Künstliche Intelligenz und Data Science WS 2023/2024

Lab 3: Logics and Intelligent Agents

Due: January 7, 2024, 11.59 CET

Dr. Vincenzo Perri
vincenzo.perri@uni-wuerzburg.de
Prof. Dr. Ingo Scholtes

Escape of the Wolpertinger: A Propositional Logic Adventure

In the mystical alpine forests of Bavaria, a whimsical creature, the Wolpertinger[1], finds itself trapped in a network of interconnected caves. A crafty Hunter has cunningly ambushed the Wolpertinger and, to facilitate the capture, has blocked the main entrance to the cave network.

The Wolpertinger knows the cave has a secret exit but does not remember its location. However, this escape route can be recognized from neighboring caves through intermittent flickers of sunlight.

After blocking the entrance, the Hunter, confident of his advantage, takes a break and eats his meal in a fixed location. Thanks to its powerful sense of smell, the Wolpertinger can smell whether the Hunter is situated in a neighboring cave. Realizing that now is its only chance to find the secret exit and elude the Hunter, the Wolpertinger springs into action.

Objective

Develop a program that employs refutation resolution in propositional logic to aid the Wolpertinger in moving through a network, avoiding the Hunter, and locating the secret exit. To achieve this, implement a theorem prover in propositional logic based on refutation resolution (e.g., DPLL). The prover takes as input a set of premises $F_1, \ldots, F_n$ and a goal formula G. The function returns

[1] In the tales of Bavarian folklore, the Wolpertinger is a mythical creature believed to inhabit the forests of Bavaria and Baden-Württemberg. It is described as a fusion of different animal elements such as wings, antlers, tail, and fangs, all intricately connected to the body of a small mammal. The most widespread description portrays the Wolpertinger as having the head of a rabbit, the body of a squirrel, the antlers of a deer, and wings, sometimes accompanied by the legs of a pheasant.

Figure 1: A rare depiction of the elusive Wolpertinger.

true if the goal formula deductively follows from the premises and false otherwise. At each step, your implementation should use the Wolpertinger's perceptions to assess the network. Based on these perceptions, the program should execute refutation resolution, and guide the Wolpertinger along the network of caves. Finally, the program should indicate the Wolpertinger's triumphant evasion upon reaching the secret exit or declare that it has been captured by the Hunter.

Network Representation

The cave network is provided as a collection of edges.

Escape Rules

The Wolpertinger follows a set of rules to navigate the network and ensure a successful escape:

1. If it is logically inferred that the secret exit is in one of the neighboring nodes, move directly to that node.

2. Use logical inference to determine potential locations of the Hunter in the neighborhood. Avoid moving to a cave where the presence of the Hunter has been logically deduced.

3. If it is logically deducible that some non-visited neighbors are safe, move to the identified safe node. In cases of multiple safe caves, randomly choose one.

4. If none of the above conditions are met and there is an adjacent cell with an unknown mark, move to that cell. If multiple such cells exist, randomly choose one.

5. If no adjacent cells have unknown marks, move to a safe node.

You are free to design additional rules to facilitate inference, but this is not a mandatory part of the assignment.

Provided Functions

In the `logic.py` file, you will find a helper class and functions designed to simplify the creation of logical sentences in Python. The initial section of `WolpertingerEscape.ipynb` offers examples showcasing their application.

Instructions

- Design an algorithmic approach for the Wolpertinger's escape based on the specified rules.
- Submit the code and a brief report detailing your implementation choices and testing results.

Submission

Submit a zip file with Python files and a brief report via WueCampus.