How did Sunrisers Hyderabad reach from bottom to top in Indian Premier League(IPL) 2024?

Bhavya Bhargava

An Enthusiast's Perspective

Dated: 2024-08-31

Data Details | Blog post | Behind the Scenes

My whole analysis is based around the question:

What were the reasons for Sunrisers Hyderabad, a recently bottom-feeding team to suddenly rise up to the Top in the internationally acclaimed Indian Premier League(IPL) this year?

The data for this analysis was obtained by web scrapping from various sources and the description for each is as below:

Copyright and Accreditation:

Starting off, The player's name and role data is web scrapped from crictracker.com who in their copyright notice they have stated that the data can be copied or extracted from their website by third parties for their personal and non-commercial use [1]. They themselves state in their DMCA policy, that the data held by them is considered available in public domain and is accredited wherever they reference it [2]. As there were some data inconsistencies in the data from crictracker.com, some wrangling had to be manually done which was done by referencing iplt20.com, IPL's official data resource.

Coming to the performance data of the teams, it was web-scrapped from statisticstimes.com which in its terms of use policy states that anyone can use the data present on the site and that it is not responsible for any inaccuracies or accuracies of the data [3]. Statisticstimes.com itself captures the data from IPL T 20 official website and as per their media creation terms and conditions no data should be used for commercial usage [4]. As we are not utilising this data for non commercial purposes it was deemed OK to use it.

Purpose for Data Usage:

Analyzing team performance differences: By examining the structure of the team over the seasons, Analysing their performance over the latest season and looking at the performance gap over the seasons we can understand the rise in the team's performance and justify the changes which have been made to the team since the previous seasons.

Metadata:

1. Player: Description – Name of the player part of SRH for that season. Datatype – Character/String.
2. Role: Description - Role of the player part of SRH for that season. Datatype – Character/String.
3. Team: Description – Initials of the team playing in 2024 season. Datatype – Character/String.
4. Win: Description - Number of matches won by the team in 2024 season. Datatype – Numeric (Double).
5. Lose: Description - Number of matches lost by the team in 2024 season. Datatype – Numeric (Double).
6. No Result: Description - Number of matches the team got no result in 2024 season due to unforeseen circumstances. Datatype – Numeric (Double).
7. Result: Description - Outcome for the team in the 2024 season. Datatype – Character/String.
8. Value: Description - Number of matches of that particular outcome for that team in 2024 season. Datatype – Numeric (Double).
9. Year: Description - Year season in which SRH played the IPL. Datatype – Numeric (Double).
10. Run Rate: Description – The net run rate of SRH over the 2024 season. It is a performance indicator based on the average runs that were scored for the team minus the average scored against them. Datatype – Numeric (Double).

