LSTM-Based Baseball Pitch Prediction based on Motion Data

Introduction

Among the most appreciated pastimes is baseball. Many individuals engage in the sport, whether by being an avid participant or simply viewing the sport as a form of entertainment, and it has brought joy to many generations of enthusiasts. The game of baseball has complex rules and player positions, which it is essential for the game to run. Among the most vital roles is the pitcher, who dictates the game with their ability to throw the ball fast.

In baseball, recognizing the different pitch types: fastball, curveball, and slider is crucial and can change the outcome of the game. However, it remains difficult even among the most experienced individuals. By combining motion data and machine learning, this project seeks to create an automated system that accurately predicts the pitch types with LSTM, further bridging human intuition and data-driven precision.

Methods (Hypothesis)

Hypothesis - We can predict the type of pitch a pitcher will throw based on their pre-pitch motion.

General Approach

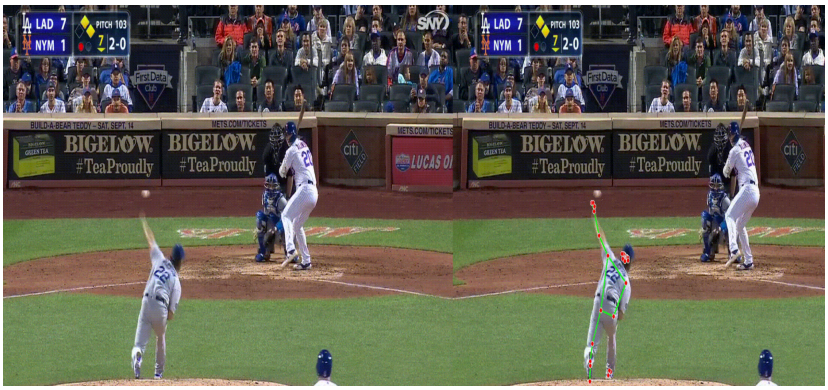
To test the hypothesis, a machine learning pipeline using Long Short-Term Memory (LSTM) networks was implemented using the Python library PyTorch. The process can be divided into four parts: data collection, preprocessing, feature extraction, and model training.

Data Collection

The dataset was manually created by collecting over 50 pitching videos for each type of pitch, for example, Fastball (FF), Slider (SL), and Curveball (CU). Each video contains the pitcher in full motion from the initial setup to the ball release.

Preprocessing and Pose Estimation

For each video, the pre-pitching motion segment was extracted to focus on the movements leading up to the throw. Then, utilizing the Python library MediaPipe, the positioning of major body joints was tracked frame-by-frame.



The figure above shows a side-by-side diagram of the pitcher's joint as keypoint coordinates and is then stored as a CSV file, forming a continuous dataset of the movements leading up to the ball throw.

Data Cleaning and Feature Preparations

Each CSV was cleaned with the following steps:

- Replaced missing or infinite values to avoid computational errors.
- Grouped joint coordinates by frames and then flattened them into arrays.
- Converted all arrays into PyTorch tensors for better model input.

- Automatically assigned labels to the corresponding pitch type - Fastball (FF = 0), Slider (SL = 1), and Curveball (CU = 2).

This process ensured that each data sample was numerically stable and formatted well.

Model Architecture and Training

The Long Short-Term Memory (LSTM) network was ultimately selected as it was effective in handling temporal data such as motion sequences. Each joint coordinate passes through the LSTM to learn the temporal dependencies between the intricate body movements. The model outputs a predicted class label that corresponds to the pitch type. The training process was performed with cross-entropy loss and the Adam optimizer in order to minimize classification errors.

Feature Importance

To determine which body regions play the most crucial role in predicting the pitch type, a conceptual ablation study was conducted. The whole body of the pitcher served as a baseline, each having a pose landmark in the pitching movement sequence.

First, we considered that removing the lower body landmarks (legs and hip) has minimal impact on the accuracy, thus indicating that it contains less discriminative information for this specific pitcher's pitch type differentiation.

On the other hand, removing the upper-body landmarks (shoulders, elbows, and wrists) would reduce the accuracy by a large margin, as these jointly encode the signatures of different pitch types in this pitcher's pitching movement.

By an ablation study it concludes that the upper-body landmarks of the pitchers are the most important feature for predicting this pitcher's pitching type, and furthermore it infers that we can extract the most relevant body part of the pitcher's movement. Thus, enhance model accuracy in a generalized model.

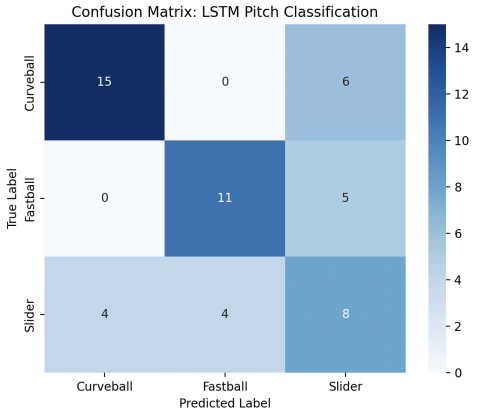
Evaluation

Post-training, the model was able to predict pitch type for unseen motion sequences. The overall performance was evaluated using metrics such as accuracy, confusion matrix, and bar graph visualization, which displayed accuracy in predicting Fastball, Slider, and Curveball motions.

