X. Appendix

In this appendix, we provide details on the implementation of our model (Section X-A) and the dataset used to validate our approach (Section X-B).

A. Implementation Details

In this section, we explain the compute resources used in this work (Section X-A1), and give further details on our input feature preparation (Section X-A2).

1) Compute Resources: This work was implemented using Python 3.8.15, with our GAPP model and baseline methods developed in PyTorch.[2] Model training was conducted on a remote GPU service using an NVIDIA T4 GPU with 51 GB of VRAM, which was facilitated through Google Colab. Training the GAPP model required a maximum of three hours for each fold. Inference for the GAPP model, used to compute the DI and DP metrics, was performed on a high-performance computing (HPC) facility. We utilised 360 parallel nodes, each with 4 GB of RAM, to handle the large-scale dataset used in our analysis. The specific HPC facility was the IRIDIS High Performance Computing Facility at the University of Southampton.

The XGBoost baseline was implemented using the XGBoost library.[3] The XGBoost model has the following hyperparameters: max_depth = 6, eta = 0.1, subsample = 0.8, colsample_bytree = 0.8. The Random forest model for the Dauxais [37] baseline was implemented using the scikit-learn library[4] and uses 200 trees (matching the original paper) with a max_depth of 6.

2) Feature Preparation: The categorical features in our input feature set (Section IV-A) are represented as binary indicators, as all of them are boolean variables. For numeric features, we apply normalisation using the StandardScaler from the scikit-learn library.

B. Dataset Details

In this section, we provide additional details about the dataset used in this work. As noted in Section VI, the dataset consists of 306 games from the 2023/24 English Premier League (EPL) season. However, not all games from the season are included, and we normalise for the number of events each team participated in. Table VI lists the teams included in the dataset along with the corresponding number of games.

Our dataset includes both on-ball events and tracking data for these games.

1) Tracking Dataset: The tracking dataset used for training and validating our model, as well as for computing the DI and DP performance metrics, records the locations of all players and the ball at a frequency of 30 Hz. Each tracking frame also includes a frame number, timestamp, and unique identifiers for each player and team. Our datasets use relative coordinates originated from the bottom-left corner of the pitch (0,0) to the top-right corner (105,68) from the perspective of the attacking team. The attacking direction is configured to always be from left to right.

TABLE VI
Game count for each team in our EPL 2023/24 dataset.

Team	Game Count	Team	Game Count
Liverpool	33	Chelsea	31
West Ham	33	Brentford	31
Burnley	33	Man United	31
Aston Villa	32	Nottingham Forest	30
Fulham	32	Arsenal	30
Man City	32	Everton	30
Luton Town	32	Tottenham	29
Bournemouth	32	Newcastle	28
Crystal Palace	31	Sheffield United	27
Brighton	31	Wolves	24

2) Event Dataset: Our dataset includes the timestamp and location of each on-ball event, along with the event type and the player responsible for the action. The possible event types can be categorised into attacking events (ball carry, cross, pass, and shot) and defensive events (challenge, clearance, and rebound). The outcome of an event, such as the location of the next pass, is determined by the location of the subsequent on-ball event in the dataset.

This data was provided by Gradient Sports.[5] A public repository containing the trained model, code, and results used in this work is available.[6] The hyperparameters provided in the paper are given in the code to aid model reproduction.

[2]https://pytorch.org/

[3]https://xgboost.readthedocs.io/en/stable/python/index.html

[4]https://scikit-learn.org/stable/index.html.

[5]https://www.gradientsports.com/.

[6]https://github.com/GregSoton/EvaluatingDefensiveInfluenceUsingGATs