Machine Learning Predictive Analytics for Player Movement Prediction in NBA: Applications, Opportunities, and Challenges

Dembe Koi Stephanos
East Tennessee State University
Johnson City, Tennessee, USA
stephanos@etsu.edu

Ghaith Husari
East Tennessee State University
Johnson City, Tennessee, USA
husari@etsu.edu

Brian T. Bennett
East Tennessee State University
Johnson City, Tennessee, USA
bennetbt@etsu.edu

Emma Stephanos
Clemson University
Clemson, South Carolina, USA
egrosse@clemson.edu

ABSTRACT

Recently, strategies of National Basketball Association (NBA) teams have evolved with the skillsets of players and the emergence of advanced analytics. This has led to a more free-flowing game in which traditional positions and play calls have been replaced with player archetypes and read-and-react offensives that operate off a variety of isolated actions. The introduction of position tracking technology by SportVU has aided the analysis of these patterns by offering a vast dataset of on-court behavior. There have been numerous attempts to identify and classify patterns by evaluating the outcomes of offensive and defensive strategies associated with actions within this dataset, a job currently done manually by reviewing game tape. Some of these classification attempts have used supervised techniques that begin with labeled sets of plays and feature sets to automate the detection of future cases. Increasingly, however, deep learning approaches such as convolutional neural networks have been used in conjunction with player trajectory images generated from positional data. This enables classification to occur in a bottom-up manner, potentially discerning unexpected patterns. Others have shifted focus from classification, instead using this positional data to evaluate the success of a given possession based on spatial factors such as defender proximity and player factors such as role or skillset. While play/action detection, classification and analysis have each been addressed in literature, a comprehensive approach that accounts for modern trends is still lacking. In this paper, we discuss various approaches to action detection and analysis and ultimately propose an outline for a deep learning approach of identification and analysis resulting in a queryable dataset complete with shot evaluations, thus combining multiple contributions into a serviceable tool capable of assisting and automating much of the work currently done by NBA professionals.

ACMSE 2021, April 15–17, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8068-3/21/04…$15.00
https://doi.org/10.1145/3409334.3452064

CCS CONCEPTS

• **Information systems → Data mining**; • **Computing methodologies → Machine learning algorithms**.

KEYWORDS

Predictive Analytics, Machine Learning, Clustering, Action Recognition, NBA Video Analysis, Data Mining, Survey

ACM Reference Format:
Dembe Koi Stephanos, Ghaith Husari, Brian T. Bennett, and Emma Stephanos. 2021. Machine Learning Predictive Analytics for Player Movement Prediction in NBA: Applications, Opportunities, and Challenges. In *2021 ACM Southeast Conference (ACMSE 2021), April 15–17, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3409334.3452064

1 INTRODUCTION

An interesting recent application of machine learning is the study of strategies and patterns in major sports. Since 2012, when SportVU first started capturing player location data during National Basketball Association (NBA) games, there have been myriad pursuits to parse this raw data and extract meaningful features about players and the game itself. Professional basketball is a fast-paced, complex system that is continuously evolving, making high-level features challenging to obtain. As a result, most work identifying and analyzing individual plays or actions within a game is still performed by humans watching game tape. Several insightful approaches have been used in attempts to remedy this by using modern machine learning tools like support vector machine (SVM) classifiers, neural networks, and clustering algorithms to automate these time-consuming tasks [10, 12, 14, 19, 21, 23]. Many of these approaches seek to limit the problem domain by focusing on a particular play or aspect of the game, but a coherent end-to-end system that parses raw data and provides AI-driven recommendations has yet to be fully realized. In this paper, we will examine and critique some of the more effective approaches to pattern mining NBA player movement data, focusing primarily on the most commonly researched play action in literature: the pick-and-roll. We ultimately propose a design for an analytic pipeline capable of play classification, pattern detection, and quality evaluation with actionable recommendations.

