Preparing for the Pitch: Using Deep Learning to Anticipate MLB's Toughest Pitchers

Evan Delcamp
Department of Statistics
University of Michigan
Ann Arbor, Michigan
edelcamp@umich.edu

***Abstract*—Major League Baseball (MLB) pitch sequencing is a complex, context-dependent process that elite pitchers use to outwit batters. This project investigates whether pitch type, location, and strike likelihood can be predicted using only in-game contextual data such as count, baserunners, and prior pitch information. Using a multi-task Long Short-Term Memory (LSTM) model trained on Statcast data from the top 20 starting pitchers by FanGraphs WAR across the 2023–2024 seasons, this paper predicts pitch type, custom pitch zone, and binary strike probability. The model incorporates pitcher-specific embeddings to handle variation in arsenals across players. While the prediction of pitch type and zone remains difficult, the model achieves a strike classification F1 score of 0.59, outperforming baseline heuristics. A pitch type prediction accuracy of approximately 45 % is also achieved, slightly exceeding the baseline of always guessing a pitcher's most common pitch. This work demonstrates the feasibility of modeling sequencing tendencies even against MLB's best pitchers and lays the groundwork for future models that incorporate batter characteristics, pitch movement, and game-theoretic elements.**

***Index Terms*—MLB, pitch sequencing, LSTM, deep learning, Statcast, multi-task learning**

I. Introduction

Many consider hitting a round baseball traveling 44.7 m/s (100 mph) with a round bat to be one of the toughest things to do in professional sports. Over time, this task has consistently become more difficult, as pitchers have gained more access to data on batter's weak points, learned to throw harder and with more movement. At the major league level, some pitchers are just about impossible to beat. Hitters are often forced to guess which pitch is coming in order to hit well, because they must alter their swing based on pitch location, speed, and movement. Machine learning, specifically an LSTM sequencing model that predicts the location, pitch type, and whether a pitch will be a strike, could potentially help with this guessing process for these hard-to-hit pitchers. For real world use, this model could allow teams to give a "cheat-sheet" to their coaches about what pitch is coming, which could be relayed to the hitter during the game through game-legal hand signals. Specifically, the insights this paper hopes to assess are:

1) To what extent can pitch type, location, and strike likelihood be predicted for elite pitchers using game context?
2) Can a multi-task LSTM model simultaneously predict pitch type, location, and strike probability to capture underlying sequencing patterns?

There have been some modeling studies done on pitch sequence previously. For the 2021 MIT Sloan Sports Analytics conference, *Decoding MLB Pitch Sequencing Strategies via Directed Graph Embeddings*, Prasad (2021) used a quantitative approach to pitch sequencing with directed graph network embeddings, and the paper found evidence that pitchers use "set-up" and "knock-out" pitches. A "set-up" pitch is to get the batter to expect something in particular, and then the "knock-out" pitch is there to get the batter to swing and miss. This paper also paid attention to whether pitch sequence led to pitcher success, and found that it did not guarantee success in a vacuum - pitchers with similar sequences had different results.

Another paper, *Pitch Sequence Complexity and Long-Term Pitcher Performance*, Bock (2015) built classifier models using multinomial logistic regression and support vector machine (SVM) algorithms to predict pitch sequences. Similar to this paper's intent, Bock's model built these predictions using the current game state. This paper impressively returned a 74.5 % ability to predict the next pitch.

II. Method

The inputs for this LSTM model are sequential pitch data, with information about prior pitch types and locations, and current game state.

The outputs of this model are pitch type (multiclass), pitch zone (multiclass), and whether or not the pitch was in the strike zone (binary).

Predicting any of these three things could be very useful for hitters, as they choose whether and how to swing at a pitch.

The dataset used was custom built from Statcast data using the pybaseball module. The dataset was filtered to the top pitchers using the measure Wins Above Replacement (WAR) from the baseball website Fangraphs, and data for those pitchers was collected over the course of the 2023-2024 seasons. Only at bats that went at least two pitches were considered.

Pitch types such as fastball, curveball, etc. that were thrown by those pitchers were indexed. In order to be considered a pitch the model could predict, the pitcher must have thrown that type of pitch at least 2.5 % of the time over the 2023-2024 seasons. The strike zone was broken into five custom zones in order to model how hitters think in real life. The zones were top left, top right, middle, bottom left, and bottom right.

The model used CrossEntropyLoss for Pitch Type and Zone, and used BCEWithLogitsLoss for the binary strike classification.

Loss curves were tracked for training and validation data in order to choose the model with the least validation loss. The model was run for 200 Epochs with a batch size of 32, and shuffle was on for training data, but off for validation data. Loss was calculated by adding the loss for each of the three outputs.

III. Results

The model achieved its lowest validation data loss at epoch 16 (Fig. 1).