Data Acquisition

For gathering the player's name and role data I have scrapped multiple pages of crictracker.com which is the fastest-growing website in India for everything cricket and is run by a few cricket fans [5][6][7].

```r
# Importing the libraries utilized
library(tidyverse)

# Importing the library for data scraping (keeping redundant packages)
library(rvest)
library(httr)

# Function to scrape player data
extract_srh_players <- function(url, year) {
  # Read the webpage content
  page <- read_html(url)

  # Finding the heading node that contains the desired table
  if (year == 2022) {
    heading_text <- paste0("Sunrisers Hyderabad (SRH) Team ", year, " players list:")
    heading_node <- page %>%
      html_nodes(xpath = paste0("//h3[contains(text(), '", heading_text, "')]"))
  } else if (year == 2023) {
    heading_text <- paste0("Sunrisers Hyderabad (SRH) Team ", year, " players list:")
    heading_node <- page %>%
      html_nodes(xpath = paste0("//h3[b[contains(text(), '", heading_text, "')]]"))
  } else if (year == 2024) {
    heading_text <- "Sunrisers Hyderabad (SRH)"
    heading_node <- page %>%
      html_nodes(xpath = paste0("//h3[google-sheets-html-origin[contains(text(), '", heading_text, "')]]"))
  } else {
    stop("Unsupported year: ", year)
  }

if (length(heading_node) == 0) {
  stop("Heading not found for year: ", year)
}

# Finding the table node following the heading node
table_node <- heading_node %>%
  html_nodes(xpath = "following-sibling::div//table") %>%
  html_node("tbody")

if (length(table_node) == 0) {
  stop("Table not found for year: ", year)
}

# Extracting the table data
table_data <- table_node %>%
  html_table(header = TRUE)

if (year == 2023) {
  # For 2023, the first row contains the header, and we need to remove the "Price" column
  table_data <- table_data[[1]] %>%
    slice(-1) %>%
    rename(Player = 1, Role = 2) %>%
    select(Player, Role)
} else if (year == 2024) {
  if (length(table_data) == 0 || length(table_data[[1]]) < 2) {
    stop("Table extraction failed or table has less than 2 columns for year: ", year)
  }
    table_data <- table_data[[1]] %>%
      rename(Player = 1, Role = 2) %>%
      select(Player, Role)
  } else {
    if (length(table_data) == 0 || length(table_data[[1]]) < 2) {
      stop("Table extraction failed or table has less than 2 columns for year: ", year)
    }

    table_data <- table_data[[1]] %>%
      select(Player, Role)
  }

  return(table_data)
}

# URLs of the pages containing the tables
url_2022 <- "https://www.crictracker.com/srh-players-list-2022-complete-sunrisers-hyderabad-squad-and-players-list-for-ipl-2022/"
url_2023 <- "https://www.crictracker.com/cricket-news/srh-team-2023-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2023/"
url_2024 <- "https://www.crictracker.com/cricket-news/srh-team-2024-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2024/"

# Extracting the players data for 2022, 2023, and 2024 and storing
players_2022 <- extract_srh_players(url_2022, 2022)
players_2023 <- extract_srh_players(url_2023, 2023)
players_2024 <- extract_srh_players(url_2024, 2024)
```

Here the function reads the webpage content using the rvest package and locates the heading node that contains the player list for the respective year. Based on the year, it adjusts the XPath query to find the correct heading and table. It then extracts the player data from the table, processes it to remove unnecessary columns and rows, and returns the cleaned data frame. The script then uses this function to scrape the player data from three specific URLs for 2022, 2023, and 2024, storing the results in players_2022, players_2023, and players_2024 data frames, respectively.

Additional Data-Wrangling to fix the player data inaccuracies

After checking against the official data source on iplt20.com it was found that there were some inaccuracies and inconsistencies in the player's name and role information [8]. Below is the code to fix it and make it usable for visualizations.

```
# Individual data entries modified to reflect the official data at iplt20.com

# Modifying roles in players_2022 data frame
players_2022 <- players_2022 %>%
  mutate(Role = case_when(
    Player == "Glenn Phillips" ~ "Batter",
    Player == "Vishnu Vinod" ~ "Wicket-Keeper",
    Role == "All-Rounder" ~ "All-rounder",
    Role == "Batsman" ~ "Batter",
    TRUE ~ Role
  ))

# Modifying roles in players_2023 data frame
players_2023 <- players_2023 %>%
  mutate(
    Player = str_replace(Player, "\\s*\\([^\\)]+\\)", ""),
    Role = case_when(
    Player == "Glenn Phillips" ~ "Batter",
    Role == "Wicket-keeper" ~ "Wicket-Keeper",
    Role == "All-Rounder" ~ "All-rounder",
    TRUE ~ Role
  ))

# Valid Roles of the Players listed
valid_roles <- c("All-rounder", "Batter", "Bowler", "Wicket-Keeper")

# Modifying roles in players_2024 data frame
players_2024 <- players_2024 %>%
  mutate(Player = str_replace(Player, "\\s*\\([^\\)]+\\)", "")) %>%
  filter(Role %in% valid_roles)
```

Capturing SRH and other Teams performance data.

