[redacted]

CIS 700: Interactive Fiction and Text Generation

February 24, 2022

HW 4: Convert WikiHow to PDDL

1. Introduction

In this project, we translated the wikiHow article “How to Survive on a Desert Island” to Planning Domain Definition Language (PDDL) format. We chose this article because surviving on a desert island is an interesting problem for works of fiction, as evidenced by the famous novels Robinson Crusoe, Lord of the Flies, and more. Besides being an interesting problem for works of fiction, a desert island is also a visually appealing environment with a rich landscape, so if we added an image component it could create an appealing game.

2. Article Portions

We selected 4 sections of the article to focus on:

- **Find clean water**: Method 1 Step 1. We selected this section because finding clean water is essential to survival, since a person would die after just a few days without water, and it is a common challenge in desert island survival stories.
- **Hunt for fish**: Method 1 Step 3. We selected fishing because although there are other ways to get food such as collecting fruits and berries, fishing would be more fun to do in a game.
- **Light a fire**: Method 2 Step 4. We selected lighting a fire because it is a very common challenge in survival stories, and requires a decent number of steps so it’s an interesting problem if we don’t have matches or a lighter.
- **Build a raft and escape**: Method 3 Step 3. We selected building a raft instead of other escape options like asking for help via radio, because it is a more dramatic escape so it is more fun for a game.

3. Actions, Types and Predicates

We defined the following actions, types and predicates for our “Survive Desert Island” domain.

Actions:

- General: go, get, build_shelter, find_other_survivors
- Find clean water: get_water, clean_water, drink_water
- Hunt for fish: make_weapon, hunt_fish, cook_fish
- Light a fire: chop_wood, carve_groove, light_fire
- Build a raft: build_raft

Types:

- Item: water wood fire rock leaves tinder raft vines spear fish
- Location: beach jungle ocean treetop
- Human: player survivor

Predicates:

- (treated ?water - water) ; True if the water has been decontaminated by boiling it
- (groove ?wood - wood) ; True if a small groove is made in wood to start a fire
- (at ?obj - object ?loc - location) ; an object is at a location
- (inventory ?player ?item) ; an item is in the player's inventory
- (connected ?loc1 - location ?dir - direction ?loc2 - location) ; location 1 is connected to location 2 in the direction
- (has_water_source ?loc - location) ; this location has a source of fresh water.
- (has_wood ?loc - location) ; this location has a wood
- (can_light_fire ?loc - location) ; this location is safe for lighting a fire
- (has_fire ?loc - location) ; this location has a fire going
- (has_shelter ?loc - location) ; this location has a shelter
- (drank ?water - water) ; the player drinks water
- (has_friend ?survivor - survivor) ; the player has found a survivor
- (has_escaped ?player - player) ; the player has built a raft and left with his fellow survivors
- (at_ocean ?loc - location) ; see if a location has access to the ocean
- (is_safe ?loc -location) ; see if a location is safe to make shelter on
- (has_fish ?loc - location) ; see if location has fish to catch
- (cooked ?item - item) ; see if item is cooked

4. Problems and Solutions

We created 4 problems, each corresponding to one step in the wikiHow article.

- **problem-clean_water**:
    - Goal: [['drank', 'water']]
    - Initial state:
        - [['can_light_fire', 'beach'], ['connected', 'beach', 'west', 'ocean'], ['connected', 'jungle', 'east', 'river'], ['connected', 'jungle', 'north', 'cave'], ['has_wood', 'jungle'], ['connected', 'jungle', 'west', 'beach'], ['connected', 'beach', 'east', 'jungle'], ['has_water_source', 'river'], ['connected', 'river', 'west', 'jungle'], ['connected', 'ocean', 'east', 'beach'], ['at', 'rock', 'ocean'], ['at', 'person', 'beach'], ['at', 'tinder', 'beach']]
    - Solution:
        - go west person beach ocean
        - get rock person ocean
        - go east person ocean beach
        - get tinder person beach
        - go east person beach jungle
        - chop_wood person jungle wood
        - go east person jungle river
        - get_water person river water
        - carve_groove person wood rock
        - light_fire person wood beach tinder fire
        - clean_water person beach water fire
        - drink_water person water
- **problem-catch_cook_fish**:
    - Goal: [['cooked', 'fish']]
    - Initial state:
        - [['has_fish', 'river'], ['connected', 'ocean', 'east', 'beach'], ['has_water_source', 'river'], ['at_ocean', 'beach'], ['can_light_fire', 'beach'], ['connected', 'jungle', 'east', 'river'], ['connected', 'river', 'west', 'jungle'], ['connected', 'beach', 'east', 'jungle'], ['connected', 'beach', 'west', 'ocean'], ['connected', 'jungle', 'west', 'beach'], ['at', 'person', 'beach'], ['at', 'survivor', 'river'], ['has_wood', 'jungle'], ['connected', 'jungle', 'north', 'cave'], ['at', 'rock', 'ocean'], ['at', 'vines', 'jungle'], ['at', 'tinder', 'beach']]
    - Solution:
        - go east person beach jungle

