



The use of machine learning in AI general intelligence



Tobias A. Dean, tadean21@earlham.edu

The Idea

Abstract

This research project explored converting images of Tetris into abstract representations digestible by AI.

Machine learning techniques and image manipulation libraries are used to extract data from the UI.

That data is then organized into one standard representation for AI use.

Artificial General Intelligence

Artificial General Intelligence is the concept of an AI that displays intelligence and adaptability on par with a human.

This adaptability requires an AI able to encounter new data and relate it to prior knowledge.

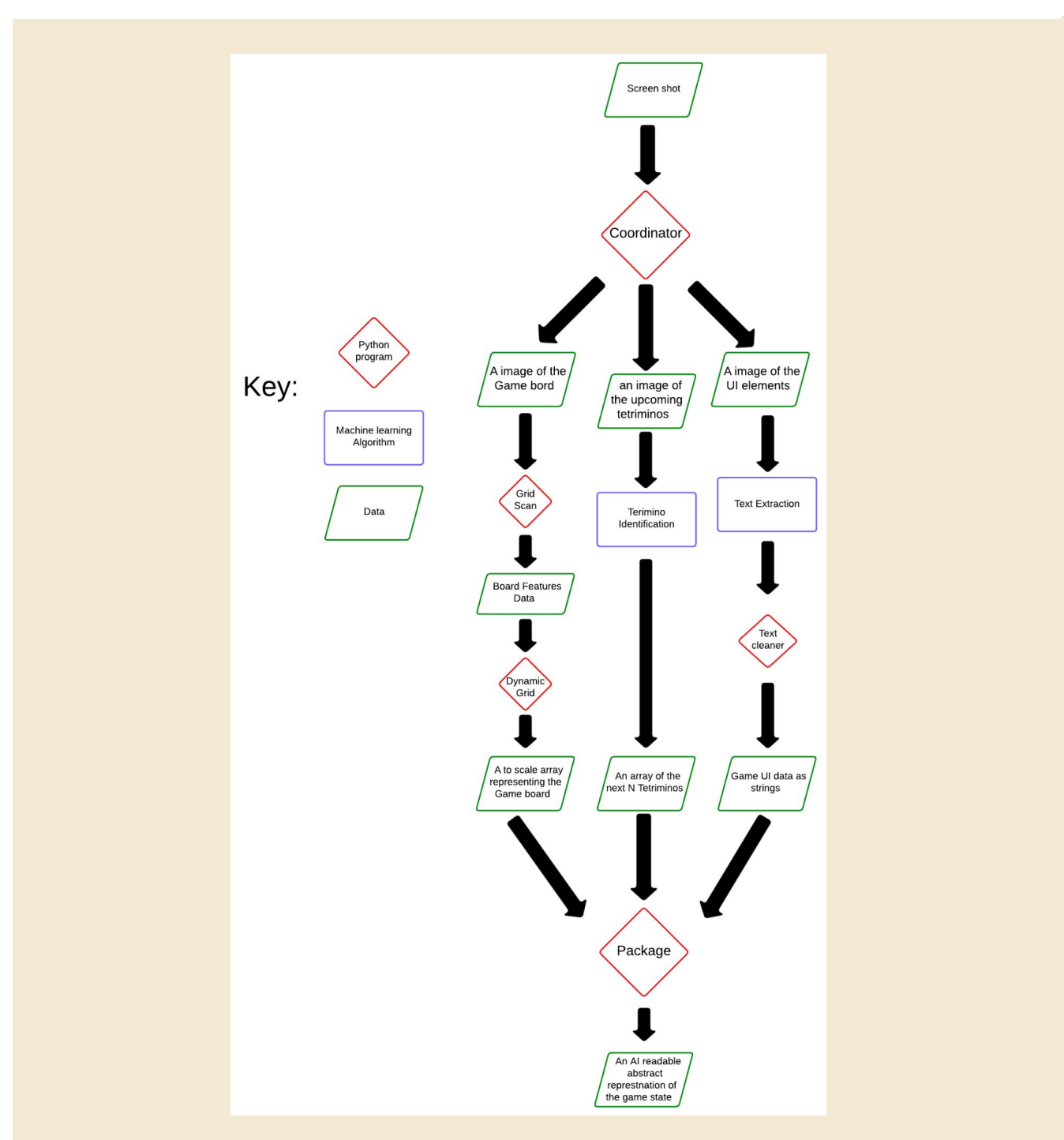
Thus, a way to reliably extract data from unfamiliar UIs important step towards AGI. This project aims to begin that step with the Tetris UI.