For grabbing the performance data for the teams in IPL 2024 and the Performance data for SRH in the couple of previous seasons I scrapped the data from a page on statisticstimes.com which is a famous data source for finding any up-to-date detailed data related to India [9].

```r
# Function for extracting IPL points table from the page
extract_ipl_points_table <- function(url) {
  # Reading the webpage content
  page <- read_html(url)

  # Finding the table node that contains the caption "IPL 2024 Points Table"
  table_node <- page %>%
    html_nodes(xpath = "//caption[contains(text(), 'IPL 2024 Points Table')]/parent::table")

  # Extracting headers from thead
  headers <- table_node %>%
    html_node("thead") %>%
    html_nodes("th") %>%
    html_text(trim = TRUE)

  # Extracting data from tbody
  data <- table_node %>%
    html_node("tbody") %>%
    html_nodes("tr") %>%
    html_nodes("td") %>%
    html_text(trim = TRUE)

  # Reshaping the data into a data frame
  data_matrix <- matrix(data, ncol = length(headers), byrow = TRUE)
  table_data <- as.data.frame(data_matrix, stringsAsFactors = FALSE)
  colnames(table_data) <- headers

  # Adjusting the column names based on the inspection
  table_data <- table_data %>%
    select(`Team`, `W`, `L`, `N/R`) %>%
    rename(Win = W, Lose = L, `No result` = `N/R`) %>%
    mutate(
      Team = str_extract(Team, "^[^\\(]+") %>% str_trim(),
      Win = as.numeric(Win),
      Lose = as.numeric(Lose) * -1,# Making Lose values negative
      `No result` = as.numeric(`No result`)
    )

  return(table_data)
}

# URL of the page containing the table
url <- "https://statisticstimes.com/sports/ipl/all-ipl-points-table.php"

# Extracting the IPL 2024 Points Table data
points_table_2024 <- extract_ipl_points_table(url)
```

Here the function reads the webpage content using the rvest package, locates the table containing the caption “IPL 2024 Points Table” using an XPath query, and extracts the headers and data from the table. The data is then reshaped into a data frame, and specific columns (Team, W, L, N/R) are selected and renamed to Win, Lose, and No result. The Lose values are made negative, and all numerical values are converted to numeric types. The cleaned data frame is returned. The script then uses this function to extract the points table data from the given URL, storing the result in the points_table_2024 data frame.

Preparing data for visualizations

As I am planning to present the data for the teams in the form of a stacked bar chart, it needs to be converted into long-form to make it easy to plot.

```r
# Transforming the data into long format
points_table_long <- points_table_2024 %>%
  pivot_longer(cols = c(Win, Lose, `No result`),
               names_to = "Result",
               values_to = "Value") %>%
  arrange(factor(Team, levels = unique(Team)))
```

