ProKabaddi API

First Author[1,2*] and Second Author[2,3†]

[1*]Department of Computer Science and Engineering, Kalinga Institute of Industrial Technology, Patia, Bhubaneswar, 751024, Odisha, India.
[2]Department of Science, Indian Institute of Technology, Madras, Sardar Patel Road, Chennai, 600036, Tamil Nadu, India.

*Corresponding author(s). E-mail(s): abc@kiit.ac.in;
Contributing authors: abc@kiit.ac.in;
[†]These authors contributed equally to this work.

Abstract

Kabaddi, a contact team sport of Indian origin, has seen a dramatic rise in global popularity, highlighted by the upcoming Kabaddi World Cup with over sixteen international teams participating, alongside the flourishing national leagues such as the Indian Pro Kabaddi League (230 million viewers) and the British Kabaddi League. We present the first open-source python module to make Kabaddi statistical data easily accessible from multiple scattered sources across the internet. The module was developed by systematically web-scraping and collecting team-wise, player-wise and match-by- match data. The data has been cleaned, organized, and categorized into team overviews and player metrics, each filterable by season. The players are classified as raiders and defenders, with their best strategies for attacking, counter-attacking, and defending against different teams highlighted. Our module enables continuous monitoring of exponentially growing data streams, aiding researchers to quickly start building upon the data to answer critical questions, such as the impact of player inclusion/exclusion on team performance, scoring patterns against specific teams, and break down opponent gameplay. The data generated from Kabaddi tournaments has been sparsely used, and coaches and players rely heavily on intuitions to make decisions and craft strategies. Our module can be utilized to build predictive models, craft uniquely strategic gameplays to target opponents and identify hidden correlations in the data. This open source module has the potential to increase time-efficiency, encourage analytical studies of Kabaddi gameplay and player dynamics and foster reproducible research. The code is here: https://github.com/annimukherjee/ProKabaddi_API

Keywords: Open Source, Kabaddi, Sports Analytics, Keyword4

1 Introduction

Kabaddi is a fast-paced team-contact sport that has been rapidly gaining international recognition and popularity. In a game that can be loosely described as a combination of rugby, American football and tag [1] two teams take turns sending a player, called the "raider" into the opponent's half with the goal of tagging as many defenders as possible and returning to their own half without being tackled. The raider must accomplish this in a single breath for an offense lasting 30 seconds, all the while chanting "kabaddi" (pronounced kuh-bud-DEE).

Similar to American football, where the offense aims to penetrate the opponent's defense, the raider in Kabaddi must evade the opposing team's tackles to score points. Meanwhile the defense must work in coordination, aiming to tackle and immobilize the raider before they can return to their own half. However, unlike football, Kabaddi requires no ball or protective gear; it's a minimalist sport that focuses purely on strategy and physical strength.

Originally a traditional Indian sport, Kabaddi was first exhibited at the 1936 Berlin Olympics. The sport has gained further prominence, being a regular feature in the Asian Games since 1990, and with the launch of Kabaddi World Cup in 2004. The sport's popularity skyrocketed with the inception of the Indian Pro Kabaddi League (PKL) in 2014, now having 230 million viewers, second only to cricket's Indian Premier League.

With success of the PKL, several regional leagues, like the European Kabaddi Championship and the British Kabaddi League, have emerged. The sport has seen establishment of dedicated Kabaddi teams in countries such as Japan, the United States, South Korea and Iran.

With growing viewership across the board, there is a growing demand for analytics. Kabaddi lacks the analytical infrastructure seen in other sports, such as hockey, and frisbee. Existing data is scattered and fragmented, making it difficult for researchers to conduct meaningful analysis. Until now, even basic datasets required manual web scraping, as seen in limited studies here [2] and here [3].

In this paper, we present the first open source module that consolidates Kabaddi data from various sources into a single, standardized framework. This Python module not only simplifies data access but also enhances the potential for advanced analytics and draw advanced strategems.

