Modeling a Goalie's position with Reinforcement Learning

Mattia Danese and Evan Loconto

1 Introduction

In the world of professional sports, nothing is more important than beating your opponent. Whether one plays on a team or individually, winning is the be-all end-all. There are many sports played professionally, however no sport is as scrutinized as the game of soccer.

Soccer, also known as football, is a sport played and watched by billions across the world. Soccer is a game involving two 11-player teams, two nets, and a soccer ball. Each team tries to score into their opponent's net more times than the other team. The one caveat however is players cannot use their hands to achieve this goal; they must use other body parts (primarily their feet). One player, however, can use their hands in certain scenarios: the goalie. The goalie is the one position in soccer that is allowed to use their hands in a designated box outside of their net. The objective of the goalie is to defend, or "save" balls from going into their teams net. Each team is only allowed one goalie on the field at a time, meaning a lot of pressure mounts on this singular position. A popular tactic numerous teams employ is to simply focus on outscoring their opponent; though, many other teams take a more defensive approach (especially if they are stronger defensively rather than offensively) and focus on minimizing the goals of the other team as a means to victory. In either case, the success of the team more often than not hinges on how well their goalie performs.

With the World Cup, one of the largest sporting tournaments worldwide, coming up, there is even more focus on the performance of teams and their individual players. This is because, in the World Cup, the stakes are much higher for the players. Players are not just competing for money and fame, they are competing for national recognition and are representing their nation on a world stage, an opportunity which only comes once every 4 years. As such, anything from in-game tactics, the changing or substitution of players, to even how players train and practice are scrutinized even more by fans, media, and other teams. It goes without saying that, at the time of the World Cup, any movement and decision made by a goalie are at everyone's center of attention.

Seeing as how vital a goalie is to a soccer team and how the World Cup is coming up, an interesting idea was brought up: where is the ideal position the goalie should stand given certain scenarios? More specifically, how should the goalie be positioned in the scenario where preventing a goal is most difficult (i.e. a 1v1 between goalie and opposing attacker)? Obviously the goalie should stand somewhere in between the ball and goal, but is there an optimal spot? Should the goalie be standing far in front of the goal or closer to the goal line? In order to determine this, the goal of our project is to create a reinforcement learning model that maps the location of a goalie given the location of the opposing attacker.

2 Background Related Work

2.1 Background Knowledge

In simplest terms, reinforcement learning is when an agent *interacts* with an environment as a means of learning a specific task. This is strictly different from other machine learning techniques because, with reinforcement learning, there is no pre-existing data enabling the agent to "grade" every decision it makes. Instead, a reinforcement learning agent begins with some *policy* (a mapping from states to actions), which initially is usually very wrong, and uses this policy to choose its next *action*. Once the agent performs the action it picked, it will observe a *reward* from the environment and transition to the next state, which is directly determined by the previous state and action taken. The agent learns by updating the approximate value of the state and action pairing (how it does so heavily depends on the chosen reinforcement learning method) which, under the necessary assumptions, will converge to the true value of that state-action pair. As such, with a good reward function, the agent will strive to either minimize its reward (i.e. when the goal is to be completed as quickly as possible) or maximize its reward and, in turn, tweak its policy (i.e. through state-action pair updates) until it has converged to an *optimal* policy.

In terms of our project, we will be using an off-policy Monte Carlo method. The basis of Monte Carlo methods is running many simulations of the agent in the environment with the purpose of accumulating a lot of data. With this observed *real* data, the agent can then estimate (and update its estimates of) the values of states and actions in the environment and formulate (and improve) some policy. Monte Carlo methods can be broken down into *On-Policy* and *Off-Policy* methods. The key difference between the two is that On-Policy Monte Carlo methods use the policy they are trying to learn for interacting with the environment while Off-Policy Monte Carlo methods have a *behavior* policy (used to determine actions) and a potentially different *target* policy (the policy being updating and improved). We chose to employ an Off-Policy, rather the On-Policy, Monte Carlo method because of the added benefit that the agent can still sample all possible actions while strictly improving the *target* policy. Additionally, Monte Carlo methods are a type of model-free reinforcement learning (the agent does not have a complete probability distribution of states and actions) which we deemed appropriate since, from the goalie's perspective, the attacker could theoretically shoot anywhere regardless of its position or the position of the goalie.