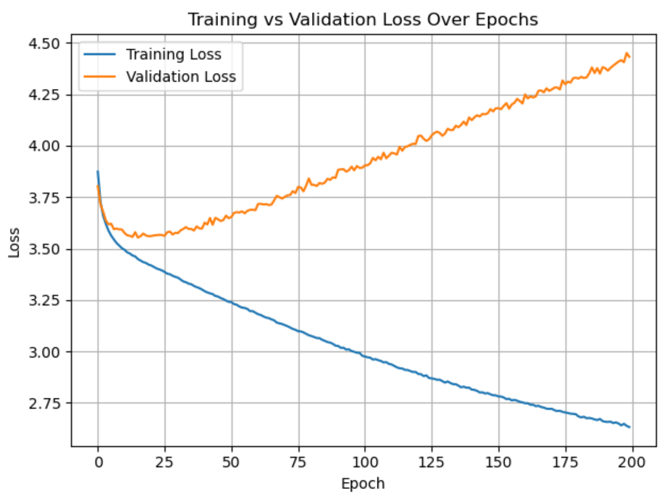

Fig. 1. Training vs Validation Loss over 200 Epochs. The model shows clear overfitting as validation loss increases while training loss decreases.

Analysis of the models sigmoid output distribution (Fig. 2) for strike probability led to choosing a classification threshold of 0.525. Predicting a ball as a strike and a strike as a ball both have similarly negative effects on the hitter when game state is not considered, so a 0.525 threshold also makes sense for this model to keep it balanced.

The confusion matrix for the pitch type prediction shows a general difficulty predicting pitches, but it definitely shows some promise (Fig. 3.). The model struggled to decide between pitches that are used similarly by pitchers. For example, Four-Seam Fastballs (FF) and Cutters (FC) are both high velocity pitches that are used similarly, and the model struggled to differentiate between them, predicting 64 Four-Seam Fastballs that were actually cutters. In general, the model may be over-predicting Four-Seam Fastballs, as evidenced by predicting over 1,600 Fastballs from the training data.

[2]Pitch type abbreviations: **CH** = Changeup, **CU** = Curveball, **FC** = Cutter, **FF** = Four-seam Fastball, **FS** = Splitter, **KC** = Knuckle Curve, **SI** = Sinker, **SL** = Slider, **ST** = Sweeper, **SV** = Slurve.

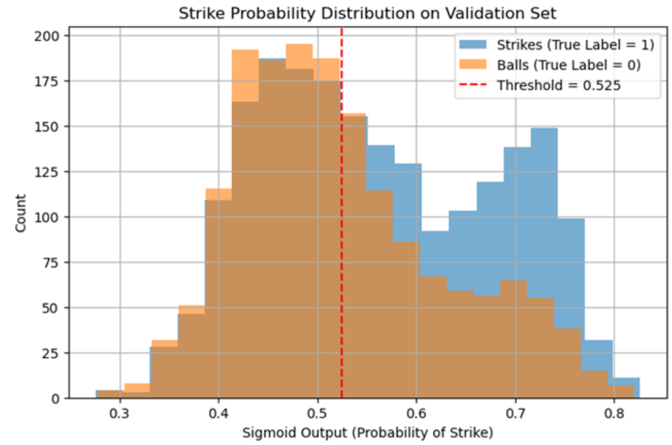

Fig. 2. Distribution of predicted strike probabilities on the validation set. The model's sigmoid outputs show a meaningful separation between true strikes and balls, with a chosen threshold of 0.525 (dashed red line) balancing precision and recall.

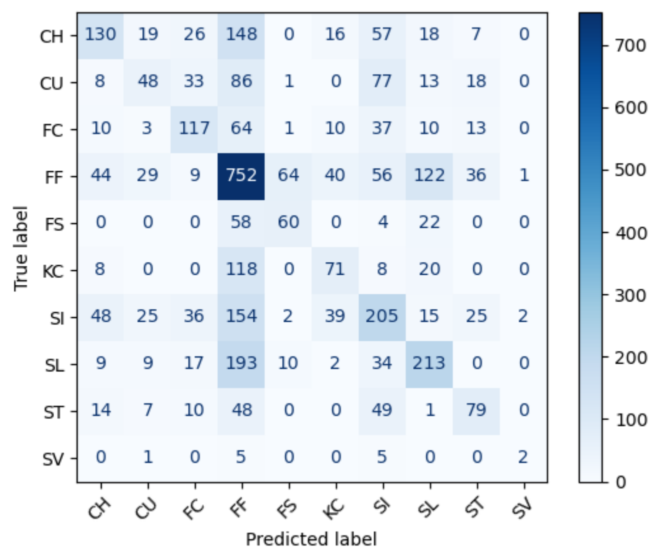

Fig. 3. Confusion matrix for pitch type prediction on the validation set.[2] While four-seam fastballs (FF) are classified with high accuracy, other pitch types such as splitters (FS) are often misclassified as fastballs, highlighting overlap in features.

The confusion matrix for the pitch zone showed a particular difficulty with pitches high (Fig. 4.). This may be related to the fact that a standard size was used for the middle zone. In baseball, the strike zone does not change horizontally, but it changes vertically depending on the height of the batter. So the middle zone for a shorter hitter could end up being a very large portion of their strike zone, complicating patterns.

