelligence to Detect the Halting Behavior of Functions

CS-04

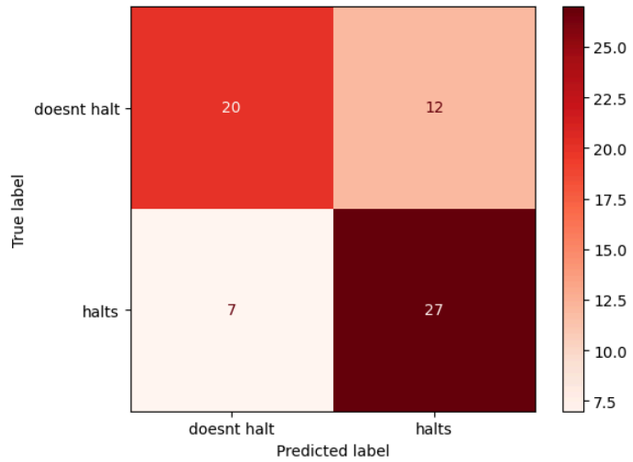
James Mathis, ASMSA, Hot Springs, AR, US

Scientific Question OR Engineering Problem

- Are there patterns in halting and non-halting functions?
- Can a neural network effectively observe them?
- Can a neural network correctly predict the halting behavior of functions at least 60% of the time?

Data Analysis & Results

- Accuracy
  - Mean: 62.0%, Standard Deviation (SD): 4.15%
- Loss
  - Mean: 0.802
  - SD: 0.0541
- Precision
  - Mean: 0.653
  - SD: 0.0493
- Recall
  - Mean: 0.566
  - SD: 0.0622



Confusion matrix for best run. Made by researcher.

Methodology OR Project Design

- Manual creation of 15 Python functions.
- Crowdsourcing via a website, yielding 5 functions.
- Script-generated random Python functions (200 total).
- Determined halting behavior through multithreading.
- Neural network with three layers: Embedding, flattening, dense
- Hyperparameter optimization for model optimizer, epochs, batch size, output dimensions.
- Training on 150 samples, evaluation on 50 samples for 100 trials.
- Recorded key metrics: Accuracy, loss, precision, recall.
- Analysis of metrics, implications, and proposed improvement (Different discriminator, Random Forest).

Interpretation & Conclusions

- Mean accuracy 62.0%, target 60%.
- Neural network success indicates observable patterns.
- Standard deviation 4.15% suggests reliable accuracy.
- Model needs 2 or 3 trials for trustworthy predictions.
- Implications for AI in solving Turing's Halting Problem and programming debugging tools.