Our module aims to standardize and aggregate kabaddi data from disparate sources, ensuring reliability and accessibility of data and foster a community of researchers and developers around Kabaddi analytics.
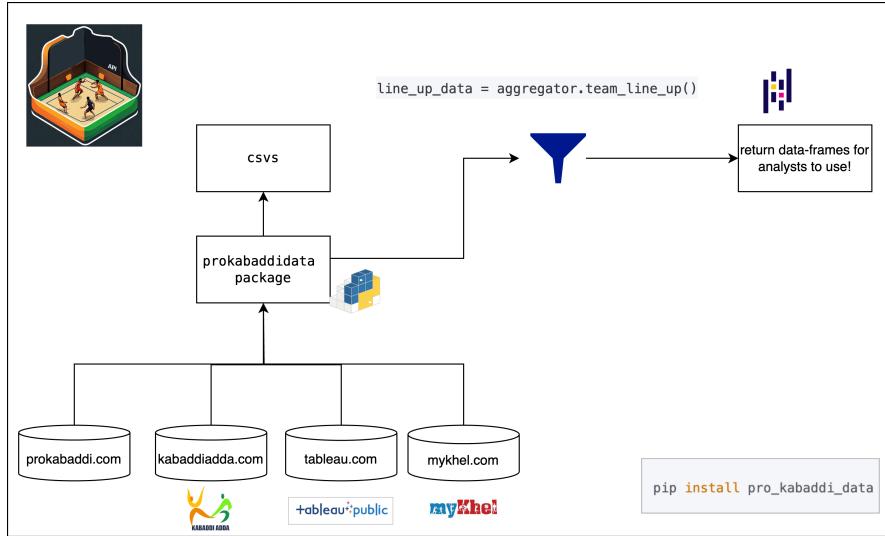
Fig. 1: Workflow of the OS Package

2 Overview of Kabaddi Play

Kabaddi is set on a rectangular court, measuring 10 by 13 metres (33 feet 43 feet) for men and 8 by 12 metres (26 feet 39 feet) for women, divided into two halves by a midline. Two teams of seven players compete on opposite ends of a court. The game consists of two 20-minute halves separated by a 5-minute halftime break during which the teams switch sides.

The offense operates as a unique "tag and return" system, distinct from the ball-focused scoring of American football. The offensive player, the raider, sprints into the opponent's half, and seeks to tag one or more defenders, and return to their own half without being tackled, all in a single breath. Each successful tag earns a point, with bonus points awarded for tagging multiple defenders or clearing the opponent's court. This is known as a "raid", and an entire raid must be completed in no more than 30 seconds. It is crucial that this be done in one breath, hence the necessity to continuously chant "kabaddi".

While Kabaddi's defense shares similarities with that of football, the defenders focus on preventing the raider from returning to their own half, rather than keeping them out. If the defenders successfully tackle and hold the raider before they can return, the defending team earns a point. Additionally, the defense scores if the raider goes out of bounds or fails to return to their half before exhaling (when they stop chanting "kabaddi". This creates a unique, high-stakes dynamic where every raid becomes a time-critical scoring opportunity for both the offense and defense.
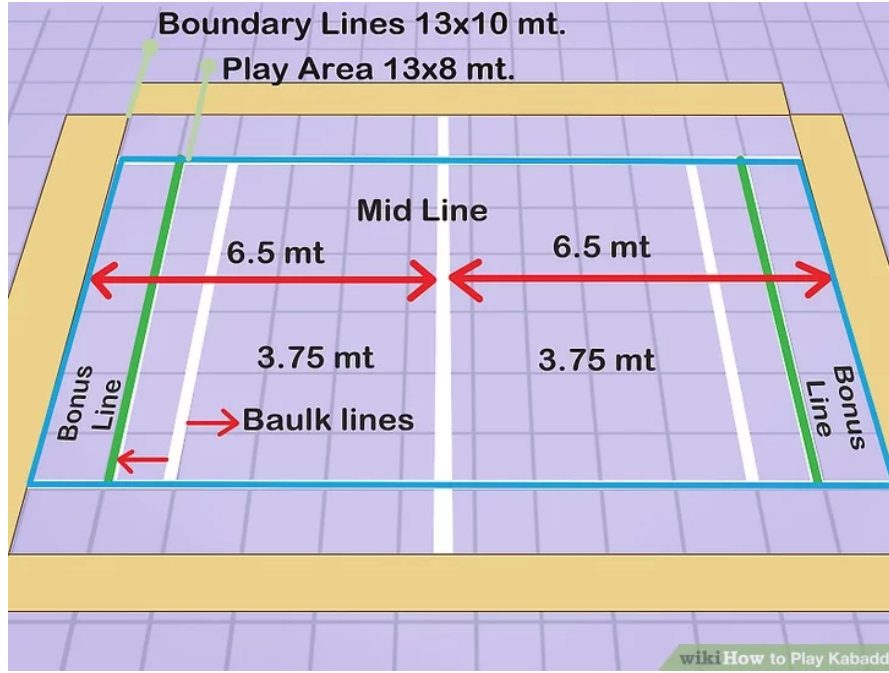
Fig. 2: Standard Style Kabaddi Mat

Players who are tagged or tackled are taken out of the game but can be "revived" when their team scores from a successful tag or tackle.