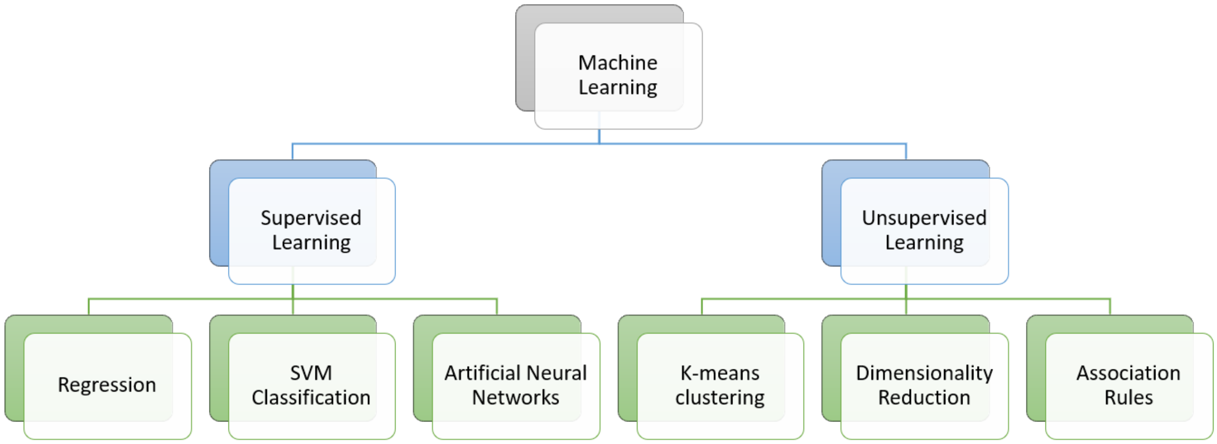


Figure 1: Taxonomy of Machine Learning Approaches Applied in the Domain

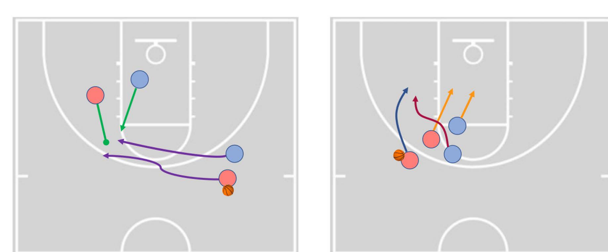
2 BACKGROUND

Basketball is a team sport in which two teams of five active players compete in timed possessions to amass as many points as possible by scoring, or getting the basketball into their goal, a circular rim positioned 10ft in the air with a glass backboard. Several restrictions are placed on players, who must dribble or bounce the ball when moving and may not leave the bounds of the rectangular court. In the NBA, games involve as many as fifteen players per team, with twenty-four seconds possessions spanned out over four, twelve-minute quarters. Traditionally, a variety of plays, coordinated actions by a team involving multiple passes and player movements, are performed in sequence with the objective of creating valuable scoring opportunities. Recently, these strategies have significantly evolved from long-scripted plays, to read-and-react offenses in which a variety of independent actions are performed by players who then read the defensive response and react accordingly.

2.1 Key Terms

First, we must define key terms pertinent to our discussion. Perhaps most importantly are two key terms we have already used: **play** and **action**. For our purposes, we will define a *play* as a strategic and intentional sequence of actions taken out by cooperating offensive players in an effort to create the space required for a valuable scoring attempt and an *action* as a discrete interaction between 2+ offensive players and their corresponding defenders. Both the approach and execution stages may be executed in a number of variants. Since only one player may be in possession of the ball at a given time, many of these plays and actions involve what is called a *screen*, in which an offensive player positions themselves firmly between a defending player and the ball-handler, forcing that defending player to navigate around them, subsequently creating space for the non-screening offensive player.

There are many versions of screens, each with several additional variants, but perhaps the most common is the on-ball screen. An on-ball screen occurs when an offensive player sets a screen for the ball-handler, and the ball-handler then guides their defender into the screen. This can be broken in to two stages: the approach and the execution, Figure 2 illustrates the *approach* and *execution* stages.



(a) Approach Stage (b) Execution Stage

Figure 2: An Illustration of an On-ball-screen Broken into Approach and Execution Stages [19]

The on-ball screen is prominent in many plays, and can even be considered a play itself, often referred to by one of its most common variants, the pick-n-roll. The on-ball screen is a favorite in the machine learning literature [10, 11, 19, 23], as it is a frequently occurring action with an easily identifiable set up and great variety of possible executions. As a result of the focus of the literature, we will also focus our discussion towards this play action.