Overall, the model showed some promise, but needs a lot of fine tuning to be useful in real life. The pitch type accuracy of 44.71 % is only slightly more accurate than always guessing each pitcher's most common pitch, which would return 41.5 % accuracy for the dataset used. Table 1 shows accuracy for pitch types, zones, and the strike outputs, as well as F1 scores

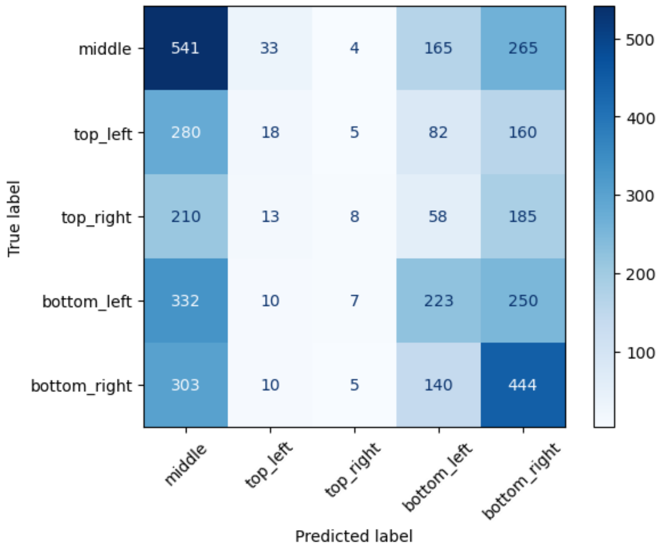

Fig. 4. Confusion matrix showing predicted versus true pitch locations across five custom-defined strike zone regions: middle, top left, top right, bottom left, and bottom right. Top left and top right were rarely predicted

for the strike output.

TABLE I
VALIDATION PERFORMANCE OF MULTI-TASK LSTM MODEL

Metric	**Score**
Strike F1 Score	0.59
Ball F1 Score	0.55
Strike Accuracy	56.81%
Pitch Type Accuracy	44.71%
Zone Accuracy	32.90%

IV. CONCLUSION

This paper explored the use of an LSTM deep learning model to predict pitch sequences of elite MLB pitchers. Despite the complexity of this task, the model did show some success, predicting the correct pitch type 44.71 % of the time versus a 41.5 % accuracy from just guessing the pitchers most common pitch every time. Predicting ball or strike also proved difficult, as the 56.81 % accuracy is only a marginal difference from the 53 % of strikes across all pitches in the dataset. Zone predicting was not very successful in this model, as the model did not predict the upper zones very often, so a more dynamic middle zone could potentially improve the modeling of zone. However, it is certainly possible that since even professional pitchers cannot always throw the ball exactly where they want it, this prediction could be too noisy.

The fact that the model leaves a lot of room for improvement also means the success could potentially be improved dramatically. The similarity of pitch types and difficulty for the model to distinguish between similar pitch types makes pitch type an ideal candidate for embedding, to better capture those similarities.

Potentially, more features could also be added based on batter tendencies. Often, pitchers are going after batter weaknesses, and this was not incorporated in this model, other than whether the batter and pitcher were the same handedness. Another important factor could be pitch velocity, since pitchers try to change speeds to throw off the hitters timing. Additional modeling variables that could be adjusted and tested further include the overall loss function if it is determined that the different outputs should be weighted differently, and working elements of team specific pitching strategies into the model using features like pitching coach and catcher.

Overall, this paper demonstrates that there is potential for a multi-task LSTM model to predict pitch type, location, and strike probability to capture underlying sequencing patterns. However, a lot more investigation is required.

ACKNOWLEDGMENT

The author would like to acknowledge the assistance of OpenAI's ChatGPT for troubleshooting support during model development and debugging.

REFERENCES

[1] A. Prasad, "Decoding MLB pitch sequencing strategies via directed graph embeddings," *MIT Sloan Sports Analytics Conference*, 2021. https://www.sloansportsconference.com/research-papers/decoding-mlb-pitch-sequencing-strategies-via-directed-graph-embeddings

[2] J. R. Bock, "Pitch sequence complexity and long-term pitcher performance," *Journal of Sports Analytics*, vol. 1, no. 1, pp. 40–55, 2015. https://www.mdpi.com/2075-4663/3/1/40

[3] J. LeDoux *et al.*, "pybaseball: Python interface for MLB Statcast data," 2016–2024. https://github.com/jldbc/pybaseball

[4] MLB Statcast, "Statcast search and tracking data," Major League Baseball. https://baseballsavant.mlb.com/

APPENDIX A
PITCHERS INCLUDED IN THE MODEL

The following 20 MLB starting pitchers, selected based on cumulative FanGraphs WAR across the 2023–2024 seasons, were used in model training and evaluation:

- Zack Wheeler
- Logan Webb
- Sonny Gray
- Tarik Skubal
- Chris Sale
- George Kirby
- Dylan Cease
- Zac Gallen
- Kevin Gausman
- Framber Valdez
- Pablo López
- Justin Steele
- Zach Eflin
- Seth Lugo
- Blake Snell
- Gerrit Cole
- Corbin Burnes
- Logan Gilbert
- Cole Ragans
- Aaron Nola