On the court, key lines play a crucial role in the point system. The baulk line, which is 3.75 meters (12.3 feet) from the midline, acts as a marker that the raider must cross to make the raid legal; if they fail to cross it without making a tag, no points are scored, and the raid is considered unsuccessful. The bonus line, located 1 meter (3.2 feet) beyond the baulk line, offers the raider an opportunity to score an additional point if they cross it while keeping at least one foot in the air and manage to return without being tackled.

On the defensive side, points are awarded not only for tackling the raider but also for forcing them out of bounds or causing them to fail in their breath-holding challenge. A team can further increase their score by achieving an "all-out," which involves eliminating all the opposing players on the court. This forces the opposition to reset their players on the court, and concede two points.

The strategic depth of Kabaddi lies in how teams deploy their raiders and defenders, utilizing formations, timing, and coordination to gain an advantage. Defensive tactics, such as chain tackles and "super tackles," where a shorthanded team successfully tackles a raider, add another layer of complexity to the game.

In Table 1 and Table 2, various defensive and offensive strategies for scoring points are described briefly.[4]

4

Defensive Move	Description
Waist/back hold	When the defender attempts to grab the raider mid-air by the waist to pin them down on the mat.
Ankle hold	The defenders attempt to stop the raider by grabbing the ankle.
Thigh hold	The defenders attempt to hold the thigh of raider with both hands.
Block	The defenders physically try to stop the raider from crossing the mid line and go back to their court.
Chain Tackle	When two or more defenders attempt a co-ordinated tackle to prevent the raider from crossing the midline
Dash	When defender pushes the raider out of the court by "dashing". Earns a point for the defense.

Table 1: Table describing various defending strategies used in Kabaddi

Offensive Move	Description
Toe-Touch	The raider attempts to score by touching just the defenders' toe.
Hand-touch	Raider attempts to score by touching the defenders' with their arm.
Front and side kick	The raider attempts to score by kicking in front or sideways.
Reverse/back Kick	Raider can turn their back towards the defender and kick backwards to score a touch point.
Leg Thrust	The raider uses their leg strength to push through the defenders; used when trapped in a hold or tackle.
Dubki (duck)	The raider ducks below the defenders to reach the half line. Used to avoid chain tackle

Table 2: Table with diverse raider strategies used in Kabaddi

3 Methodology/Data Collection

The Kabaddi data module was developed as a comprehensive Python package that collects and collates live data from various sources through web scraping. This data is systematically stored in a central repository for ease of access and further analysis. The package includes a wide array of functions, with the most critical ones prefixed by "(insert name)" to return actionable data, such as player statistics and match outcomes.

In addition to these core functions, the package offers several helper functions specifically designed to process, parse and clean the raw data into structured. These functions often begin with the "(insert name)" prefix and are essential for cleaning and preparing the data retrieved through the web scraping process. While these processing functions were developed as part of the Kabaddi package, they are not expected to be used by anyone for general Kabaddi analytics. However, to ensure full reproducibility and transparency, we provide both the web scraping functions and the collated data. This allows researchers to not only access pre-processed data (by function calls) but also to replicate the data collection process.

4 Installing ProKabaddi API

ProKabaddi API, (insert latest version name) is available on the Python Package Index (PyPi) and can be downloaded using pip.

```
pip install pro_kabaddi_data
```

Source Name	Website URL	Data Type	Frequency of Update	Source	Integration Status
Pro Kabaddi Website	www.prokabaddi.com	Match statistics, player performance	Real-time (during season)	HTML	Fully Integrated
ESPN Kabaddi	www.espn.in/kabaddi	Match scores, team statistics	After each match	HTML	Partially Integrated
Tableau	www.tableau.com	Player profiles, performance history	Weekly updates	HTML	Fully Integrated
Kabaddi Adda	www.kabaddiadda.com	Player Auction Data	Real-time	API	Fully Integrated
MyKhel	https://www.mykhel.com/ kabaddi/pro-kabaddi-league	Player Auction Data	Real-time	HTML	Fully Integrated

Table 3: Data sources

The class can be initialized with the below. All the functions belong to this class and can be accessed accordingly.

```
from prokabaddidata import prokabaddidata

# Initialize the aggregator
aggregator = prokabaddidata.KabaddiDataAggregator()
```

5 Module Usage

Our module enables an analyst to start from the very basic data, such as the season standings, and move to advanced statistics such as, most effective raider against 7 defenders for a team!