The remainder of this paper is organized as follows. In Section 3, we discuss the dataset and the technology used to track and collect positioning data of the ball and players during a game. A systematic and detailed review of state-of-the-art studies on the different types of machine learning systems applied and proposed in this domain is given in Section 4. In Section 5, we present the current challenges and promising opportunities regarding the use of machine learning for NBA predictive analytics. A conclusion is drawn in Section 6.

3 DATA

In 2012, the NBA began tracking player movements during games by installing a total of six cameras that capture the location of all ten players on the court, as well as the ball and three referees, twenty-five times per second. This data was publicly available through the

2015-2016 season and inspired a great deal of interest among the sport's analytic community, eventually making its way to machine learning experts [1]. The data itself is collected in a JSON object, typically around 600 MB in size, containing a variety of defining information such as the teams and players involved, the date and location of the game and the coordinates of the players broken up by quarter and event. Figure 3 illustrates a detailed breakdown of the JSON object.

An event occurs when one team takes possession of the ball and ends either when they score, a foul or turnover is committed or in other unique circumstances such as clock malfunction. Since the raw positional data of the players does not paint a full picture, this dataset is often combined with other event specific data, such as the type of shot taken or other play specific details that allow for a more cohesive view of on court behaviors. Many techniques are used to abstract this low-level data, but most include the embedding of the coordinates into vectors or trajectory images that represent a player's full range of motion over the course of a given possession.

4 A REVIEW OF MACHINE LEARNING APPROACHES FOR NBA ANALYTICS

In this section, we create two broad categories to review significant research published on machine learning-based NBA analytics. These two categories are (1) supervised (classification) approaches and (2) unsupervised (including clustering) approaches. In the case of supervised machine learning, a human researcher chooses an event (e.g., an action) that occurs in the data, and then a classification model is built to automatically find all occurrences of that event in the dataset. Unsupervised machine learning, on the other hand, examines the naturally occurring patterns within the data, and analyzes these patterns and their frequencies to potentially reveal new knowledge, groupings, or strategies in the data. While these approaches are different, they are often used together to identify actions and examine general patterns. The key points of each publication are discussed in the following subsections.

4.1 Supervised Learning

Supervised learning infers a function from labeled training data to create knowledge structures that support the process of classifying new instances into a set of predefined classes [16]. The input for this type of learning algorithms is a collection of sample instances that are pre-classified (labeled) into a predefined set of classes. The output of this process is a classification model that is constructed by analyzing the training data and producing an inferred function that can determine the class (label) for unseen instances with a reasonable accuracy.

There are two key steps in supervised learning:

- **Training:** Analyzes the training data (labeled instances) and constructs a classification model.
- **Testing:** Uses the constructed (trained) model to classify new instances and reports the accuracy.

There are various supervised learning algorithms that differ in their approach to analyzing, inferring, and generalizing knowledge from the labeled training data to construct the classification model.

Due to their top-down nature, supervised learning efforts in NBA analytics, are often more focused on identifying and labeling certain events in the dataset. Specifically, in the case of the on-ball-screen, research works in this area formulate and utilize a set of rules around the distances and durations of time that coordinating offensive players and the ball spend in proximity to each other. Approaches that use a sliding window to extract features of a certain event are frequently used to ensure the capturing of all actions, as it is very difficult to specify the point in a given possession at which an action has started [6]. Using a sliding window, the screen moment can be specified by identifying the moment right before the players begin moving again [11].

Yu and Chung [23] propose an SVM binary classifier to automatically identify on-ball screens. First, they propose an algorithm to identify event candidates that may contain on-ball screens. This algorithm uses a set of rules that are devised manually to extract on-ball screen candidates and filter out other events. Then, these candidates are analyzed manually (by humans) to determine and label each as on-ball screen or otherwise. Once the labeled dataset (actions and their labels) is created, the set is split into two parts: a training set, and a testing set. This split is often a 9:1, where 90% of the data is used for training and 10% is used for testing. The SVM binary classifier was trained using four distance- and speed-based player features. After the training phase, the testing set was fed to the classifier to evaluate its performance to identify on-ball screens and it achieved a recall of 90.46%.