Why Tetris

In summary, Tetris is an old, well-known, and simple game, ideal for testing.

For these reasons, Tetris has a long history of use in testing machine learning, a history I hope to contribute to.

Early Software Architecture



The software architecture at the beginning of the development stage.

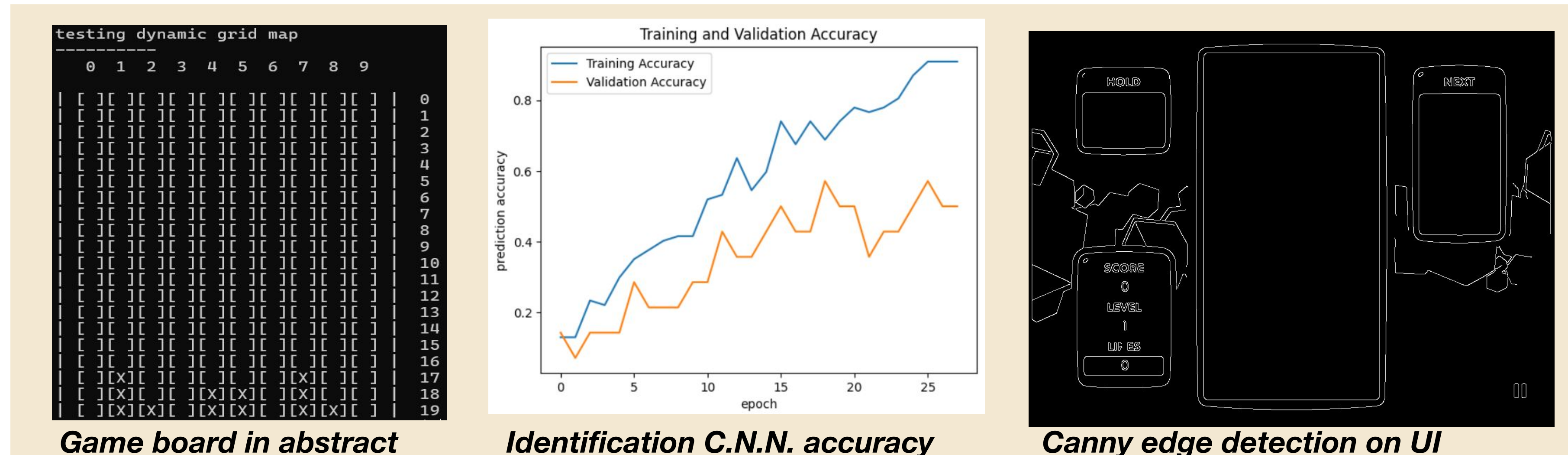
The Process

Machine learning tech

The was initially proposed with one algorithm to extract all the data in mind but it quickly became apparent that this would be insufficient. The first neural network, used to classify tetrominoes, was written with Keras, an open source Python library in TensorFlow.

ImageMagick was an invaluable tool for this project. Both for data augmentation and clearing the background of tetrominoes taken from gameplay images.

Early Iterations and results



The Convolutional Neural Network, CNN, was trained to identify unplaced game pieces. The Canny edge detection provide by ImageMagick outlined and isolated the edges of the most prominent UI features.