        - get vines person jungle
        - chop_wood person jungle wood
        - go west person jungle beach
        - get tinder person beach
        - go west person beach ocean
        - get rock person ocean
        - carve_groove person wood rock
        - light_fire person wood beach tinder fire
        - make_weapon rock person wood vines spear
        - hunt_fish person river spear fish
        - cook_fish person fish fire beach
- **problem-start_fire**:
    - Goal: [['at', 'fire', 'beach']]
    - Initial state:
        - [['has_wood', 'jungle'], ['at', 'person', 'beach'], ['at', 'rock', 'ocean'], ['connected', 'jungle', 'north', 'cave'], ['connected', 'beach', 'east', 'jungle'], ['connected', 'beach', 'west', 'ocean'], ['connected', 'jungle', 'west', 'beach'], ['connected', 'ocean', 'east', 'beach'], ['at', 'tinder', 'beach'], ['can_light_fire', 'beach']]
    - Solution:
        - go east person beach jungle
        - chop_wood person jungle wood
        - go west person jungle beach
        - get tinder person beach
        - go west person beach ocean
        - get rock person ocean
        - carve_groove person wood rock
        - light_fire person wood beach tinder fire
- **problem-escape_island:**
    - Goal: [['has_escaped', 'person']]
    - Initial state:
        - [['has_water_source', 'river'], ['connected', 'jungle', 'north', 'cave'], ['can_light_fire', 'beach'], ['connected', 'river', 'west', 'jungle'], ['at', 'tinder', 'beach'], ['at', 'person', 'beach'], ['connected', 'beach', 'east', 'jungle'], ['has_wood', 'jungle'], ['at', 'rock', 'ocean'], ['at', 'survivor', 'river'], ['at_ocean', 'beach'], ['connected', 'beach', 'west', 'ocean'], ['connected', 'ocean', 'east', 'beach'], ['connected', 'jungle', 'east', 'river'], ['connected', 'jungle', 'west', 'beach'], ['at', 'vines', 'jungle']]
    - Solution:

- go east person beach jungle
- go east person jungle river
- find_other_survivors river survivor person
- go west person river jungle
- get vines person jungle
- chop_wood person jungle wood
- go west person jungle beach
- build_raft beach vines person wood

5. Limitations of PDDL

PDDL has no concept of time, so some wikiHow instructions like “eat a small portion and wait 1-2 hours before eating the rest to avoid poisonous food” are hard to implement. There might be a way to define “wait” as an action, but defining hours, days, etc. is not straightforward. PDDL also requires you to specify everything, even common sense knowledge that wikiHow writers have. For example, wikiHow assumes that readers know that if you don’t drink water you will become dehydrated and die, but we would have to specify that in PDDL.

6. Using Desert Island PDDL for Text Adventure Game

While each of the 4 problems alone are not interesting enough challenges for a text adventure game, we could combine them to create an interesting game. We could create a game where the ultimate goal is to escape the island, but perhaps it takes weeks to gather the materials to build the raft, so you need to learn how to start a fire, get clean water, and fish in the meantime. To make it even more interesting, you could allow interaction with the other survivor you find, and either fight or befriend them which would lead to distinct outcomes.

7. GPT-3 for Converting WikiHow to PDDL

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

We could use GPT-3 to translate a wikiHow article to PDDL, with some human supervision to edit the generated PDDLs. First, we’d need to collect many JSON files such as the one we produced in this homework mapping the steps in the article to problems in the PDDL, and mapping phrases in the article to actions, types, and predicates in the PDDL. Second, we would tune a GPT-3 model to take in a wikiHow article, and output a JSON object like the one generated in class. Third, we would apply this model to a few wikiHow articles, and

manually edit the outputs to make sure they are reasonable. Finally, we would parse the edited JSON files to create a PDDL from it.

Although it would be possible to skip the manual editing step, I think that at least initially it is necessary to produce reasonable PDDLs. Once we have produced a good amount of edited JSON files, we could retrain the model with this larger training set and improve it.