2.2 Related Work

Our project is certainly not the first that applies reinforcement learning to the game of soccer, or skills related to the game of soccer. There have been numerous projects and published papers regarding reinforcement learning agents in some *soccer* environment, though not many cover the problem of optimizing goalie position. Instead, the vast majority of published papers take the perspective of the attacker and have the agent learn to kick the ball, shoot and score, or optimize goal-scoring scenarios. Interestingly, there are also some papers that revolve around optimizing strategy and tactics or evaluating how likely it is for a particular team to score or concede a goal given some state (i.e. snapshot) of a soccer game.

One paper, published just a couple of months ago by Ji et al., regards a Quadrupedal Robot learning how to accurately kick a soccer ball and hit random targets. Seeing as the agent and environment were not simulated and instead the experiment was conducted in the physical world, where aspects like friction and the deformity of the ball are hard to quantify, Ji et al. took the approach of developing a *model-free* RL framework. In order to aid the agent's learning, Ji et al. also decided to deconstruct the goal of hitting a random target into two subproblems: motion control when kicking the soccer ball with one leg and balancing on the other three and motion planning in order to find the optimal way to kick the soccer ball [2]. Even though our experiment deals with the position of the agent and does not consider teaching movement to a physical robot, the work of Ji et al. is still relevant to us. Our project will also deal with randomness (the position of the attacker) and following a similar methodology of decomposing the agent's goal into subproblems (such as learning where the goalie should be and learning whether the goalie should dive or not) could prove beneficial.

A second paper worth mentioning, which is more comparable to our experiment, is *Refinement of Soccer Agents' Positions Using Reinforcement Learning*. In this paper, the author, Tomohito Andou, configures an 11-agent soccer team that will compete in the simulated RoboCup tournament. The specifics of the tournament are elaborated on in their paper. The problem Andou tackles in their paper is learning the optimal *position* for each agent on his RoboCup team. As is explained in the paper, the basis of learning the optimal position for an agent is answering the following question: what is the position where I (the agent) can kick or get the ball most frequently given some state of the game? This overarching question is akin to our experiment because our goalie agent will also have to answer a similar question: what is the position where I (the agent) can save or get the ball most frequently given some state of the game? The experiment undertaken by Andou is clearly more complex than ours, since it takes into account a complete snapshot of the entire field and multiple agents, though a few aspects of their methodology can still be applied to our experiment. For example, the distance to the ball and to the goal are deemed integral by Andou in order to determine agent position. In our experiment, we will make the same assumptions and base the goalie's position on where the attacker is relative to the goal (how far back and how far left or right). Additionally, Andou utilizes stochastic gradient ascent (SGA) as the learning method for the agents' positions. Though not completely necessary for our experiment given the relatively simple and discrete state space, this is a method we could explore in the future if we choose to expand our experiment and allow attacker and goalie to continuously move before the attacker kicks the soccer ball. [1]

3 Technical Approach / Methodology / Theoretical Framework

3.1 Overview of Approach

We will approach this problem by having the agent start at a fairly neutral starting position: on the goal line and in the middle of the goal. The position of the attacker will be randomized and the location of where the attacker will shoot the soccer ball will be computed by a probability distribution based on the attacker's position. The agent will then be given the position of the attacker and, solely based on this, will decide on a next action which will potentially bring it to a new location. The chosen action is akin to the agent moving itself to where it thinks it has the best chances of preventing a goal. Finally, the attacker will "shoot" the ball. A reward -1 is given to the agent if the attacker scores, otherwise the agent will receive a reward of +1. The agent is then reset back to the neutral starting position and the process starts again.

3.2 Methodology

As alluded to previously, our agent will be employing an Off-Policy Monte Carlo method in order to learn the optimal position to be in given the position of an attacker. The *target* policy of the agent will be the greedy policy and the *behavior* will be the ϵ-greedy policy. Our experiment will consist of one continuous *episode* where the agent will keep repeating the procedure described in the previous subsection until its policy has converged, which we quantify based on how frequently the optimal action changes per attacker position. Seeing as our experiment will deal with discrete positions, we will ensure that the state space will allow for policy convergence within a reasonable amount of time.

