Literature Review

**Introduction**

This chapter presents a comprehensive review of the existing literature on sports analytics platforms, with a particular focus on rugby. It explores the core methodologies and technologies used in data collection, analysis, and visualization, emphasizing the competitive landscape in sports analytics. By examining existing tools, frameworks, and techniques, this chapter aims to highlight the potential of integrating web scraping, APIs, and machine learning to enhance performance insights in rugby.

The literature review delves into common data collection techniques, including web scraping and API usage, which are critical for gathering real-time match data, player statistics, and game outcomes. In terms of visualization, the chapter covers the use of the Django framework and database management solutions that provide the foundation for creating interactive and insightful graphical representations of player and team metrics. It discusses effective visualization strategies, including graph types best suited for conveying trends, comparisons, and performance patterns in sports data.

Machine learning applications in sports analytics are explored, with a focus on their role in predictive modelling and performance assessment. The chapter also addresses the importance of security and privacy in web scraping practices, touching on the ethical implications of data collection. Through this literature review, a clearer understanding of current challenges, innovations, and future prospects in sports analytics platforms for rugby is established, setting the stage for subsequent chapters of this project.

**Sports Analytics in Rugby**

**Competitive Analysis**

In conducting a competitive analysis, I evaluated several existing applications in the sports analytics space, specifically focusing on Ultimate Rugby, RugbyPass, Flashscore, and FotMob. Each of these applications offers a unique set of features that inspired various aspects of my project.

FotMob, a soccer-focused application, was my primary inspiration. Its well-designed user interface provides users with easy access to comprehensive soccer information, including league tables, cup standings, team and player information, and most importantly, in-depth game statistics and player performance metrics. Inspired by FotMob, I aim to implement similar features for rugby, such as current, upcoming, and past fixtures, alongside predicted lineups and injury/suspension updates, which would later be updated with official lineup announcements.

Ultimate Rugby also offers features that align well with my project’s goals. The web version provides detailed statistical breakdowns, such as possession and territory by half, as well as attack and defence metrics. These breakdowns offer a valuable model for presenting rugby-specific data and ensuring users can access insightful statistics.

RugbyPass incorporates several graphical elements that I found particularly effective, including a points flow chart and attack/ defence profiles. Additionally, their ruck speed metric provides a unique performance measure. RugbyPass also presents stats on set plays, carries, line breaks, and tackles completed, while the head-to-head comparison page offers an innovative way to visualize team form, which is something I aim to include in my platform.

Flashscore, while not exclusive to rugby, provides an extensive archive page displaying league champions over the years. This historical data presentation is especially useful for leagues like the United Rugby Championship (URC), which has undergone several name changes over time. I also appreciated the inclusion of player nationality on the lineup page, adding a global perspective that could be interesting for rugby fans.

Notably, most of these applications include a betting odds feature. However, I have chosen to omit this from my project, as it would not be appropriate for an academic setting.

**Data Collection Techniques**

Effective data collection is critical for the development of a robust rugby analytics application. In this section, I will discuss both the techniques I intend to use for this project and some additional data collection methods that have been utilized in professional rugby analytics.

To collect my initial dataset, I will use **web scraping techniques** to gather detailed player, club, and match statistics. Specifically, I plan to leverage Python libraries such as BeautifulSoup, Selenium, and Playwright. These tools enable efficient data extraction from websites, offering functionalities for HTML parsing, automated website navigation, and data retrieval across various leagues, teams, and players. I chose Python for its extensive ecosystem of libraries tailored to web scraping, which simplifies and accelerates the data collection process (Mitchell, 2015).

If I were affiliated with a rugby club, I would also consider using **wearable technology** for enhanced data collection. Devices like GPS trackers and heart rate monitors are widely used in rugby to monitor player movement, speed, and distance covered, as well as physiological responses during games. These wearable technologies provide valuable insights into player performance and physical demands.

Additionally, professional rugby has introduced **smart ball technology**, which debuted in 2021. The "smart ball" uses embedded sensors to capture real-time metrics such as speed, distance, hang time, rotation, and trajectory. Unfortunately, while this data can be highly insightful for analytics, it is not publicly accessible and therefore cannot be directly incorporated into this project.

By combining web scraping with an understanding of advanced data collection methods in rugby, I aim to create a comprehensive dataset that will support the application’s analytics capabilities.

**Data Visualization**

For this project, I will use Python's Django framework for backend processing and data handling. Django is well-suited to process rugby data—such as match scores, player statistics, and league standings—using libraries like Pandas for efficient data preprocessing and organization. For visualizing this data, I plan to leverage Matplotlib, Plotly, Seaborn, and Django Chart.js. Django’s flexibility allows visualizations to be embedded directly into HTML templates or served as JSON data, which can then be rendered using JavaScript libraries like Chart.js for a seamless front-end experience.