McQueen and Guttag [11] propose an approach that utilizes a set of rules to identify actions. The data are then segmented into periods of time around these actions. Using this approach, they extracted 30 continuous features from each action. Once these features were extracted, they were discretized into five binary features based on quintiles. This resulted in a 150-dimensional feature vector for every action. The data were then labeled and split into approximately 51.9% training and 48.1% validation. Then, an SVM linear classifier was constructed using the training data. This process was repeated 3,200 times using different splits of data. The classifier achieved a recall of 82% and precision of 80%.

McIntyre et al. [10] use a more robust dataset consisting of 270,823 ball screens. The labeled dataset was split into 70% training and 30% testing sets. Four classifiers were trained to identify four different types of ball screens. These classifiers identified 270,723 screens in total, achieving a high recall of 83% and precision of 78% in identifying the type of on-ball screens where the defender stays between the ball handler and the screener.

Another approach proposed by Kates [6] uses the SportVU data to detect six different plays, where each play has 20 labeled instances at minimum. The approach trained six SVM classifiers (one for each play type) using 8:2 stratified data split. The classifiers predicted the plays with a collective accuracy of 72.6% and an F-score of 72.7%.

Artificial Neural Networks (ANN), is a learning algorithm that is based on a set of connected nodes. Each connection between two nodes communicates a signal to the destination node. This signal is a real number and the output of each node is calculated by a non-linear function that aggregates the node's inputs in a processes

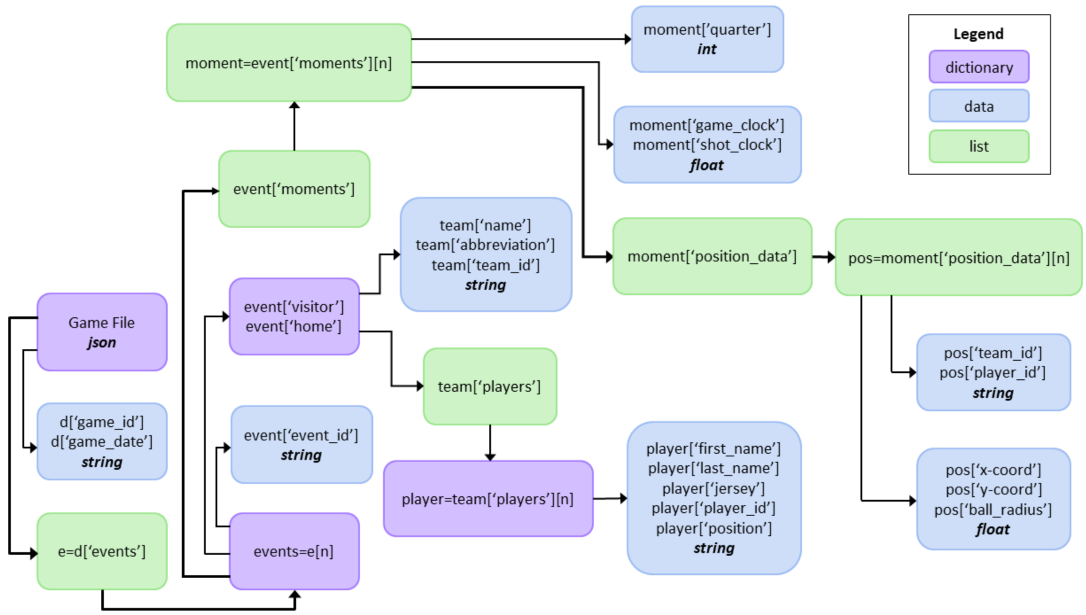


Figure 3: A Visual Breakdown of a JSON Object Representing a Single Game Capture from the SportVU Dataset [1]

that is loosely based on the biological neural networks in the human Brain. Figure 4 illustrates a collection of nodes communicating together in a two-layer ANN. Wang and Zemel [21] use variants of neural networks to automatically classify play sequences to 11 selected play classes (or offense strategies). Recurrent Neural Networks (RNNs) achieved the highest accuracy , top-3 accuracy of 80%, in classifying 95 unlabeled sequences (approximately 6% of the data) into 11 possible play classes. Understandably, the simple neural networks achieved a lower top-3 accuracy of 77% when compared with RNN. This is due to the ability of RNN to better handle and learn from sequential data of variable length as it accumulates change over time.

