Utilizing Graph Neural Networks to Model Major League Baseball At-Bat Outcomes

Selah Dean

Abstract

Baseball is a game of numbers and in recent years the use of data analytics has grown tremendously as teams look for any advantage they can get. Baseball analytics has traditionally relied on basic statistical models and machine learning approaches that tend to treat players in isolation. This approach overlooks the complex dependencies between players especially when considering the pitcher versus hitter match up that is central to the game. This project proposes modeling pitcher-hitter matchups as a bipartite graph, where nodes represent players and edges capture detailed at-bat interactions. It will utilize graph neural networks (GNNs) to predict future at-bat outcomes leveraging the relational structures in the data. The research will explore different GNN architectures and assess the importance of different features in the predictions. This work contributes to both baseball analytics and the broader field of sports analytics by introducing a novel application of GNNs to individual player interactions.

Introduction

Baseball is built from the hitter versus pitcher matchup. The hitter is trying to reach base while the pitcher is trying to get an out. Managers are also trying to find the matchup where their hitter or pitcher is most likely to succeed. Many factors can contribute to the potential outcome of the at-bat including simple attributes such as handedness, but there are also more complex attributes such as the pitcher's arsenal or the hitter's tendencies.

Traditional baseball analytical methods rely on statistical models and machine learning approaches that often treat players in isolation or use aggregate statistics. However, these methods may overlook the intricate relationships between pitchers and hitters. To address this limitation, this project will model the pitcher-hitter matchups as a bipartite graph, where nodes represent pitchers and hitters with their relevant attributes, and an edge exists between two nodes when a pitcher and hitter have previously faced each other. The edge contains features representing their previous at-bat including when the at-bat occurred, the number of pitches in the at-bat, and the ultimate outcome of the at-bat (strikeout, walk, single, etc.). Multiple edges can exist between the same pitcher and hitter if they have faced each other more than once over the season or their careers.

This project explores the use of graph neural networks (GNNs) to predict at-bat outcomes by leveraging the structural relationships within the pitcher-hitter network. It will examine if a GNN can effectively model pitcher-hitter interactions to improve at-bat outcome predictions. It will also assess what features contribute the most to predictive performance and how historical interactions influence future matchups. To answer these questions, this study will train a GNN on historical Major League Baseball (MLB) data using datasets containing individual pitcher and hitter statistics as well as detailed at-bat records. By capturing the complex dependencies between players, this approach aims to offer further insight into the pitcher-hitter matchup.

Related Work

The use of data analytics in sports especially in baseball has increased greatly since the early 2000s. In recent years machine learning applications in sports analytics have grown as well. Koseler & Stephan (2017) [1] did a systematic review of machine learning applications in baseball looking into how machine learning algorithms can be applied to find valuable insights into player and team performance. The literature they found primarily used traditional machine learning algorithms and found that support vector machines and k-nearest neighbors are some of the most common methods.

More recent research has demonstrated the effectiveness of GNNs in sports analytics. Tracy et al. (2023) [2] explored the use of graph-based encodings in volleyball. They modeled player interactions and game sequences using graph convolutional networks (GCNs), graph attention networks (GATs), and Graph Transformers. They found that GNN-based models significantly outperformed traditional machine learning models by preserving the inter-player dependencies and sequential interactions. Similarly, Xenopoulos & Silva (2021) [3] developed a general graph representation of game states that can be applied to a variety of sports. They used their model to predict yards gained in American football and estimate win probability in esports. They also found improved performance and they were able to answer the "what if" questions that occur in sports through their modeling of player interactions. These studies demonstrate the generalizability of GNNs in sports analytics and supports the idea that relational modeling improves predictive accuracy.

It appears that GNNs have yet to applied to the problem of predicting player matchup outcomes in baseball. Silver (2020) [4] developed a neural network model named Singlearity-PA that is designed to predict the outcome of plate appearances in MLB. Singlearity-PA provides more accurate predictions across various scenarios when compared to traditional methods that have been used for this problem such as log5. The model includes various predictive variables and handles specific game situations. The research I propose builds on use of deep learning but exploits the network structure of the data. In a flat deep learning model, at-bats are treated as independent events, so the evolving relationship between a pitcher and hitter is not captured. A graph-based model is also able to leverage the contextual information from other similar matchups.