The primary visualization types I intend to use include line graphs, area graphs, bar graphs, radar charts, tables, pie charts, stacked bar charts, heat maps, bubble charts, timeline charts, and stacked area charts. Below is a breakdown of each chart type, its use case, and the insights it offers:

1. **Score Progression During a Match**:
   * **Line Graph**: Illustrates score progression over time, revealing momentum shifts and score accumulation by each team.
   * **Area Graph**: Adds depth to the line graph by filling areas under the curve, emphasizing cumulative scoring trends.
2. **Player Performance Metrics**:
   * **Bar Graph**: Allows comparison of individual player statistics, such as tackles, passes, and tries scored, either across games or among players in a single match.
   * **Radar Chart**: Ideal for multi-metric comparisons for individual players, displaying attributes like tackling, passing, and scoring in a single, cohesive visual.
3. **League Standings and Points Table**:
   * **Table Visualization**: A straightforward table for ranking teams by points, wins, losses, and draws, suitable for structured, easy-to-read data.
   * **Bar Chart/Heat Map**: A horizontal bar chart can visually display league standings or points, where each team’s bar length reflects points or rankings.
4. **Win/Loss Ratios**:
   * **Pie Chart/Donut Chart**: Shows each team’s win, loss, and draw ratios across the season, providing a snapshot of performance distribution.
   * **Stacked Bar Chart**: Combines wins, losses, and draws in one bar per team, allowing for side-by-side team comparisons.
5. **Head-to-Head Comparison**:
   * **Heat Maps**: Displays field zones with high possession, tackles, or scoring opportunities, offering spatial insights valuable for strategy and tactics.
   * **Bubble Chart**: Represents scoring frequency from different field areas or tackles by player position, highlighting hotspots and impact zones.
6. **Match Events and Play Sequences**:
   * **Timeline Chart**: Plots significant game events (e.g., tries, penalties, substitutions) in chronological order, giving an overview of match flow and key moments.
   * **Stacked Area Chart**: Visualizes time-based control metrics, such as possession over halves or quarters, to indicate dominance periods and how control shifts throughout a match.

With these visualizations, I aim to create a comprehensive league performance dashboard that displays critical metrics like points, try conversions, and territory gained. Additionally, player profile pages will include radar charts and line graphs tracking individual performance across the current and previous seasons.

Leveraging Django’s backend data-serving capabilities alongside Python’s visualization libraries, I can develop a user-friendly, interactive interface for exploring rugby data. This approach not only enhances data accessibility but also adds analytical depth through tailored, dynamic visualizations.

**Ethical and Security Considerations in Web Scraping**

Web scraping, while valuable for gathering data from publicly accessible websites, brings forth notable ethical, privacy, and security concerns that must be addressed to ensure responsible and respectful data usage. Ethical web scraping not only involves compliance with legal standards but also emphasizes a collaborative approach between scrapers and site owners, fostering a balanced online ecosystem where data is accessed and used responsibly.

**Principles of Ethical Web Scraping**

Ethical web scraping is grounded in principles of transparency, respect for data ownership, and minimal impact on site performance. According to (Densmore, 2017), an "ethical scraper" operates with the understanding that public APIs should be prioritized over direct scraping whenever possible, reducing unnecessary strain on web servers. Furthermore, ethical scrapers should provide a clear User Agent string, allowing website administrators to identify and contact the scraper in case of performance issues or ethical concerns. By adhering to rate limits and only gathering necessary data, scrapers can avoid causing unintentional disruption, such as Denial of Service (DoS) incidents, which could impair access for legitimate users (Khder, 2021).

Ethical scrapers are also encouraged to find ways to add value to the sites they access. This might involve driving organic traffic back to the website or crediting the data source in applications or reports. Such practices help build trust and create a more symbiotic relationship between the scraper and the site owner, as scrapers strive to use data for creating new value rather than replicating or redistributing it without permission.

**Privacy Concerns in Data Scraping**

Privacy is another critical consideration in web scraping, especially regarding personal data that could be subject to privacy laws such as the General Data Protection Regulation (GDPR). Data collection without user consent, particularly for personally identifiable information (PII), can result in privacy violations. Ethical web scrapers should therefore prioritize aggregating and anonymizing data, significantly reducing the risk of privacy infringements while preserving valuable insights for analysis (Ali Hussain Ahmad & Gafar Zen Alabdeen Salh Hassan, 2024).

Moreover, transparency regarding the use of scraped data builds trust with users and aligns with responsible data practices. Users whose data is publicly available may not expect it to be repurposed or analysed elsewhere, so implementing notice or consent mechanisms, even if informal, can serve as a best practice to mitigate privacy risks.

