Workshop No. 1 — Kaggle Systems Engineering Analysis

Juan David Escallon Guzmán Juan Diego Lozano Luna
Jorge Eduardo Muñoz Gomez

April 5, 2025

Contents

0.1 Main Objective and Limitations of the AI Football Competition

Objectives of the Competition

The main objective of the competition organized by Manchester City F.C. and Google Research is to develop artificial intelligence agents with the ability to play football in a simulated environment.

In order to face all kinds of environments without fear of the failure that would come from doing so in official matches, the competition aims to, through simulated scenarios, observe the efficiency of different actions and possibilities that a player could face, thanks to all the data it has, such as position, ball rotation, and many other elements. But these scenarios are simulated under perfect conditions like weather or the field, which in a real match are fundamental variables that could greatly affect the development of the game.

Limitations

As mentioned earlier, these scenarios are simulated under perfect conditions. This creates a big limitation as it cannot transmit 100% of the environment that would be faced in a real match.

On the other hand, even though the environment presents a large number of possibilities and plays for the artificial intelligence to use and clearly improve, not all the range of possibilities that a real-life player has are present, such as tactical fouls, prepared set-piece plays, specific dribbles, or even acting to waste time on the field.

In addition to the lack of context that artificial intelligence could face, a player drawing 0-0 in the 20th minute would not play the same way as when losing 1-0 in the 80th minute. A sport like football, in which a match lasts so long, has different moments to face, and these must be handled differently depending on the needs of the moment the team is going through.

A major limitation that can be seen is the mentality and pressure that a player may face in an important match, which is clearly not reflected in artificial intelligence. And this is a very decisive factor when facing a match, in addition to the physical discomfort that players may suffer, which is not taken into account at any time.

Available Data (Observations)

Each agent receives, at every step of the game, an observation of the full game state, which includes:

- Position, velocity, and fatigue of all players (from both teams).
- Ball position and possession.
- Current match score.
- Previous actions (*sticky actions*).
- Information about the currently controlled player.
- Current game mode (corner kick, throw-in, etc.).

Available Actions (19 in total)

Each agent can select one action per turn, from the following:

- Movement: `Action.Top`, `Action.Bottom`, `Action.Left`, `Action.Right`.
- Sprinting: `Action.Sprint`.
- Shooting and passing: `Action.Shot`, `Action.Pass`, `Action.LongPass`, `Action.HighPass`.
- Ball control: `Action.Dribble`, `Action.ReleaseDribble`.
- Defense: `Action.Slide`.
- Other direction and control actions.

0.2 Key elements of the competition

Environment observations:

- Ball information: position, direction, rotation, and possession.
- Team information: position, direction, fatigue level, cards, and roles.
- Controlled player: specifies which player is being controlled at the moment.
- Game mode: includes contextual information such as corner kick, penalty, match start, etc.

Available actions:

- Up to 19 different actions can be executed, including movement, passes, shots, sprint, dribble, among others. These are detailed in the observation file.

Agent configurations:

- Architecture of the model used.
- Training algorithm applied.
- Observation representation technique: can be raw, simple115_v2, or pixels.

System process:

- To evaluate its performance, each agent automatically pits itself against other agents with comparable skill levels (determined by μ).
- Eight matches are played every day. Each agent's score is modified in accordance with:
  - The match's outcome (win, draw, or loss).
  - The variation (based on μ) between the expected and actual result.
  - The degree of uncertainty (σ) connected to every agent.

System outputs:

- Approximate rating: Denoted by the μ value of the agent.
- Last leaderboard: A list displaying each team's best agent.
- Performance history: Record of the performance of all agents.

0.3 Relationships Mapping

Based on the key elements identificated in the past sections, it's proposed the next diagram to representate the system of the competition, where it was drew the parts of the system, and the most significant relationships between them:

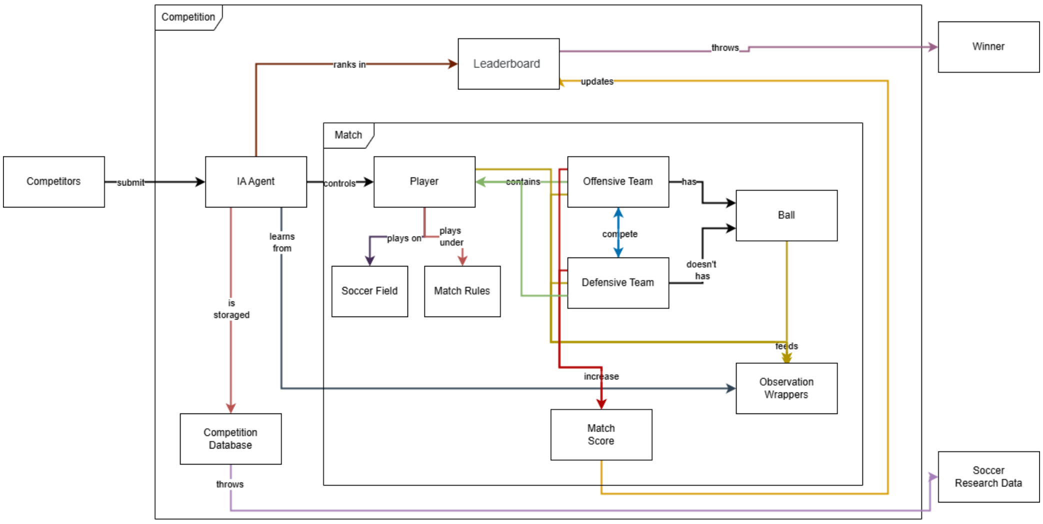

Figure 1: Competition as a system

1. **Inputs:**
   - Competitors who wants to participate in the competition.
2. **Elements of the system:**
   - IA Agents submitted by competitors.
   - Leaderboard that shows the top of the best IA Agents.
   - Competition Database that stores all of the submissions made by the competitors.
   - Match sub-system where the IA Agents compete with each other, composed of:
     - Players.
     - Soccer field.
     - Match rules.
     - Match score.
     - Offensive team.
     - Defensive team.
     - Ball.
     - Observation wrappers.
3. **Outputs:**
   - Winners of the competition.

- Soccer Research Data.

System Flow

- The system recieves competitors who submit their IA Agent to participate, having two processes with it: the IA agent is stored on the Database of the competition, and the IA agent is tested in a match against itself to test if it works. The competitor can upload up to five agents per day.

- If it doesn't pass the test, it will be returned as error. Otherwise, it is asignated with a base μ that represents the estimated skill of the agent, and a σ that represents the uncertainty of that estimate, and will decrease everytime that the agent plays.

- During the competition, the IA Agent with similar μ will play matches against each other. The winner increase its μ, while the loser decrease it. If it's a draw, both μ will move closer towards their mean. Im any case, the leaderboard will be updated.

- Starting a match, the simulation will assign randomly an IA agent per team. The teams are defined as left team, and right team, and unlike a real match of soccer, the teams will not switch sides in all the game.

- During the match, Observation wrappers are generated and delivered to the IA Agents constantly, those contains general information about the match, like the match mode, position of all players, which player has the ball, etc. Everytime an observation wrappers is generated, the match moves forward one step. The duration of the match is 30000 steps in total. No aditional time can be added.

- One IA Agent only controls one player of its team at time. If its team has the ball, the player will be the one with the ball. Otherwise, the player will be the closest to the ball. In every step IA Agents have to take decision based on the observation wrappers generated.

- At the end of the competition, the competitors on the podium receive awards. While all the submits made during the competition are left to be used by the Google Research Football.

0.4 Sensitivity Analysis

Sensitivity Analysis – Questions

The objective of this analysis is to determine to what extent the system output, represented by the agent's estimated performance (μ), varies in response to changes in different inputs or configurations. The aim is to identify which factors have the greatest impact on the agent's overall performance.

1. **Type of data observation**
   - Does the agent achieve better performance using observations in simple115_v2, pixels, or raw format?
   - The analysis will determine which type of input allows the model to better understand the environment.
2. **Actions used**
   - Does using a reduced or extended set of actions significantly affect the agent's performance?
   - It will be evaluated whether limiting or expanding the action options improves or worsens the agent's ability to make effective decisions.
3. **Training techniques**
   - How sensitive is the performance (μ) to changes in the training algorithm used?
   - Comparison among methods such as PPO, A3C, and DQN to identify which achieves higher learning efficiency.
4. **Submission frequency and quantity**
   - Does the frequency of agent submissions to the evaluation system influence performance?
   - It will be investigated whether it's more effective to submit agents continuously or only when significant performance improvements are observed.
5. **Initial uncertainty (σ) and early learning**
   - How beneficial is it to make multiple submissions during the early training stages compared to later stages?

- The goal is to understand whether an early approach helps reduce uncertainty and accelerate rating convergence.

6. **Agent policy change (exploration vs exploitation)**
   - Do agents with more conservative strategies (exploitation) or more risk-taking strategies (exploration) achieve better results?
   - This will analyze which approach tends to produce a higher μ value in the long term.

