After viewing and running the examples from the github, it was determined that the evaluator script, while having similar outputs into open evolve, has logic that is intimate with the problem itself. Thus it is determined that any algorithm developed to be evolved has to be relative to the issue itself, which must be hand coded. It would be much more convenient if there was some sort of automatic evaluator writing script, especially since there are already many variables introduced into the system to modify performance of the output. This would extend the categories of data we are looking for. From these:

- Base Data(data of our resources prior)
- Context Amount (how much context fed to the AI affects the efficiency of the output)
- Context Quality (how much of the context fed to the AI is actually related to the problem)
- Human Format of Prompting (how humans may or may not prompt the AI)
- Base LLM to LLM Comparison (given the same variables across the board how do other LLMs hold up against each other)
- Human and Machine Language Compatibility (How does changing the human language, vocabulary and programming language affect quality)
- Problem Type (how does open evolve handle types of problems)
- Evolution vs Knowns (How does the quality of the output of open evolve hold up against the current known quality).

To also include variables for the evaluator script, which have yet to be determined. The current intent is to formulate the methods of creating an evaluator script and then manually implement them. Automation of this evaluator script creation after speaking with the sponsor was determined to be a stretch goal.

For the current categories the measurement for each of these should be simple, as the evaluator script must already compare correctness and speed across each iteration. However for the categories that relate to the prompt, a custom script will have to be created to handle testing different prompts across the same instance for convenience.