Outside of the area of sports, there has been research done to explore the use of GNNs for modeling dynamic relationships and predicting future interactions. Kim et al. (2019) [5] propose an edge-labeling GNN (EGNN) which improves on few-shot leaning by updating edge-labels and considering both intra-cluster similarity and inter-cluster dissimilarity. This model also performs well for predicting multiple classes. The concepts from EGNN are relevant to the setting of modeling at-bat outcomes since the focus is on predicting edge labels. Additionally, Rossi et al. (2020) [6] introduce a temporal graph network (TGN) which is used for learning on continuously evolving graphs. The experiments in the study focus on link prediction rather than edge labeling and find that their results significantly outperform the baseline models. While the task in this study is future edge prediction rather than edge label prediction, the concept of a dynamic graph is relevant to modeling at-bat outcomes. Both pitchers and hitters make adjustments over time, so time can be an important element when considering the pitcher-hitter relationship. My proposed project of modeling MLB at-bat outcomes will build upon concepts from the EGNN and TGN models that have been introduced in existing literature.

Proposed Activities

I propose to develop a graph neural network (GNN) model to predict at-bat outcomes in Major League Baseball (MLB) by leveraging a bipartite graph representation of pitcher-hitter interactions. While GNNs have been applied to team-based sports such as basketball and volleyball, this project aims to apply these concepts for individual player interactions in baseball. It appears that the graphical structure of the pitcher-hitter matchup has yet to be used in current baseball research. I intend to complete the following steps for this project:

1. Data Collection and Preprocessing
    a. Determine timeframe to collect data from
    b. Create edge data set from Retrosheet play-by-play data
    c. Create pitcher and hitter node data sets from statistics on Baseball Savant
2. Model Selection
    a. Compare multiple GNN architecture such as GCN, GraphSAGE, and GAT
    b. Select one architecture to fully implement and tune
    c. Assess the importance of certain features
3. Evaluate Model
    a. Use cross-validation to assess accuracy
    b. Develop visualizations to help analyze the model's performance
    c. Assess calibration of prediction probabilities
4. Discussion
    a. Address the following questions
        i. Can a GNN effectively model pitcher-hitter interactions to improve at-bat outcome predictions?
        ii. What features contribute the most to the model?
        iii. How do historical matchups influence future matchups?
    b. Determine what further work can be done

Expected Outcomes and Deliverables

The goal of this project is to evaluate a graph neural network (GNN) model's ability to predict at-bat outcomes between MLB pitchers and hitters. The primary deliverable of this project will be a GNN model that is trained on historical MLB pitcher-hitter data. The model's prediction accuracy and computation time will be measured. Since this is a multi-class classification problem, both micro and macro average accuracy metrics as well as individual class accuracy metrics will be assessed. F1-score and ROC-AUC are two accuracy metrics that will be considered. Additionally, the calibration of the prediction probabilities will be assessed using metrics such as expected calibration error and Brier score. The computation time of the model will be measured to determine the scalability of the model. This model will only be trained using a few seasons worth of data, but in the future more data could be used to help improve the predictions. Additionally, this project deliverables will include an assessment of the features used and visualizations to assist in the interpretation of the model's decisions. Finally, this project will identify what future work can be done based on the results of the model.

References

[1] K. Koseler and M. Stephan, "Machine Learning Applications in Baseball: A Systematic Literature Review," *Applied Artificial Intelligence*, vol. 31, no. 9–10, pp. 745–763, Nov. 2017, doi: 10.1080/08839514.2018.1442991.

[2] R. Tracy, H. Xia, A. Rasla, Y.-F. Wang, and A. Singh, "Graph Encoding and Neural Network Approaches for Volleyball Analytics: From Game Outcome to Individual Play Predictions," 2023, *arXiv*. doi: 10.48550/ARXIV.2308.11142.

[3] P. Xenopoulos and C. Silva, "Graph neural networks to predict sports outcomes," in *2021 IEEE International Conference on Big Data (Big Data)*, Orlando, FL, USA: IEEE, Dec. 2021, pp. 1757–1763. doi: 10.1109/BigData52589.2021.9671833.

[4] J. Silver, "Singlearity: Using A Neural Network to Predict the Outcome of Plate Appearances," Baseball Prospectus. Accessed: Mar. 10, 2025. [Online]. Available: https://www.baseballprospectus.com/news/article/59993/singlearity-using-a-neural-network-to-predict-the-outcome-of-plate-appearances/

[5] J. Kim, T. Kim, S. Kim, and C. D. Yoo, "Edge-Labeling Graph Neural Network for Few-Shot Learning," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Jun. 2019, pp. 11–20. doi: 10.1109/CVPR.2019.00010.

[6] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, and M. Bronstein, "Temporal Graph Networks for Deep Learning on Dynamic Graphs," 2020, arXiv. doi: 10.48550/ARXIV.2006.10637.