Data/Input from Teacher via GUI (initially only text input and then working towards other modalities including audio and visual media)

Parse-In Module: Classification of the input – data is parsed using through different 'computational seives' to categorise and record the data and where the input came from.

Core module: monitors all behaviour taking place in the program – and simultaneously analyses this information to allow the AGI to restructure itself.

Processing Module: Parsed data is then assigned to the relevant nodes on the neural network and deep learning is applied to look for patterns/match new patterns to existing patterns. (This is a constant process even

Metacognitive module: records the behaviour of the core driver adapting the algorithms using a strict evolutionary process which only selects the 'fittest' algorithms. It is the only module capable of recursion.

Parse-Out Module: Classification of the output – transformed data is parsed through 'computational sieve' to categorise responses and select an appropriate output.

Output is sent to Teacher via GUI.

Key:

A -> B – indicates flow of data from A to B.

Purple – GUI input/output

Red – analytical and synthetic reasoning for classification/organisation purposes.

Blue – analytical deep learning algorithm with overlaid Bayesian reasoning using Boolean logic.

Green – Core module which represents human episodic memory – critical for advanced learning. This module represents a constant log of all computations (including the metacognitive module) with its own deep learning algorithm, Bayesian reasoning and Boolean logic. Capable of combining/modifying algorithms.

Orange – Metacognitive module – critical for the creative process. At its heart has a random seed which allows it to mutate different adapted algorithms. Selects algorithms based on their 'fitness' and evolves solutions to knowledge problems using deep learning, Bayesian reasoning and Boolean logic as applied to the core driver and processing module either directly or through the core module. When faced with an intractable problem it employs recursion to analyse itself until it finds a possible method for solving the problem at hand which it then send to processing until a solution is found.