Logic-LM Task 5: Testing and Results

For Task 5, I set out to reimplement a simplified yet fully compliant version of the Logic-LM reasoning pipeline. My goal was to closely follow the core workflow described in the Logic-LM paper, while adapting it to work with a knowledge base built from CIFAR-10 image labels.

To start, I converted the CIFAR-10 labels (such as "cat", "truck", etc.) into symbolic logic facts. Alongside these facts, I wrote simple logic rules that define which labels represent animals and which represent vehicles. This setup allowed the system to answer questions like "Is cat an animal?" or "Is truck a vehicle?" by reasoning over these facts and rules.

A key design choice was to store knowledge as objects (for example, Animal(cat) or Vehicle(truck)). This approach makes the knowledge base more flexible and organized, allowing for easier expansion and richer reasoning. By representing entities and their relationships in this way, the system can more naturally handle new facts or rules, and the logic remains clear and maintainable.

To process natural language questions, I integrated a real large language model (LLM) from Together AI (Llama-3-8b-chat-hf). The LLM translates each question into a symbolic logic statement, which is then passed to a backward-chaining symbolic solver. This solver checks whether the logic statement is entailed by the knowledge base.

Additionally, I included a self-refinement loop. If the LLM's output isn't in the correct format or the solver encounters an error, the system uses feedback from the solver to prompt the LLM to revise its answer. This iterative process aligns with the self-refinement mechanism described in the Logic-LM paper and ensures more robust symbolic reasoning.

Note that the unification (matching of variables to arguments in rules) in my solver is implemented for one parameter at a time. In other words, each rule and query is handled in the form Predicate(argument), which is sufficient for the types of logic and queries used in this project.

When I ran the tests in Google Colab, the solver correctly answered "Yes" for cases like "cat" being an animal and "truck" being a vehicle, and "No" when the label didn't fit the rule (such as "automobile" as an animal). Some queries produced negated symbolic forms, which the solver currently treats as negative answers.

GitHub Repo: https://github.com/suvalavala/llm-logic-kb