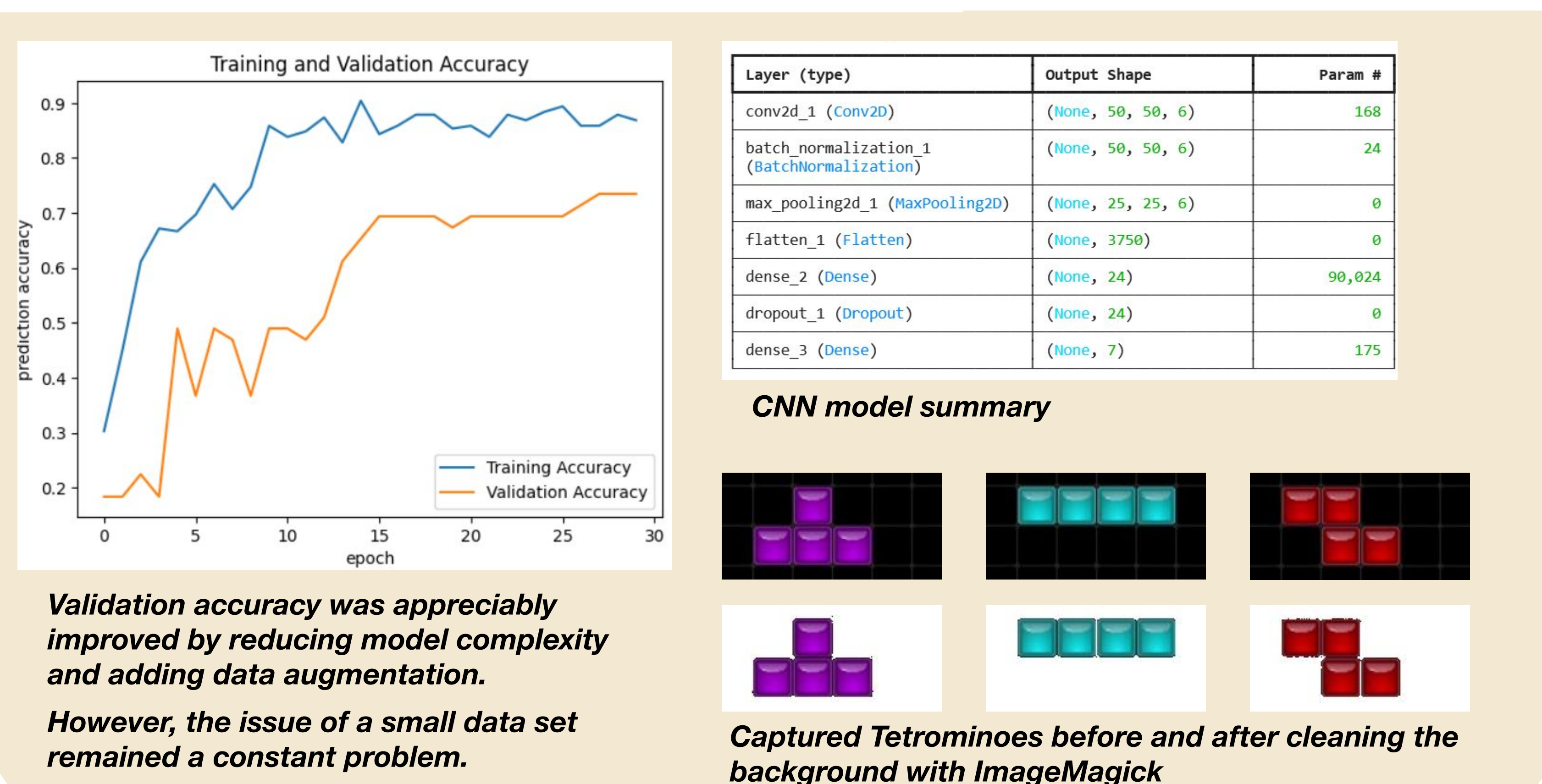
Complications Encountered

The most frequent source of complications was the nicht nature of the problem.

A wide variety of resources already exist for machine learning projects, but few of them were designed with such shallow problems in mind.

Similarly, I could find no existing data sets for the Tetris game and UI components.

Later Convolutional Network Iterations



The Results

Final Outputs



Comparing several frames to their abstract representation a pattern emerges.

The grid had no problem tracking the moves it and updating the board.

However, the distortion of the cleaned tetrominos significantly reduced Identify accuracy.

Conclusions

Although a the results show promise, there is still a long way to go.

The underlying ideas seem sound, but the process is to vulnerable to noise.

This vulnerability must be addressed for this project to bear fruit.

Future work

Pairing this it with an AI would allow for more conclusive testing of how digestible the produced abstracts are.

However, Before that the issues with prediction accuracy must be addressed.

The first step to this would simply be expanding the data set used in training.