Result (Test)

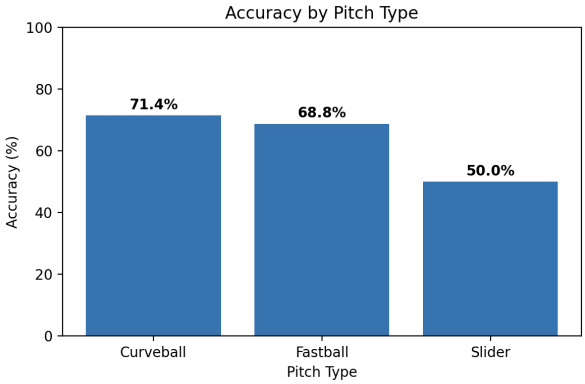
After training the LSTM model on the preprocessed motion dataset, it was then evaluated on unseen samples representing the three pitch types: Fastball (FF), Slider (SL), and Curveball (CU). The model overall predicted the pitch types with moderate accuracy, in many instances correctly identifying Fastballs and Curveballs; however, it showed some confusion between Sliders and Curveballs due to the similar nature of the throw.

A confusion matrix was generated to visualize the model’s accuracy across the three pitch types. The diagonal cells represent all correct predictions, and the off-diagonal cells indicate misclassifications.



As shown in the figure above, most Curveballs and Fastballs were correctly classified, as shown in the numbers placed diagonally. However, Sliders were misclassified as Curveballs, and vice versa, indicating the model's limitations in distinguishing the two pitching types. This may be attributed to the similar pre-pitch mechanics and spin patterns.

A bar chart of per-class accuracy was also generated to provide a more visually coherent representation of model performance across pitch types.



The figure above shows that Curveball leads in accuracy at 71.4%, the narrowest trailed by Fastball at 68.8%, and Slider at 50.0%. This suggests that the LSTM pipeline was more successful in identifying Curveballs and Fastballs, while Sliders are more difficult to distinguish - possibly due to the overlapping body motion characteristics or subtle trajectory differences before the pitch release.

Overall, the LSTM model was able to effectively capture the pitch type characteristics, capturing temporal dependencies in the movement sequences. Nonetheless, expanding the dataset, refining the motion extraction, or considering biomechanical parameters, such as arm angle and wrist rotation, could further enhance the accuracy of the model and potentially reduce confusion between highly similar pitches.

Discussion

The results overall confirm that the pre-pitch body motion contains recognizable patterns that correspond to different pitching types. The LSTM has successfully captured temporal dependencies in the motion data, allowing it to predict different pitch types with differing accuracy. However, certain limitations were still present during the development:

- Motion Similarity: Overlapping movements due to the similar nature of the pre-pitch motion lead to misclassification.
- Pose Estimation Noise: Variability in joint detection may have introduced data noise.

Moreover, the program could overall benefit from several enhancements that could improve accuracy and generalization. A major improvement would be increasing the dataset to include a more diverse variety of pitchers, camera angles, and environmental conditions would provide a more concrete representation of motion data and reduce overfitting to the specific pitching styles. Another enhancement that could be implemented is the incorporation of a 3D motion capture or optical flow feature could be a major improvement in spatial accuracy by capturing depth and velocity information that a 2D pose could fully estimate. Additionally, implementing advanced architectures such as attention-based LSTMs and Transformer models could improve the temporal understanding by having the network focus on key motion frames throughout the whole pitching sequence. Lastly, applying the data augmentation techniques, such as mirroring body poses or adjusting playback speeds, could increase the diversity within the dataset and robustness in the noise of the movements or recording conditions. Combined, these improvements would gain a more accurate and reliable system for predicting pitch types based on the pre-pitch movements.

Each Member's Contribution

This project was a collaborative effort in which each individual was assigned their own respective responsibilities in combining technical implementation, model design, and data preparation. Each member applied their existing expertise to make this project possible.

Davian Buana implemented the full pipeline for pose data preprocessing, LSTM model construction using the Python library PyTorch, and evaluated the model performance through visualization via confusion matrix and bar graphs.

Mateo Henriquez contributed to the design of the machine learning pipeline, mainly assisting in the selection of the LSTM architecture after considering other models, tuning model parameters, and evaluating strategies to improve classification performance.

Jung Hyun Park spearheaded the data collection and management, gathering the data from the pitching video data for Fastball, Slider, and Curveball categories. He ensured high-quality data and consistency during the preprocessing stage.

Through consistent collaboration and review, the team was able to successfully implement integrated data engineering, model design, and analysis into a compact framework for a baseball pitch prediction model.

References

- GeeksforGeeks (2019). What is LSTM, Long Short-Term Memory? [online] GeeksforGeeks. Available at: https://www.geeksforgeeks.org/deep-learning/deep-learning-introduction-to-long-short-term-memory/.
- Pytorch.org. (2024). Learning PyTorch with Examples — PyTorch Tutorials 2.7.0+cu126 documentation.[online] Available at: https://docs.pytorch.org/tutorials/beginner/pytorch_with_examples.html.

- mediapipe.readthedocs.io. (n.d.). layout: forward target: https://developers.google.com/mediapipe/ title: MediaPipe in Python parent: Getting Started has_children: true has_toc: false nav_order: 3 — MediaPipe v0.7.5 documentation. [online] Available at: https://mediapipe.readthedocs.io/en/latest/getting_started/python.html.
- Pytorch.org. (2024). Datasets & DataLoaders — PyTorch Tutorials 2.7.0+cu126 documentation. [online] Available at: https://docs.pytorch.org/tutorials/beginner/basics/data_tutorial.html.
- Lo, T.-C., Lee, C.-Y., Chen, C.-L., Hsieh, T.-Y., Chen, C.-H. and Lin, Y.-K. (2025). Application of Machine Learning Models for Baseball Outcome Prediction. Applied Sciences, [online] 15(13), pp.7081–7081. doi:https://doi.org/10.3390/app15137081.
- Pitching Dataset: https://drive.google.com/drive/folders/1hXhuYaFO7BRnTka2cpRQDer6cmQ8bMcg?usp=sharing (Please refer to this link for access)