A current limitation of this practice is researchers being responsible for creating the labeled dataset. While this is likely okay for more explicit play actions like the on-ball-screen, it introduces a level of bias, is less suited for more subtle or nuanced actions, and would be better performed by a domain expert [23]. These supervised learning approaches work well for the base task of identification and classification and are thus an excellent first step toward a more robust pipeline capable of providing real analytic insights. Despite the high accuracy, an on-ball-screen dataset alone cannot provide great value to the problem domain. This is where unsupervised approaches can be more effective and provide more insightful analysis. They can take this refined data as inputs and produce interesting, novel insights into the patterns and strategies present in the player movement, narrowing those patterns to specific variations in both the approach and execution of the on-ball-screens identified by the supervised approach.

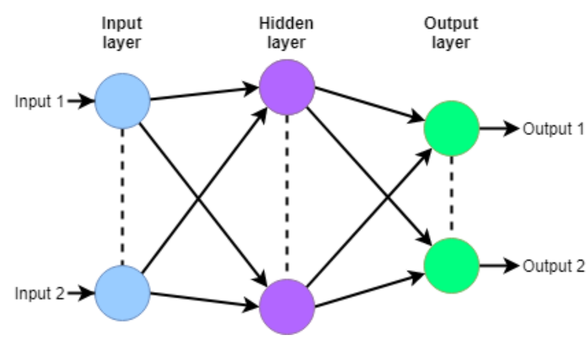


Figure 4: An Illustration of a Two-layer Artificial Neural Network

4.2 Unsupervised Learning

Unsupervised learning (or clustering) is a type of machine learning that allows the model to work with minimal or no human supervision by finding the natural clusters (groups) in the data using heuristics without reference to labeled outcomes by humans [4].

Clustering focuses on finding patterns in the data and then creates groups of instances with similar properties. This similarity is calculated by a distance function, such as Euclidean distance. Depending on the approach, these clusters can be exclusive, in which instances belong to only one cluster, or they can be overlapping, where an instance could belong to more than one cluster. Another type of clustering is probabilistic clustering where an instance belongs to a cluster with a certain probability. Hierarchical clusters occur when there are parent clusters at the top level, and each of these parent clusters is further refined to smaller, more specific clusters [22].

There are three types of clustering methods:

- **K-means algorithm:** Forms clusters in numeric domains, grouping instances into exclusive (disjoint) clusters. Algorithm 1 shows the pseudo code of K-means clustering.
- **Incremental clustering:** Creates a hierarchical grouping of instances.
- **Probability-based methods:** Assign an instance to a cluster based on a probability score for membership.

Algorithm 1: K-means Clustering [9]

Input: K, and unlabeled data set $\{x_1, \ldots, x_N\}$.
Output: Cluster centers $\{s_k\}$ and the assignment of the data points $\{\pi_{nk}\}$

```
Randomly initialize {s_k}
repeat
    for n := 1 to N do
        for k := 1 to K do
            if k = argmin_i ||s_i - x_i||^2 then
                π_nk := 1
            else
                π_nk := 0
            end
        end
    end
    for k := 1 to K do
        s_k := (Σ_n x_n π_nk) / (Σ_n π_nk)
    end
until {π_nk} or {s_k} don't change;
```

Unsupervised learning is becoming increasingly popular among sports scientists. Research works that utilize unsupervised learning methods are more geared towards pattern extraction and strategy analysis than supervised efforts. However, they often leverage the classification potential of supervised models to do so.

Nistala and Guttag [14] and Nistala [13] utilize cluster analysis to assign similar attacking movements into groups, such as movements along sidelines, run along the baseline, and other attack movements. First, they constructed 3 million trajectory images from NBA player tracking data collected by the STATS SportVU player tracking system [1]. They then utilized a numerical abstraction algorithm for player movements and ran the K-means clustering algorithm on the 3 million trajectory images to group similar trajectories together. The algorithm grouped the attacking player movements into 20 clusters. Manual evaluation of these clusters (by selecting case studies from each one) showed that deep learning can be used to learn patterns of basketball attack movements.