3.3 Environment

A real-world "field" and "goal" can be described as almost like two connected sides of a cube: the field being the XY plane, the goal being the XZ plane, and both being perpendicular to each other. As such, we will represent both the field and the goal as independent 2D arrays. Arrays are efficient and simple data structures and have the added benefit of discretizing both the location of the attacker/goalie and where in the goal the attacker will shoot. The state-space of our experiment will be all possible locations of the attacker and the action-space will be all actions that the goalie can take (from the starting position) to get to any other valid goalie position. It is important to note that the location of the goalie is not part of the state-space because the goalie will start at the same starting location at the beginning of each iteration (actions are picked solely based on where the attacker is). Lastly, we will also enforce certain positional restrictions on the goalie (cannot move very far forward from the goal or past either post) in order to make the experiment more realistic and applicable to real-life scenarios.

4 Evaluation

In terms of success, our approach will be evaluated based on how much our agent can "save" the ball. This will be the main factor in success, seeing as our hypothesis is that there is, in fact, an ideal spot for the goalie to stand in order to save the ball. A goalie should save more balls if they are in an ideal position, meaning a goalie's save rate should be the determiner of the success of our model. We aim to track the save rate of the agent, and plan to include a graphical model of progress over a large number of iterations. Once we reach a state where the policy is close to optimal (where the save rate reaches a relatively high rate), we hope to gather information related to the goalie's position. Considering the position of the goalie will depend on the attacker's location, this means we should have a large range of positioning locations for each place the attacker is on the field. We then hope to map this set of data out as a visual aid. This can be done either by a set of pictures or an animation to show how our end product looks like.

A preliminary hypothesis we came to is that our agent's ideal location will be at the angle in which the goalie covers the greatest amount of goal between the attacker and goal. One question we are looking forward to see answered is how far up or back should the goalie be from the goal. With the goalie being further up, the angle of possible goal to score at is cut off meaning a greater save rate. However, there would be a lot less reaction time for the goalie to save a shot, and the attacker will have more of an ability to "chip" the ball over the goalie the further out he is. Our end implementation will most likely impact this result, and we plan to test and try to account for this in the most realistic way possible.

5 Timeline and Individual Responsibilities

In terms of timeline for the project, we plan on spending a fair bit of time researching the best methodology our algorithm should use. While we have a sense to use an Off-Policy Monte Carlo model, we still believe we should further our knowledge to ensure this is the best approach. This may be in the form of looking at similar projects, asking our peers/other teams how they are going about their work, to even further readings into the book or research papers. After this research phase, we aim to work on the implementation. We have talked, and have decided on meeting at least once a week to either give a status update on the coding we are doing, or to get together to work on the main piece of our project. Ideally, we have these steps done by the beginning of December. Once they are done, we plan on making either an animation or a set of images that represents what the optimal policy for our agent is. After this is done, we plan on making a presentation and doing the write up.

In terms of planning future work, we plan for each group member to work on all aspects of the research, implementation, and visual representation. Seeing as we are two people, we believe there is enough work that each person can work on a given part at the same time. Despite this, we still plan to split up several key aspects of the work, allowing for us to work individually to speed up the implementation process. We both plan on looking further into relevant projects/related work, and we both plan on working directly with the model, but nothing has been set in stone regarding what each person will be in charge of. Our plan going forward is meeting after more preliminary research on our first meeting. We then will set goals for the week and assign certain tasks to each other.

In terms of milestones, here is our vision:

1. Finish project proposal (11/5)
2. Research further into other projects (11/6)
3. Start implementation (11/8)
    (a) Checkup meeting 1 (11/8)
    (b) Checkup meeting 2 (11/15)
    (c) Checkup meeting 3 (11/22)
4. Thanksgiving break (11/23 - 11/27)
5. Finish implementation (11/29)
6. Animation/picture representation (12/4)
7. Write up and presentation (12/11)

References

[1] Tomohito Andou. *Refinement of Soccer Agents' Positions Using Reinforcement Learning.* Springer-Verlag Berlin Heidelberg, 1998.

[2] Yandong Ji, Zhongyu Li, Yinan Sun, Xue Bin Peng, Sergey Levine, Glen Berseth, and Koushil Sreenath. *Hierarchical Reinforcement Learning for Precise Soccer Shooting Skills using a Quadrupedal Robot.* 2022.