7. **Fatigue, roles, and red cards**
   - How useful is it to include this type of additional context in the observations?
   - It will be assessed whether considering these variables improves the agent's decisions during gameplay.

8. **Goals and goal difference**
   - How much does the number of goals or goal difference influence μ variation?
   - The aim is to understand whether the final match score directly impacts the rating adjustment.

Sensitivity Analysis – Hypotheses

1. **Type of observation**
   - Hypothesis: The way environment data is represented significantly affects the model's learning ability.
   - Expected observation:
     - simple115_v2: allows faster and more efficient learning, with a higher initial μ.
     - raw: may lead to lower performance if not properly processed.
     - pixels: requires more episodes and computational resources.
   - Sensitivity: High – Small changes in data representation can have a considerable impact on performance.

2. **Action set used**

- Hypothesis: A broader action set improves the agent's adaptability in complex situations.
- Expected observation:
  - Basic actions: quick but limited strategy.
  - Extended actions: enables more strategic and versatile gameplay.
- Sensitivity: Medium – Influences playing style and win probability.

3. **Training algorithm**
   - Hypothesis: Some algorithms promote better exploration or converge more efficiently.
   - Expected observation:
     - PPO: stable and commonly used as a baseline.
     - DQN: may struggle in continuous or complex environments.
     - A3C: can be unstable, but trains policy and value function simultaneously.
   - Sensitivity: High – Directly affects how the agent learns from the environment.

4. **Submission frequency and quantity**
   - Hypothesis: The submission strategy influences the agent's visibility and optimization.
   - Expected observation:
     - Frequent submissions: can quickly rank the best agent.
     - Selective and optimized submissions: avoid penalties from low-performing agents.
   - Sensitivity: Medium – Affects the amount and quality of feedback received to improve the model.

5. **Initial uncertainty (σ) and early learning**
   - Hypothesis: Early stages are critical for establishing good positioning in the matchmaking system.
   - Expected observation:
     - Early submissions with good performance allow for rapid ranking.

    - Late submissions may not reach optimal μ before the competition ends.
- Sensitivity: High – Initial uncertainty (σ) and playtime strongly influence the rating.

6. **Play style (exploration vs. exploitation)**
    - Hypothesis: Agents with high exploration may discover new strategies but compromise stability.
    - Expected observation:
        - High exploration: generates unexpected behaviors and potential tactical advantages.
        - Low exploration: produces safer but more predictable behaviors.
    - Sensitivity: Medium – May be beneficial or detrimental depending on the tournament stage.

7. **Fatigue, cards, and player roles**
    - Hypothesis: These factors may influence agent behavior, although their impact depends on the model and how the environment penalizes those conditions.
    - Expected observation: They will only be useful if the agent is advanced enough to learn from these signals.
    - Sensitivity: Medium – Add value in more complex and context-aware models.

8. **Goals and goal difference**
    - Observation: The scoring system μ is based solely on the match result (win, draw, or loss), not on the number of goals.
    - Sensitivity: Low – Winning by one or many goals does not affect μ update.

Visualization of Hypothetical Results (As part of the sensitivity analysis)

A graph was created based on hypothetical results, constructed from reasonable assumptions consistent with the rating update system rules (μ). This visualization does not use real data but reflects expected behaviors according to the dynamics of the competition environment.

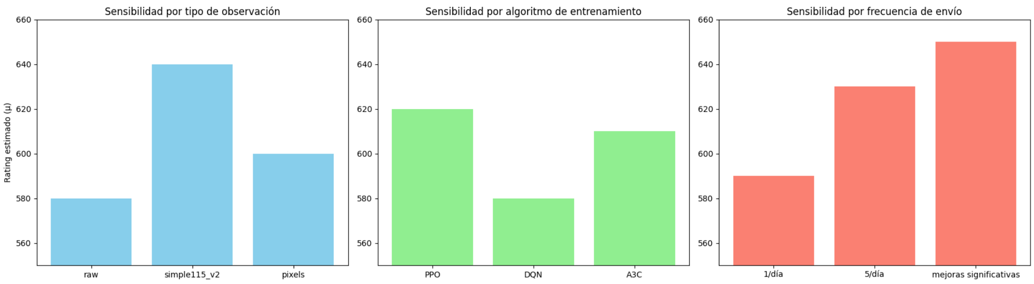

Figure 2: Score Projection Charts

Initial hypothesis:

The analysis starts with a base value of $\mu = 600$, common in matchmaking systems like TrueSkill or similar.