For example to get play-by play data

```
# Load data
match_id = '60'
season = '1'
match_events = aggregator.get_match_events(season, match_id)
print(match_events)
```

This would return all the play-by-play events in a dataframe, with extremely detailed view.

event_no	event	event_id	event_text	raider_id	...
0	Empty Raid	3	Rohit Kumar Empty Raid	326.0	...
1	Successful Raid	1	Sachin Tanwar raids successfully	757.0	...
2	Empty Raid	3	Rohit Kumar Empty Raid	326.0	...
3	Empty Raid	3	Rohit Gulia Empty Raid	686.0	...
4	Unsuccessful Raid	2	Sumit Singh Unsuccessful Raid	363.0	...
⋮	⋮	⋮	⋮	⋮	⋮
85	Unsuccessful Raid	2	Vinod Kumar Unsuccessful Raid	764.0	...

event_no	event	event_id	event_text	raider_id	...
86	Successful Raid	1	Rohit Kumar raids successfully	326.0	...
87	Empty Raid	3	More GB Empty Raid	772.0	...
88	Empty Raid	3	Mahender Singh Empty Raid	769.0	...
89	Empty Raid	3	More GB Empty Raid	772.0	...

Additional columns: 'raiding_team_id', 'defender_id', 'defending_team_id', 'raid_points', 'raid_touch_points', 'raid_bonus_points', 'raid_technical_points', 'raid_all_out_points', 'defending_capture_points', 'defending_bonus_points', 'defending_technical_points', 'defending_all_out_points', 'defending_points', 'super_raid', 'super_tackle', 'clock', 'status_id', 'do_or_die', 'review', 'score', 'defenders', 'reason', 'player_id', 'team_id', 'substituted_by'
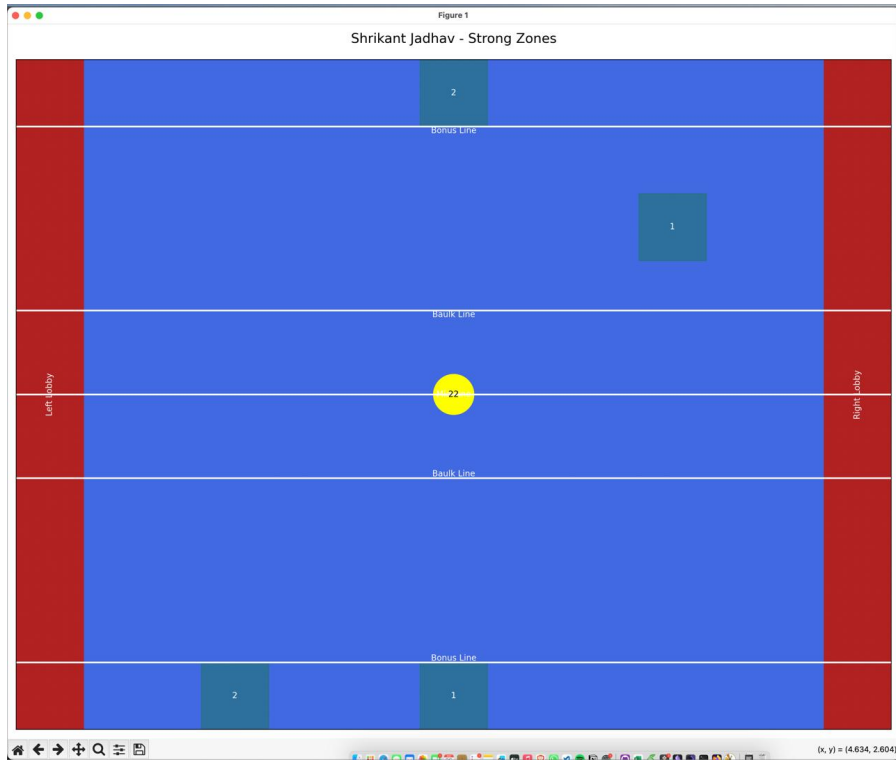

Fig. 3: Strong Zones of a raider

5.1 Standings

The `getStandings()` function serves as a starting point for researchers unfamiliar with the data.

