Modeling Employer-Employee Salary Negotiations Using Mixed Strategies and Multi-Agent Reinforcement Learning

Sep 22, 2022

CSEC 491 Project Description
Student: Alexander Chen
Advisor: James Glenn

Introduction

Salary negotiation, especially for candidates aiming for high-paying tech roles, can result in compensation increases in the range of tens of thousands of dollars. Today, there are many sources of written and video content offering advice for optimal salary negotiation techniques. Furthermore, salary negotiation services, including Levels and Candor, provide expert help in negotiating offers. These services rely on teams of recruiters and coaches who work with individuals to craft a script to increase their offers.

On the employer side, negotiation tactics for hiring managers as well as a wealth of industry compensation data offered by third-party companies attempt to balance reducing compensation costs with successfully hiring top candidates.

This project aims to model salary negotiation as a game, with a pool of candidates and employers making offers, negotiating offers, and ultimately accepting or rejecting offers.

Description

Researching different strategies and multi-agent reinforcement learning algorithms/packages

Some employers/candidates will have fixed strategies reflecting existing salary negotiation behavior. This could include accepting the first offer that meets an employee's criteria without any negotiation, refusing to negotiate and offering all employees the same offer, and more. These strategies will need to be researched and defined.

Additionally, some candidates will use reinforcement learning to determine the best action to take in the environment, with the reward being a function of compensation and other factors valued by the candidate. This project will focus on the implementation of existing multi-agent reinforcement learning algorithms, likely using packages

Modeling the salary negotiation game

Inputs

1. What are the variables?
    a. Competing offers
    b. Current compensation
    c. Different compensation components (base, stock, bonus, relocation, benefits, etc.)
    d. Employee demand for role
    e. Employer's willingness to pay, employee's willingness to accept
    f. Employee's desired location
    g. Position/industry
    h. Years of experience/employee competence
    i. Perception of external job market
    j. Discount rate (employees value offers made earlier versus later)
2. Compensation data
    a. Levels.fyi
    b. Comparably
    c. H1B salary data

Agent strategies

Strategies will need to be defined and implemented in the game.

Candidate agents using multi-agent reinforcement learning will be implemented with a package like PettingZoo. PettingZoo is a standard API for multi-agent reinforcement learning, and includes several model environments as well as support for third-party environments. PettingZoo interfaces with Agent Environment Cycle (AEC) games. In AEC game models, agents observe, take actions, rewards are emitted from the other agents, and the next agent to act is chosen. Thus, the game model for salary negotiation would need to follow this sequential pattern.

Using PettingZoo, fixed strategies (like only offering candidates the same offer, or accepting the first offer that meets criteria) can be implemented as separate agent policies. Additionally, manual control can be used to allow a human user to control an agent within the game environment. This could be used to make the project more interactive, allowing a person to see how they would navigate the recruiting and negotiation process.

Outputs

1. Some representation of the current state of the game at each step (likely a text-based terminal output)
    a. Attributes of each agent

    b. Results achieved by each agent

Deliverables

1. Code repository and instructions to start training simulation for game model environment
    a. Explanation of output: what actions do we observe game agents taking to optimize compensation and other factors?
2. Presentation describing process and results

How does this combine computer science and economics subjects and techniques?

This project combines game theory and mathematical analysis along with software development and computational intelligence techniques to build an end-to-end game environment with agents that learn to play the game.