CPSC 490 Project Description: Ticket To Ride

Background:

*Ticket To Ride* is a multiplayer board game focused on path building. The board consists of a map with cities connected by paths of different color and length, as shown below.

At the onset of each game, players are given 4 train cards at random as well as destination cards specifying routes they must complete, such as Los Angeles to Atlanta. Each destination card comes with a
number of points earned for completing the path, the same amount of points that are lost should the player not connect the two cities. Each player has 45 trains with which they can build.

Each turn, a player has three options to choose from.

1. A player may draw train cards
   a. The table will have 5 train cards face up and a deck to its side. A player may choose a train card from one of the 5 face up, or choose to draw at random. A player gets two choices, unless with their first choice, they choose a locomotive -- a wildcard that can be used as any train.

2. A player may build a path.
   a. To build paths, players must play train cards corresponding to the path on the map. For example, to build from Miami to Charleston, a player must play four pink trains.
   b. To build a gray path such as Los Angeles to Las Vegas, a player may play any two trains of the same color.
   c. Only one path can be built per turn, and it must be built in its entirety.

3. A player may draw additional destination cards
   a. Three cards are drawn, and the player must keep at least one.

The game ends when one player has two or fewer trains remaining with which they can build. At that point each player, including the player with two or fewer trains remaining, plays only one more turn.

Players score points by successfully completing destination cards, lose points if they fail to complete destination cards, and score points for each path they built on the map. Players get one point for each path of length one; 2 for 2; 4 for 3; 7 for 4; 10 for 5; and 15 for 6. The winner is the player who scores the most total points.

The entire rulebook can be found here.

Project:

In the fall of my junior year, I took CPSC 474: Computational Intelligence for Games with Professor James Glenn, a class in which I learned the basic skills of building bots to play games. For my senior thesis, under the guidance of Professor Glenn, I will be applying and further developing my knowledge of computational intelligence to create a bot to play Ticket To Ride.

The objective of this project is two fold: first, to use reinforcement learning to create a bot to play Ticket To Ride; and second, once the bot is developed, to investigate its ability to successfully perform transfer learning when the conditions of the game are changed. The goal is not just to build a bot that maximizes its ability to play Ticket To Ride as described in the rulebook but to construct a generalized bot to successfully play both Ticket To Ride and similar variants. Potential variants include: altering the map, such as adding, removing, or adjusting paths between cities; altering scoring rules, such as the points scored for certain path lengths or scoring rules surrounding destination cards; and altering game resources, such as the number of train cards that exist in the deck or number of train pieces each player has available. Understanding what specific variants the bot is able to learn successfully is a key goal of this project.

In order to complete this project, I will begin with an existing implementation of the game from the following GitHub repo. To successfully complete the project, scripts will be coded in Python to successfully interface with the existing Java implementation of the game. From there, the Python scripts
should be able to successfully read the game state and reply with the best move. Furthermore, Python will be used for the learning itself, utilizing Tensorflow libraries.

For an explanation of the game Ticket To Ride, please refer to the background section below.

**Deliverables:**

1. Successfully implemented interfacing from the Java Ticket To Ride implementation to Python scripts, where learning and decision making will take place. This interfacing must be in two directions: the Java code must output the game state, which must then be read via Python for learning or move making. Moves must then be communicated back to the Java scripts, where gameplay continues. Furthermore, it should be noted that existing agents exist within the GitHub repo. These may be used to assist with learning, but it is more likely that new meta-actions will be implemented either on the Java or Python side.

2. A bot, built via reinforcement learning, optimized to play the existing rules of the game.

3. A written report detailing how game state was encoded for learning, parameters used in the model described in point 2, and what the bot’s apparent strategy is for optimizing its gameplay.

4. * A bot, also built via reinforcement learning, built for generalized learning to conduct transfer learning (note: this bot may, in fact, be the same as the bot built in point 2).

5. * A written report detailing Ticket To Ride variants that will be used for transfer learning.

6. * Successful implementation of variants as described in point 5 paired with the transfer learning on said variants.

7. * A written report detailing the success of the bot’s transfer learning, described in point 4, with the goal of understanding under what circumstances transfer learning is more or less successful.

8. A final project report, as described in the CPSC 490 guidelines.

Note: points marked with an * are to be completed given time exists within the semester but are not mandatory to the completion of the 490 project as described.