7

```
standings = get_pkl_standings()
print(standings)
```

The returned dataframe includes essential details such as team IDs, which can be used to get stats for team level analysis.

Group	Season	Team_Name	$\cdots$	Qualified
Main	7	Dabang Delhi K.C.	$\cdots$	True
Main	7	Bengal Warriorz	$\cdots$	True
Main	7	UP Yoddhas	$\cdots$	True
Main	7	U Mumba	$\cdots$	True
Main	7	Haryana Steelers	$\cdots$	True
Main	7	Bengaluru Bulls	$\cdots$	True
Main	7	Jaipur Pink Panthers	$\cdots$	False
Main	7	Patna Pirates	$\cdots$	False
Main	7	Gujarat Giants	$\cdots$	False
Main	7	Puneri Paltan	$\cdots$	False
Main	7	Telugu Titans	$\cdots$	False
Main	7	Tamil Thalaivas	$\cdots$	False

Table 5: League Standings

Additional columns :Matches played, Wins Lost Tied Draws No Result League points Score diff Qualified

5.2 Team-Level Analysis

Once the standings is accessed, researchers might wish to dive deeper into team-level data. The Kabaddi API supports this with three key functions:

- **Aggregated Team Statistics:**

Rank	Season	Team ID	Team Name	All-outs Inflicted/Match	Avg Tackle Points/Match
1.0	4	Bengal Warriors	1.571	0.622	7.571
2.0	4	Bengal Warriors	0.429	0.699	8.286
3.0	4	Bengal Warriors	0.250	0.586	8.563
4.0	4	Bengal Warriors	0.500	0.638	8.357
5.0	4	Bengal Warriors	1.125	0.335	7.542
6.0	4	Bengal Warriors	1.000	0.378	8.087
7.0	4	Bengal Warriors	1.708	0.398	9.042
8.0	4	Bengal Warriors	1.227	0.368	7.545
9.0	4	Bengal Warriors	1.364	0.442	9.273
10.0	4	Bengal Warriors	1.409	0.421	8.727
–	4	Bengal Warriors	1.169	0.046	8.318

Table 6: Bengal Warriors Performance Across Seasons

- **Match Details for a Team:** All matches played by a specific team within a season.

Match Name	Match ID	Tour Name	Date	Result
Match 10	295	Pro Kabaddi League Season 5, 2017	8/2/2017	Bengal Warriors beat Telugu Titans (30-24)
Match 16	301	Pro Kabaddi League Season 5, 2017	8/6/2017	Bengal Warriors beat U.P. Yoddha (40-20)
Match 20	305	Pro Kabaddi League Season 5, 2017	8/9/2017	Bengaluru Bulls beat Bengal Warriors (31-25)
⋮	⋮	⋮	⋮	⋮
Match 131	416	Pro Kabaddi League Season 5, 2017	10/20/2017	Match Tied
Qualifier 1	420	Pro Kabaddi League Season 5, 2017	10/24/2017	Gujarat Fortunegiants beat Bengal Warriors (42-
Qualifier 2	422	Pro Kabaddi League Season 5, 2017	10/26/2017	Patna Pirates beat Bengal Warriors (47-44)

Table 7: Pro Kabaddi League Season 5 (2017) Match Results

- **Player Roster:** The API also allows retrieval of the player roster for a team in a given season. This includes player IDs and other relevant information, which can be used for more detailed player-level analysis.

ID	Name	Jersey	Capt.	Played	Starter	Top Raider	Top Defender
143	Maninder Singh	9	✓	✓	✓	✓	
5114	Vaibhav Garje	15		✓	✓		
4978	Balaji D	2		✓	✓		
3151	Manoj Gowda	27		✓	✓		
106	Shrikant Jadhav	22		✓	✓		
3103	Shubham Shinde	69		✓	✓		✓
161	Girish Maruti Ernak	8		✓	✓		
146	Surender Nada	1		✓			
41	Deepak Hooda	5					
4195	Sakthivel R	7		✓			
4935	Aslam Thambi	17					
4929	Soleiman Pahlevani	33					
4723	Rohit	42		✓			
2025	Ajinkya Kapre	45					
764	Vinod Kumar	3		✓	✓		
202	Ashish Sangwan	10		✓	✓		✓
4675	Akash Pikalmunde	4					
5120	R Guhan	71		✓			
3100	Parveen Satpal	99					
5097	Parshant Kumar	90					

Table 8: Player Information

5.3 Matches

After obtaining the Match Id, `getMatches()` function can be used provides an overview of matches.