It evaluates how different configurations impact the evolution of the estimated rating during the first few days of the competition, within a possible range between 580 and 650.

This range is justified under the assumption that:

- Agents are matched against opponents of similar level.
- μ updates after each match are moderate.
- In the early days, μ changes reflect the adaptation and initial learning process.

Variables analyzed:

- Type of observation (simple115_v2, raw, pixels)
- Training algorithm (PPO, A3C, DQN)
- Submission frequency (high vs. low frequency)

Results Obtained (Graphs):

The graph shows how, depending on the configuration:

- Agents using simple115_v2 tend to show a faster and more stable growth curve.
- Agents trained with PPO display a more consistent progression compared to A3C or DQN.

- Frequent submissions allow for quicker adjustments in the system, favoring continuous improvement in μ, while sporadic submissions show less variation in the short term.

Conclusions

Variables	Sensitivity	Importance
Training algorithm	High	Affects decisions $\rightarrow$ Wins $\rightarrow$ Affects μ
Type of observation	High	More information $\rightarrow$ Better performance
Submission frequency	High	Better μ estimates and feedback
Action set	Medium	Influences the agent's tactical execution
Fatigue, Roles, Cards	Medium	Useful in specific moments
Play style	Medium	Specific, infrequent cases
Goals/Goal difference	Low	Does not affect μ updates

0.5 Chaos Theory in Football

A sport like football is largely subject to chance and, therefore, to chaos. In it, a single individual action can completely change the course of the match—such as a foul that leads to a player's expulsion, a counterattack that occurs independently of the game's context, a mistake by a player or even the referee. Even factors like the field and the weather are fundamental elements that do not depend on the players themselves and can seriously affect the outcome of the match.

All these types of situations are quite difficult to transfer into simulation environments, as these are usually carried out under perfect conditions. If such cases were incorporated into simulations, the results could be more efficient and realistic.

In these environments, artificial intelligence can misinterpret data due to random factors. An AI agent might develop a very effective strategy that allows it to control the match for most of the time, but due to one of these isolated events, it could interpret its strategy as ineffective.

Factors such as rebounds strongly influence the outcome of matches, and they are unpredictable, generating dangerous plays that could impact the final result.

0.6 Conclusion

From a system's point of view, the AI Football Competition illustrates a complex interplay between inputs (agent submissions and parameters), processing (match simulations and learning methods), and outputs (performance grades and research outcomes). The design of the system facilitates scalable evaluation via auto-matching, whereby agents of similar skill levels compete in sandboxed environments. This setup guarantees scalability and perpetual benchmarking, with participants able to tune their models according to observations from the leaderboard.

The system also has some contemporary limitations since it operates in a closed-loop fashion. The simulation provides structured observations, including the position of the players, where the ball travels, and game state. Still, it operates within a structured environment, discluding real-life challenges such as environmental conditions and human unpredictability. The structured action space prevents unusual behaviors as agents cannot implement unorthodox strategies outside designated actions. Furthermore, while the rating system is effective for establishing rankings, it may inadvertently encourage conservative playing styles that exploit limitations in simulation rather than showcasing genuine football skills.

Future improvements could include adding complexity by adding changing factors, like changing pitch conditions and referee mistakes, and adaptive rules that change depending on the agents' performance. A feedback loop based on real match data could also help connect the simulation back to reality. Ultimately, the strength of the competition is in the pipelined evaluation framework, but its long-term sustainability is in accepting the messy, nonlinear systems that characterize real football where agents have to deal not only with adversaries, but with the intrinsic uncertainty of the game itself.

Bibliography

[1] Google. *Google Research Football with Kaggle*, 2020. https://www.kaggle.com/competitions/google-football/overview. Accessed: 2025-04-04.

[2] Rules of Sport. *Football Rules.* https://www.rulesofsport.com/sports/football.html. Accessed: 2025-04-04.

[3] Kaggle. *Football Environment JSON Config.* https://github.com/Kaggle/kaggle-environments/blob/master/kaggle_environments/envs/football/football.json. Accessed: 2025-04-04.

[4] Kaggle. *Football Environment Python File.* https://github.com/Kaggle/kaggle-environments/blob/master/kaggle_environments/envs/football/football.py. Accessed: 2025-04-04.

[5] Google Research. *Google Research Football GitHub Repository.* https://github.com/google-research/football/. Accessed: 2025-04-04.

[6] Google Research. *Google Football Observation Documentation.* https://github.com/google-research/football/blob/master/gfootball/doc/observation.md#raw-observations. Accessed: 2025-04-04.