Brooks [2] proposes an approach utilizing unsupervised machine learning to characterize patterns of play for NBA teams on offense. First, the approach builds an image for each player's movement on offense to give a starting point for comparing player movements across different possessions. They then utilized the k-means clustering algorithm using the Python package scikit-learn [15] to cluster 60,000 instances of possessions. The number of clusters was set to k = 30, as it represented the "knee" in the response curve between k and the average within-cluster distance. This process produced 30 clusters each containing around 2,000 instances. The 30 resulting clusters were evaluated manually (using human judgement) by checking the top 10 instances that are closest to the center of the cluster. This evaluation confirmed that instances in the same cluster had similar patterns of movement for the players and the ball. Each cluster was then given a description that describes the movement pattern (or play) of its instances. Table 1 shows a sample of 5 clusters with their description.

Lutz et al. [8] proposes an approach that uses statistics such as field goals and assists to cluster similar players into 10 categories. Franks et al. [5] utilizes non-negative matrix factorization (NMF) to cluster defense players based on the locations of field goals.

Sampaio et al. [18] clustered players with similar features such as attacking, defense and passing statistics. Sampaio et al. [17] and Teramoto et al. [20] applied dimensionality reduction techniques such as principal components analysis (PCA) in basketball. Algorithm 3 shows a pseudo code for PCA.

5 CHALLENGES AND OPPORTUNITIES

Since 2005, many NBA video analytic research works have been proposed that have motivated and impacted the domain. In this section, we summarize the key challenges identified in the literature and highlight proposed solutions and directions for future research. We divide the challenges and proposed direction into three categories: (1) feature engineering and extraction, (2) deeper information discovery of actions, and (3) effective video content preprocessing and noise filtration.

5.1 Feature Engineering and Extraction

Machine learning solutions developed for traditional video- and image-based processing are not effective to provide NBA analytics using NBA video tapes. The main reason behind this is that traditional video-based solutions aim to detect the variations of three main features: shape, color, and position. However, solutions for video-based NBA analytics aim to process videos by detecting multiple human actions executed simultaneously. Such data may have extremely similar instances in terms of shape, color, and position. However, effective features are generated by successfully detecting various player actions.

A promising approach to address this challenge is to generate robust and more comprehensive features that analyze the context of human movement dynamics. Extracting and utilizing multiple detailed metrics and labels from videos is more appropriate than a single description [3].

5.2 Deeper Information Discovery of Actions

In NBA basketball, the same play can be executed in multiple variations. This means using traditional metrics and distance functions (e.g., Euclidean distance) will not be sufficient to simultaneously recognize similar plays and differentiate between different ones.

Li et al. [7] showed that processing video frames within only a small temporal region is not effective for recognizing actions in videos. Moreover, they discovered that long-range dynamic information and deep features are essential for the discrimination of complex activities with shared sub-actions. This is a promising approach to adopt in NBA action and play recognition as it provides

Table 1: Resulting Play-groups Using Cluster Analysis

Cluster	Description
1	On-ball screen, pass screener or far side corner
2	Screen to the right of the top of they key, pass to the right corner
3	On-ball screen on right elbow, drive or pass to the center
4	left side post-up
5	Pick-and-roll at the top of the key

more detailed metrics that allow for more accurate similarity and dissimilarity estimations for actions and plays.

5.3 Effective Video Preprocessing and Noise Filtration

The nature of the data in this domain is unbalanced such that most of the game might not have any plays or actions of interest being executed. For example, Yu [23] discovered that only 2 minutes of on-ball screens (each taking about 1.3 seconds) occur in the entire 48 minutes of a standard game. This imbalance between events of interest and other events results in a ratio of approximately 4% relevant data. Without innovative and effective filtering of irrelevant data, the difficulty in identifying events of interest will continue to increase.

Algorithms that utilize rules designed manually by sport experts remain the sole method of filtering irrelevant content. Algorithm 2 shows an example of a rule that finds the ball handler (if any). If the algorithm returns NULL, then the frame can be discarded as no action (or play) is executed without the ball. Sampaio et al. [17] and Teramoto et al. [20] proposed using Principle Component Analysis (PCA) for data reduction. Also, Nistala and Guttag [14] proposed using Convolutional Neural Network (CNN) to represent (approximate) the data in an abstract manner which not only reduces the size of the data but also allows for more relaxed movement comparisons among actions and plays.