5.4 Granular Match Analysis: Play-by-Play Details

Allows to analyze the match with respect to the clock progression.

Match Name	Match ID	Tour Name	$\cdots$	Patna Pirates Score	Haryana Score
Match 1	3029	Pro Kabaddi League Season 10, 2023	$\cdots$	–	42
Match 2	3030	Pro Kabaddi League Season 10, 2023	$\cdots$	–	–
Match 3	3031	Pro Kabaddi League Season 10, 2023	$\cdots$	34	–
Match 4	3032	Pro Kabaddi League Season 10, 2023	$\cdots$	–	–
Match 5	3033	Pro Kabaddi League Season 10, 2023	$\cdots$	–	54
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
Eliminator 1	3161	Pro Kabaddi League Season 10, 2023	$\cdots$	37	–
Eliminator 2	3162	Pro Kabaddi League Season 10, 2023	$\cdots$	–	42
Semi Final 1	3163	Pro Kabaddi League Season 10, 2023	$\cdots$	21	–
Semi Final 2	3164	Pro Kabaddi League Season 10, 2023	$\cdots$	–	31
Final	3165	Pro Kabaddi League Season 10, 2023	$\cdots$	–	25

Table 9: Pro Kabaddi League Season 10 Matches

```
match_events = api.get_match_events(season=1, match_id=60)
print(match_events)
```

event_no	event	event_id	event_text	raider_id	...
1	Empty Raid	3	Rohit Kumar Empty Raid	326.0	...
2	Successful Raid	1	Sachin Tanwar raids successfully	757.0	...
3	Empty Raid	3	Rohit Kumar Empty Raid	326.0	...
4	Empty Raid	3	Rohit Gulia Empty Raid	686.0	...
5	Unsuccessful Raid	2	Sumit Singh Unsuccessful Raid	363.0	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
86	Unsuccessful Raid	2	Vinod Kumar Unsuccessful Raid	764.0	...
87	Successful Raid	1	Rohit Kumar raids successfully	326.0	...
88	Empty Raid	3	More GB Empty Raid	772.0	...
89	Empty Raid	3	Mahender Singh Empty Raid	769.0	...
90	Empty Raid	3	More GB Empty Raid	772.0	...

Table 10: Play-by-Play Details for Match ID 60

Additional columns: Raiding Team ID, Defender ID, Defending Team ID, Raid Points, Raid Touch Points, Raid Bonus Points, Raid Technical Points, Raid All Out Points, Defending Capture Points, Defending Bonus Points, Defending Technical Points, Defending All Out Points, Defending Points, Super Raid, Super Tackle, Clock, Status ID, Do or Die, Review, Score, Defenders, Reason, Player ID, Team ID, and Substituted By.

5.5 Player-Level Analysis

Finally, using the player IDs obtained from a team's roster, the API enables detailed player-level analysis. You can access aggregated statistics for each player, such as the total number of successful raids and tackles in a season. The API also offers insights

into the player's starting status, the number of matches started versus played, and specific performance metrics for raiders, like their success in raids relative to the number of defenders faced.

6 Conclusion

The Kabaddi API is structured to reflect the natural flow of Kabaddi analysis, from league standings to granular match and player data. This modular and intuitive approach ensures that researchers can easily navigate the data, focusing on the specific aspects that are most relevant to their analysis. The API not only simplifies data retrieval but also empowers researchers to conduct thorough, multi-layered analyses of the sport, from high-level overviews to detailed, event-level investigations.

References

[1] Garfinkel P.: Rugby Meets Red Rover: Kabaddi Has Captured the Heart of India. Available from: https://www.nytimes.com/2018/10/02/sports/kabaddi-india.html.

[2] Singh P, Parashar B, Agrawal S, Mudgal K, Singh P. Kabaddi: a quantitative approach to machine learning model in Pro Kabaddi. In: Proceedings of Second International Conference on Computational Electronics for Wireless Communications: ICCWC 2022. Springer; 2023. p. 243–260.

[3] Parmar M.: KABADDI: From an intuitive to an quantitative approach for analysis, predictions and strategy.

[4] Singh S, Srivastava DP, Patvardhan C. Game theoretic analysis of Kabaddi. J Stat Appl Prob. 2023;12(1):313–319.