Capturing SRH performance data over 3 seasons

Data was captured from the same page as all teams performance data on statisticstimes.com and is processed for ease of understanding [9].

```r
extract_net_rr <- function(url, year) {
  # Read the webpage content
  page <- read_html(url)

  # Find the table node that contains the caption with the respective year
  table_node <- page %>%
    html_nodes(xpath = paste0("//caption[contains(text(), 'IPL ", year, " Points Table')]/parent::table"))

  if (length(table_node) == 0) {
    stop("Table not found with caption 'IPL ", year, " Points Table'")
  }

  # Extract headers from thead
  headers <- table_node %>%
    html_node("thead") %>%
    html_nodes("th") %>%
    html_text(trim = TRUE)

  # Extract data from tbody
  data <- table_node %>%
    html_node("tbody") %>%
    html_nodes("tr") %>%
    map(~ html_nodes(.x, "td") %>% html_text(trim = TRUE)) %>%
    map(~ set_names(.x, headers)) %>%
    bind_rows()

  # Find the Net RR for SRH
  net_rr <- data %>%
    filter(str_detect(Team, "SRH|Sunrisers Hyderabad")) %>%
    pull(`Net RR`)

  # Return the year and Net RR as a data frame
  return(data.frame(Year = year, `Run Rate` = net_rr, stringsAsFactors = FALSE))
}

# URL of the page containing the points tables
url_points <- "https://statisticstimes.com/sports/ipl/all-ipl-points-table.php"

# Extracting the Net RR for SRH for 2022 and 2023
net_rr_2022 <- extract_net_rr(url_points, 2022)
net_rr_2023 <- extract_net_rr(url_points, 2023)
net_rr_2024 <- extract_net_rr(url_points, 2024)

# Combine the data frames
net_rr_combined <- bind_rows(net_rr_2022, net_rr_2023, net_rr_2024) %>%
  rename_with(~ gsub("[.]", " ", .)) %>%
  mutate(`Run Rate` = as.numeric(gsub("[^0-9.-]", "", `Run Rate`)))
```

Here the above function reads the webpage content using the rvest package, locates the table using an XPath query, and extracts headers and data. It reshapes the data into a data frame, filters for SRH's Net RR, and returns a data frame with the year and Net RR. The script then uses this function to get the Net RR for SRH for 2022, 2023, and 2024, combines the results, and ensures the Run Rate column values are numeric. The final data frame net_rr_combined is displayed, showing the Net RR for SRH across the specified years.

References

[1] CricTracker. (n.d.). Copyright Notice. CricTracker. https://www.crictracker.com/copyright-notice/ (https://www.crictracker.com/copyright-notice/)

[2] CricTracker. (n.d.-b). DMCA. CricTracker. https://www.crictracker.com/dmca/ (https://www.crictracker.com/dmca/)

[3] Terms of use - StatisticsTimes.com. (n.d.). https://statisticstimes.com/termsofuse.php (https://statisticstimes.com/termsofuse.php)

[4] Indian Premier League Official Website. (n.d.). https://www.iplt20.com/about/media-accreditation-terms--conditions?id=10&page=1 (https://www.iplt20.com/about/media-accreditation-terms--conditions?id=10&page=1)

[5] Nair, K. (2022, April 2). SRH Players List 2022: Complete Sunrisers Hyderabad squad and players list for IPL 2022. CricTracker. https://www.crictracker.com/srh-players-list-2022-complete-sunrisers-hyderabad-squad-and-players-list-for-ipl-2022/ (https://www.crictracker.com/srh-players-list-2022-complete-sunrisers-hyderabad-squad-and-players-list-for-ipl-2022/)

[6] Staff, C. (2023, December 19). SRH Team 2023 Player List: Complete Sunrisers Hyderabad (SRH) squad and players list for IPL 2023. CricTracker. https://www.crictracker.com/cricket-news/srh-team-2023-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2023/ (https://www.crictracker.com/cricket-news/srh-team-2023-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2023/)

[7] S, A. P. (2024, January 3). SRH Team 2024 Player LIST: Complete Sunrisers Hyderabad (SRH) squad and players list for IPL 2024. CricTracker. https://www.crictracker.com/cricket-news/srh-team-2024-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2024/ (https://www.crictracker.com/cricket-news/srh-team-2024-player-list-complete-sunrisers-hyderabad-srh-squad-and-players-list-for-ipl-2024/)

[8] SunRisers Hyderabad | IPL 2024 Team analysis and players. (n.d.). https://www.iplt20.com/teams/sunrisers-hyderabad (https://www.iplt20.com/teams/sunrisers-hyderabad)

[9] IPL points Table (2008-2024) - StatisticsTimes.com. (n.d.). https://statisticstimes.com/sports/ipl/all-ipl-points-table.php (https://statisticstimes.com/sports/ipl/all-ipl-points-table.php)