CIS700 HW4 Writeup

[redacted]

February 27, 2022

Contents

1 What wikiHow article did you pick and why? 1

2 What portions of the article did you select to translate to PDDL? 1

3 Give some examples of the actions, types, and predicates you used in your domain. 2
3.1 Actions . . . 2
3.2 Types . . . 2
3.3 Predicates . . . 2

4 Explain what goal you selected for your problem, and give the initial state and solution that you created. 2

5 What limitations of PDDL did you encounter that makes it difficult to precisely convert a wikiHow description into PDDL? 3

6 Could your PDDL be used as an interesting challenge for a text-adventure-style game? If so, how? If not, what would needed to create an interesting challenge? 3

7 Discuss how you might use GPT-3 to automatically or semi-automatically convert a wikiHow article to PDDL? 3

1 What wikiHow article did you pick and why?

We picked "How to Survive in the Jungle". This is because we thought that that the setting was more conducive to the types of environments we had seen so far. In particular, there were many types of possible locations (stream, grove, thicket, etc.) and many well-defined actions and requirements. Other wikiHow articles had more abstract goals such as "How to Spy on Someone" which was less clear cut than needing to boil water so that it is clean to drink. Also, we thought that many of the goals and predicates in this environment would be more universally (or at least broadly) applicable.

2 What portions of the article did you select to translate to PDDL?

The three high-level tasks we translated to PDDL were:

1. collecting and purifying water
2. gathering and cooking food
3. gathering materials to build a shelter

One issue is that the language used in PDDL and the language used in wikiHow were noticeably different. In PDDL, many of the important ideas were in abstractions - such as noting that a location would have the properties `has_fire` or `has_shelter`, or that a stream location was a `water_source`. Unfortunately, since

wikiHow was meant for humans to read, many of these obvious ideas that could be formulated as logical predicates were not specifically enumerated in the wikiHow article. For example, building a shelter entailed creation of a thing that was fixed at a location, so we needed a predicate like `has_shelter ?loc`. And if we boiled water, we would add `inventory ?player ?clean_water`, because it was implicit in natural language that when one produced boiled water, it would likely be carried around in a container.

Moreover, the article did not specify general specifications like the fact that humans need water, need food, or need shelter. Instead, its writing took these for granted (as these are obvious to all human readers), so we encoded them as the relevant predicates `is_thirsty`, `is_hungry`, and `is_tired`.

3 Give some examples of the actions, types, and predicates you used in your domain.

3.1 Actions

- Collect water
- Make fire
- Boil water
- Get (food)
- Cook food
- Make fire

3.2 Types

`container`, `igniter`, `sticks`, `leaves`, and `eatable` were all subtypes of `item`. `water` and `food` were subtypes of `eatable`.

3.3 Predicates

Collecting water required a container in inventory and a water source at the player's location. Building a shelter required the player to have sticks and leaves in their possession, and for a tree to be present at the player's location.

4 Explain what goal you selected for your problem, and give the initial state and solution that you created.

The goal for our final problem was to complete tasks that would be roughly needed to successfully make it through a period of time in the jungle - collect water and boil it, collect and cook food, gather materials and build shelter. The initial states had predicates related to being hungry, thirsty, and tired, which were solved by eating food, drinking water, and sleeping in shelter, respectively. We also checked that the player did not become "sick" (from eating poisonous or contaminated food or water) in accomplishing these goals.

The initial states had the player at a camp with nothing in their inventory. A rough sketch of the steps needed to complete all goals is:

1. Collect a bottle and lighter from camp
2. Go to the stream and collect water
3. Go to the grove and collect sticks
4. Go to the thicket and collect fruit and leaves
5. Start a fire
6. Boil water and cook fruit
7. Drink water and eat fruit
8. Go to grove and collect sticks again
9. Build shelter at grove

10. Go to sleep at grove

5 What limitations of PDDL did you encounter that makes it difficult to precisely convert a wikiHow description into PDDL?

One issue we encountered was when we tried to use `or` statements in our predicates. To circumvent this, we used DeMorgan's laws or we broke our actions into multiple copies for each of the possible cases of the disjunction.

Another thing we found was annoying was that concepts could be described in multiple ways. Water that had been made safe to drink could have had code representations of "boiled", "clean", or "not contaminated" despite these all describing the same thing. Moreover, the wikiHow articles themselves were often not that precise and relied on the fact that they were talking to a human.

We also noticed that the wikiHow article often failed to sufficiently describe relevant conditions that we as humans knew were necessary (e.g. needing wood and some kind of ignition method to start a fire).

Another issue we ran into was trying to efficiently use subtypes and accidentally being too permissive. We originally had water as a subtype of food so that we could re-use the predicate `is_contaminated`, but we realized that one could satisfy their hunger by drinking water, which is not what we intended. So we ended up making food and water separate subtypes of a new type `eatable`.

6 Could your PDDL be used as an interesting challenge for a text-adventure-style game? If so, how? If not, what would needed to create an interesting challenge?

We don't think our PDDL would be that interesting a challenge for a text-adventure-style-game. The main reason for this is that we primarily focused on making sure many things from the article were present, but that didn't create engaging puzzles or difficulty in accomplishing these tasks in a text adventure. Every action was in some sense locally achievable and also kind of obvious.

Moreover, the domain simply isn't that compelling because the main way to fail would be accidentally eating a poisonous mushroom rather than being killed by a troll. A more interesting challenge might involve a timer that limits the number of moves a player could make or other threats like wild animals, but these seem very difficult to encode in PDDL.

7 Discuss how you might use GPT-3 to automatically or semi-automatically convert a wikiHow article to PDDL?

Our experience here indicated that it will be very challenging for GPT-3 to automatically or semi-automatically convert a wikiHow article into a PDDL file. We relied on our human intuition to have a macro-level understanding of unwritten conditions and how they could be satisfied, which was especially important when the article did not describe dependencies well. Also, the language in the article omits many key details that would be known to humans but may not be known to GPT-3. Even with the benefit of isolating the phrases used for each PDDL component, interpreting them in a vacuum as PDDL would be challenging even for a human. The main reason we were able to do this is because we had context from the overarching article that, when used together with background knowledge, allowed us to construct a logically valid and interesting PDDL domain.