[redacted]

[redacted] [redacted]

1 Introduction

In this homework we convert a WikiHow article to PDDL language representation.

2 Questions

2.1 What wikiHow article did you pick and why?

We picked a WikiHow article on how to survive a shark attack and generated related PDDL domain and problems from it. The first reason we picked it is its soundness of plot, characters, and means to get out of the situation. It provided the trapped person of multiple ways to self-defense and / or escape from the shark attack. It also contains various states (being attacked, spotting the shark, etc.) that we could combine into the scenario with flexible logic chains. The second reason is that we want to work on something new, and this article is has a less fixed structure on the order of things. Thus we can better focus on the dynamic changes of the scenario.

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

We transformed the entire article into PDDL. However, for different components of the story, we transformed into different characters of the PDDL domain.

2.3 Give some example of the actions, types, and predicates you used in your domain.

For example, for "spotting the shark / ship" we combined them into a set of properties and actions. We made an action "spot" which takes in objects to be spotted; the predicate related here is "visible". At first, we specified both the ship and the shark to be plain "objects", but later we found that the searching algorithm would take too long. Thus we had to specify special types for them and modify related action signatures.

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

We selected 4 goals for our problems, very much coinciding with the specified order of escaping as in the article:
1. Get into a defense position. In the initial state, the npc is in the sea and asking for help; he is not wounded. There is also a shark not yet spotted lurking and a ship far away. (We used this for the common initialization for all scenarios.) And the solution is to just utilize your own body as weapons and get into a defensive position.
2. Repel the shark. To do this, we have to either make it dizzy (picking up a big stone) or to hurt it (by finding a spear near the shore).
3. Board the ship. To board the ship (in order to final escape the scenario), the npc needs to spot it after moving to the deep sea location where the ship is docked. After boarding the ship, he could get some medic attention.
4. All-done problem. The npc is required to do a combination of the previous 3 tasks, but the order is more subtle. For example, you cannot attack the shark after you have boarded the ship. Also, if you spend too much time fighting the shark, you might get into a deadend loop.

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

One of the problems is that we cannot do conditional effects. (We tried by importing the conditional-effect module, but it was not supported by the parser.) This leads to great inflexibility for modifying the game state conditionally. For ex-

ample, we cannot say ”if the shark bites the npc and the npc is already wounded, then he is out (he died)”. We must use additional, discrete states to track the npc's degree of woundedness and make multiple copies of the action accordingly (which we did not do, otherwise a simple logic requires too much actions and redundant predicates).
This leads to the other problem we found, which is we do not have a numerical evalution function supported by the parser. We cannot specify as indices how the person is, e.g. health, power, etc.; thus we cannot make the game more vivid.

2.6 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 think that our PDDL has great potential in being revised into a text-adventure-style game. It has abundant content and potentially many more plots than being specified by the WikiHow article. (For example, we might be able to include specific actions and a fighting simulation.) However, again due to the function restraints of the PDDL parser, much of this is not feasible.

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

First, we might need to fine-tune a GPT-3 model, by feeding in dataset structured as Prompt, Article, PDDL format. Then, we might use prompt learning to actually generate outputs from the model by feeding in only prompts and the original article. I would expect the model to learn very fast on the formatting of parameters (as it did on most JSON format inputs); however, whether it will generalize and combine various aspects (character / entity recognition, action recognition, parsing, and summarizing) is still unknown.