Algorithm 2: Finding the Ball-handler

find ball-handler(video frame)
BH ← getmin(distance between Ball and players)
if *distance between BH and Ball* < 5 *ft.* **then**
    return H
else
    Return NULL
end

Annotation Scalability. Another challenge in this domain is the difficulty of obtaining a larger number of labeled events. This is due to the fact that an annotator needs to watch a tape to judge when a certain play begins and ends and classify what the play was. In addition to being a time- and labor-consuming approach, the beginning and end times of a certain action or play could be debatable, which impacts the effectiveness of machine learning approaches.

Despite the many research works and proposed solutions for action and play recognition in NBA data, many problems remain due to the complexity of recognizing and differentiating between actions and sub-actions of basketball plays. It is still challenging to develop a unified framework with a machine learning pipeline that can filter, abstract, and label actions and plays in an accurate manner. Engineered features that take into account contextual information about advantage creation, such as space created, defensive switches, and other player skillsets are needed to train more accurate models. Also, utilizing and tailoring deep features to distill complex actions and strategies into sub-actions and sub-strategies is important for comparing actions with more accurate measures (i.e., fine-grain distance functions), such that those types of complex questions regarding on-ball screens can be answered. This pipeline would consist of supervised learning techniques to classify the on-ball screen dataset, a neural network and clustering approach to group the various approach and execution pairings present in that dataset, and a quality evaluation tool that takes into account features of advantage creation like space created or defensive switches favoring the offense and player skillsets.

Algorithm 3: Principle Component Analysis [9]

Input: L, and input vectors of an unlabeled or labeled data set $\{x_1, \ldots, x_N\}$.
Output: The projected data set $\{x_1, \ldots, x_N\}$, and basis vectors $\{w_j\}$, which form the principal subspace
$\bar{x} := \frac{1}{N}\sum_n x_n$
$S := \frac{1}{N}\sum_n (x_n - \bar{x})(x_n - \bar{x})^T$
$\{w_j\}$:= the L eigenvectors of S corresponding to the L largest eigenvalues
for $n := 1$ *to* N **do**
    for $j := 1$ *to* L **do**
        $z_{nj} := (x_n - \bar{x})^T w_j$
    end
end

6 CONCLUSION

In this paper, we surveyed approaches to NBA content analysis, the types of machine learning algorithms used to classify and extract patterns from player movement data, the challenges with feature engineering and noise filtration present in these pipelines, and the current solutions implemented in this domain. Ultimately, play-action detection, classification, and analysis are mostly still in the early stages. While great strides have been made in assembling machine learning strategies to perform these tasks, the resulting work is mostly too limited to truly represent the intricate system that is a basketball game. By limiting our domain, we can begin to see how such an overall system might be assembled. Identifying and classifying basic actions is nearly a solved problem, and advances in deep learning show great promise for pattern mining such a complex system. The domain-specific quality evaluation has also progressed to better represent the intent of specific actions that are not directly tied to an outcome.

We believe there is a great deal of room to build out this pipeline specifically for on-ball screens and to eventually branch out to other less prominent actions. Such a comprehensive pipeline would take in the raw game data as a JSON object, preprocess and select candidates from the coordinate data, classify those results and perform pattern extraction, and evaluate the created advantage among the various detected strategies. Once similar pipelines can be constructed to analyze defensive and offensive strategies present among all ten players on the court, it may even be possible to completely automate the tedious work currently performed by staff videographers and coaches.

ACKNOWLEDGMENTS

This work was supported in part by the Niswonger Research Fellowship in Computer Science. The authors would like to thank the Niswonger Foundation for the support of this research. We would also like to thank our anonymous reviewers for their useful comments.