**Security and Data Integrity in Web Scraping**

In web scraping, security considerations focus primarily on data integrity and compliance with site-specific access restrictions. Maintaining data accuracy is essential in applications that rely on near real-time data, as inaccuracies could lead to flawed analytics or decision-making (Khder, 2021). Additionally, websites often implement IP blocking and bot detection mechanisms to safeguard against unauthorized or excessive scraping activities. Circumventing these restrictions, such as by using IP rotation, raises ethical and legal issues, potentially violating the website’s terms of service and compromising site security (Zachary Gold & Mark Latonero, 2018).

The role of the robots.txt file is central to ethical web scraping. While this file is not legally enforceable, it serves as a guiding standard by indicating which parts of a website are open to bot’s and which are restricted. Adhering to these specifications showcases respect for the site owner’s preferences and highlights a commitment to responsible data collection (Zachary Gold & Mark Latonero, 2018). By observing the robots.txt file, scrapers demonstrate adherence to widely accepted ethical practices, reducing the likelihood of unauthorized access and fostering greater transparency.

**The Role of Ethical Site Owners**

Just as scrapers have ethical responsibilities, so do website owners. Ethical site owners, as (Densmore, 2017)points out, should be open to allowing access for scrapers that abide by transparent and respectful practices. Rather than blocking such scrapers outright, site owners are encouraged to engage constructively, reaching out when concerns arise instead of resorting to permanent blocks. Providing a public API as an alternative to direct scraping is another proactive step site owners can take, reducing server strain while making data accessible for responsible use.

**Role of robots.txt in Ethical Web Scraping**

The robots.txt file is a widely adopted mechanism that websites use to communicate their preferences regarding automated access. Although non-binding, adhering to robots.txt is a generally accepted standard of ethical web scraping. Websites can specify which sections are accessible to bots and which are restricted. For example: Figure 1, Figure 2

A black background with green text

Description automatically generated

Figure 1

A computer screen with white text

Description automatically generated

Figure 2 Munster Rugby's Robots.txt

In this example, bots are requested to avoid /private-data/, while /public-data/ is accessible. This guidance, if followed, respects the website’s preferences and safeguards against unauthorized scraping. The use of robots.txt provides important direction for ethical scraping by indicating acceptable parts of the website for data extraction. In the case of a sports analytics platform, examining each site’s robots.txt can help ensure that the scraping targets only permissible data sections, aligning with responsible data gathering practices. Adhering to robots.txt demonstrates respect for the autonomy of the website owner and underscores a commitment to ethical web scraping.

**Machine Learning**

(needs more research)

**Conclusion**

This literature review has outlined the foundational methodologies and technologies that construct sports analytics platforms, particularly in the context of rugby. By examining data collection techniques, such as web scraping and API integrations, alongside advanced data visualization strategies using Django and other tools, this chapter has emphasized the possibilities for developing a comprehensive rugby analytics platform. The integration of these methods offers robust avenues for enhancing insights into player and team performance, bringing added depth to sports data analysis.

In addition, this review has indicated the importance of ethical considerations in data collection practices. Embracing privacy, security, and transparency within web scraping protocols reflects a commitment to responsible data use. Through careful adherence to ethical guidelines, such as respecting robots.txt directives and prioritizing privacy, the project aligns itself with best practices that build trust and integrity in sports data analytics.

Looking ahead, the application of machine learning holds significant promise for predictive modelling and performance assessment in rugby. While additional research into machine learning applications remains necessary, the insights gained from this literature review lay the groundwork for implementing data-driven decision-making tools within the platform. In conclusion, the exploration of current technologies, ethical frameworks, and visualization techniques establishes a well-rounded basis for developing an innovative and user-friendly rugby analytics platform.

# References

Ali Hussain Ahmad & Gafar Zen Alabdeen Salh Hassan, 2024. Web Mining overview: Techniques, Tools, and Ethical Implications. *Journal of Computer Sciences,* pp. 86-94.

Densmore, J., 2017. *Ethics in Web Scraping.* [Online]   
Available at: https://towardsdatascience.com/ethics-in-web-scraping-b96b18136f01

Khder, M. A., 2021. Web scraping or web crawling: State of art, techniques, approaches, and application. *International Journal of Advances in Soft Computing & Its Applications,* Volume 13, p. 25.

Mitchell, R., 2015. *Web Scraping with Python: Collecting Data from the Modern Web.* Sebastopol, CA: O'Reilly Media.

Zachary Gold & Mark Latonero, 2018. Robots welcome: Ethical and legal considerations for web crawling and scraping. *Washington Journal of Law, Technology & Arts,* 13(3).

[Figure 1 Example of Robots.txt 6](#_Toc181619851)

[Figure 2 Munster Rugby's Robots.txt 6](#_Toc181619852)