REFERENCES

[1] [n.d.]. STATS SportVU® Basketball Player Tracking. https://www.stats.com/sportvu-basketball/.
[2] J. D. Brooks. 2018. *Using Machine Learning to Derive Insights from Sports Location Data.* Ph.D. Dissertation. Massachusetts Institute of Technology.
[3] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. 2014. Decaf: A Deep Convolutional Activation Feature for Generic Visual Recognition. In *International Conference on Machine Learning*. Beijing, China, 647–655.
[4] D. H. Fisher, M. J. Pazzani, and P. Langley. 2014. *Concept Formation: Knowledge and Experience in Unsupervised Learning.* Morgan Kaufmann.
[5] A. Franks, A. Miller, L. Bornn, K. Goldsberry, et al. 2015. Characterizing the Spatial Structure of Defensive Skill in Professional Basketball. *The Annals of Applied Statistics* 9, 1 (2015), 94–121.
[6] M. Kates. 2014. *Player Motion Analysis: Automatically Classifying NBA Plays.* Ph.D. Dissertation. Massachusetts Institute of Technology.
[7] Y. Li, W. Li, V. Mahadevan, and N. Vasconcelos. 2016. VLAD3: Encoding Dynamics of Deep Features for Action Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* Las Vegas, NV, USA.
[8] D. Lutz. 2012. A Cluster Analysis of NBA Players. In *Proceedings of the MIT Sloan Sports Analytics Conference.* Boston, MA, USA.
[9] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth. 2018. Machine Learning for Internet of Things Data Analysis: A Survey. *Digital Communications and Networks* 4, 3 (2018), 161–175.
[10] A. McIntyre, J. Brooks, J. Guttag, and J. Wiens. 2016. Recognizing and Analyzing Ball Screen Defense in the NBA. In *Proceedings of the MIT Sloan Sports Analytics Conference.* Boston, MA, USA, 11–12.
[11] A. McQueen, J. Wiens, and J. Guttag. 2014. Automatically Recognizing On-ball Screens. In *Proceedings of the MIT Sloan Sports Analytics Conference.* Boston, MA, USA.
[12] S. Narayan. 2019. *Applications of Machine Learning: Basketball Strategy.* Ph.D. Dissertation. Massachusetts Institute of Technology.
[13] A. Nistala. 2018. *Using Deep Learning to Understand Patterns of Player Movement in Basketball.* Ph.D. Dissertation. Massachusetts Institute of Technology.
[14] A. Nistala and J. Guttag. 2019. Using Deep Learning to Understand Patterns of Player Movement in the NBA. In *Proceedings of the MIT Sloan Sports Analytics Conference.* Boston, MA, USA, 1–14.
[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. 2011. Scikit-learn: Machine Learning in Python. *the Journal of Machine Learning Research* 12 (2011), 2825–2830.
[16] Y. Reich and S. J. Fenves. 1991. The Formation and Use of Abstract Concepts in Design. In *Concept Formation.* Elsevier, 323–353.
[17] J. Sampaio, E. J. Drinkwater, and N. M. Leite. 2010. Effects of Season Period, Team Quality, and Playing Time on Basketball Players' Game-related Statistics. *European Journal of Sport Science* 10, 2 (2010), 141–149.
[18] J. Sampaio, T. McGarry, J. Calleja-González, S. Jiménez Sáiz, X. Schelling i del Alcázar, and M. Balciunas. 2015. Exploring Game Performance in the National Basketball Association Using Player Tracking Data. *PloS one* 10, 7 (2015), e0132894.
[19] N. Seward. 2018. *Building an On-ball Screen Dataset Using Supervised and Unsupervised Learning.* Ph.D. Dissertation. University of Ontario Institute of Technology.
[20] M. Teramoto, C. L. Cross, R. H. Rieger, T. G. Maak, and S. E. Willick. 2018. Predictive Validity of National Basketball Association Draft Combine on Future Performance. *The Journal of Strength & Conditioning Research* 32, 2 (2018), 396–408.
[21] K-C Wang and R. Zemel. 2016. Classifying NBA Offensive Plays Using Neural Networks. In *Proceedings of the MIT Sloan Sports Analytics Conference.* Boston, MA, USA.
[22] I. H. Witten and E. Frank. 2002. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. *ACM Sigmod Record* 31, 1 (2002), 76–77.
[23] A. Yu and S. S. Chung. 2019. Automatic Identification and Analysis of Basketball Plays: NBA On-Ball Screens. In *2019 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD).* IEEE, Honolulu, Hawaii, 29–34.