Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH

Instytut Informatyki

Studia Podyplomowe
Big Data - przetwarzanie i analiza dużych zbiorów danych

PRACA KOŃCOWA

Paweł Safuryn

Wykorzystanie analizy big data na statystykach meczów tenisowych w celu poprawy przygotowania i wyników zawodnika

Opiekun pracy
mgr inż. Damian Warszawski

Warszawa, 2024

STRESZCZENIE

Analityka danych w tenisie ziemnym jest tematem stosunkowo nowym i nierozwiniętym pomimo ogromnej ilości szczegółowych danych zbieranych podczas meczów profesjonalnych rozgrywek. Celem pracy jest zbadanie potencjału analizy ogólnodostępnych statystyk meczowych w celu poprawy przygotowania i wyników zawodnika. Surowe dane (262 pliki CSV o całkowitym rozmiarze 395,7 MiB) zawierające wysokopoziomowe statystyki meczów zostały pozyskane z Internetu. Dane zostały oczyszczone, przetworzone i przeanalizowane z użyciem narzędzi big data takich jakich AWS CLI, Amazon S3, AWS Glue, Amazon EMR, czy PySpark. Cały kod został udostępniony w repozytorium na GitHubie (https://github.com/safurynp/pw-big-data-final-project). Praca analityczna skupiła się na zrozumieniu wpływu rodzaju tenisowej nawierzchni na wyniki zawodnika, analizie serwisu na różnych poziomach rankingu i wśród zawodników o różnym wzroście oraz analizie wpływu zmęczenia na wyniki zawodników. Użyteczność analizy tenisowych danych została wykazana pomimo użycia stosunkowo małego zbioru danych w kontekście big data. Rozszerzenie tego zbioru i przeprowadzenie analiz na większych i bardziej różnorodnych danych (np. historia meczu punkt po punkcie, nagrania wideo meczów, informacje z systemu Hawk-Eye) ma potencjał zrewolucjonizować podejście trenerów, organizacji i samych zawodników do ich tenisowego rozwoju.

Słowa kluczowe: tenis, big, data, analiza

Using big data analysis on statistics of tennis matches to improve player preparation and performance

Data analysis in tennis is a relatively new and underdeveloped topic despite huge amount of detailed data collected during professional tennis matches. The goal of this work is exploring the potential of analysing widely-available tennis match statistics in order to improve player preparation and performance. Raw data (262 CSV files with total size of 395,7 MiB) containing high-level match statistics were obtained from the internet. The data was cleaned, transformed and analysed with the use of big data tools like AWS CLI, Amazon S3, AWS Glue, Amazon EMR or PySpark. All code was made publicly available on a GitHub repository (https://github.com/safurynp/pw-big-data-final-project). The analytical work focused on understanding the effect of tennis surface on player performance, analysing serve quality for players with different ranking and height, and analysing how player tiredness affects their performance. The usefulness of tennis data analysis has been demonstrated despite the use of a relatively small dataset in the context of big data. Expanding this dataset and conducting analyses on larger and more diverse datasets (such as point-by-point match history, video recordings of tennis matches, data from the Hawk-Eye system) has the potential to revolutionise the ways in with coaches, organisations, and players themselves approach their tennis development.

Keywords: tennis, big, data, analysis

Spis treści

Akronimy

API Application Programming Interface

ATP Association of Tennis Professionals

AWS Amazon Web Services

CLI Command-line Interface

EMR Elastic MapReduce

ETL Extract, Transform, Load

GUI Graphical User Interface

ID Identification

MiB Mebibajt (1 048 576 bajtów)

ML Machine Learning

WTA Women's Tennis Association

1 Wstęp

Tenis ziemny jest sportem indywidualnym, w którym nawet najmniejsza różnica może decydować o zwycięstwie. Zdarza się, że zwycięzca meczu zdobywa mniej punktów niż przegrany. Możliwe jest to dzięki (nieco skomplikowanym) zasadom punktacji. W tenisie, podobnie jak w innych sportach, analityka danych może właśnie stanowić tę kluczową różnicę między zwycięstwem a porażką. Jednak, w przeciwieństwie do innych sportów (np. baseball, piłka nożna), analityka danych w tenisie jest tematem stosunkowo nowym i nierozwiniętym pomimo ogromnej ilości szczegółowych danych zbieranych podczas meczów profesjonalnych rozgrywek. Powodem może być to, że tenis jest dosyć drogim sportem indywidualnym (w zespole zawodnika zazwyczaj nie ma miejsca dla analityka danych) o tradycyjnym wizerunku, który skutecznie zniechęca organizacje tenisowe, trenerów i zawodników do jakichkolwiek zmian. Niemniej jednak danych jest dużo i potencjał do zmian istnieje. Do takich danych należą między innymi:

- Wysokopoziomowe statystyki meczów, np. wynik (sety, gemy), liczba asów, podwójnych błędów, wygranych punktów po pierwszym i drugim serwisie, itd.
- Szczegółowe statystyki meczów, np. historia meczu punkt po punkcie.
- Nagrania wideo meczów tenisowych, które mogą zostać poddane analizie, np. śledzenie piłki i ruchu zawodników.
- Informacje z systemu Hawk-Eye (elektroniczny system umożliwiający rozstrzygnięcie kontrowersyjnych punktów), np. trajektoria piłki, prędkość piłki, rotacja piłki itd.
- Warunki fizyczne zawodników.

Niestety większość szczegółowych danych nie była i nadal nie jest dostępna publicznie. Do tej pory, dostęp do tych danych był dostępny za opłatą i tylko dla profesjonalnych zawodników. Poza samym dostępem do danych dochodzą koszty zatrudnienia analityka danych, co sprawia, że zaawansowana analiza danych pozostaje zarezerwowana głównie dla najbardziej zamożnych graczy. Dopiero w drugiej połowie roku 2023 zaczęły pojawiać się głosy, że organizacje tenisowe (np. ATP) otwierają się na udostępnianie (w ograniczonym zakresie) danych z systemów takich jak Hawk-Eye [1].

Główną korzyścią analizy tenisowych danych jest to, że pozwala ona zauważyć powtarzalne wzorce, których człowiek nie jest w stanie dostrzec podczas oglądania meczów na żywo lub na nagraniu wideo. Pozwala to zawodnikom (i ich trenerom) na, między innymi:

- Zrozumienie słabych stron zawodnika i przygotowanie odpowiedniego treningu.
- Zrozumienie słabych i mocnych stron przeciwnika i opracowanie odpowiedniej taktyki na mecz.
- Odpowiednie planowanie startów turniejowych i przerw między turniejami.

Ponadto szczegółowa analiza danych może pomóc narodowym organizacjom tenisowym w identyfikowaniu młodych talentów i efektywnym rozdysponowywaniu środków na ich rozwój.

1.1 Cel i zakres pracy

Celem pracy jest, jak sugeruje tytuł („Wykorzystanie analizy big data na statystykach meczów tenisowych w celu poprawy przygotowania i wyników zawodnika"), zbadanie potencjału analizy ogólnodostępnych statystyk meczowych w celu poprawy przygotowania i wyników zawodnika. Ze względu na trudności z pozyskaniem szczegółowych danych (opisane wcześniej) źródłem są surowe dane (wysokopoziomowe statystyki meczów, rankingi i informacje o zawodnikach) zebrane przez tenisowego pasjonata [2, 3]. Zbieranie tych danych opisane jest w sekcji 4.1. Używając tych danych, praca skupia się na:

- Rozpoznaniu najlepszych i najgorszych nawierzchni (kort twardy, kort ziemny, trawa, dywan) dla tenisisty.
- Analizie jakości serwisu na różnych poziomach rankingu.
- Analizie wyników zawodników różnych narodowości w zależności od poziomu zmęczenia.
- Analizie jakości i skuteczności pierwszego serwisu w zależności od wzrostu zawodnika.

Skutecznie przeprowadzone powyższe analizy mogą pomóc zawodnikom, trenerom i organizacjom tenisowym lepiej planować ich pracę i osiągać lepsze wyniki.

Do przetwarzania i analizy tenisowych zbiorów danych wykorzystane zostały metody, narzędzia i aplikacje do przetwarzania dużych zbirów danych (*big data*). Opisane są one w sekcji 3. Ważnym celem pracy było także zastosowanie kontroli wersji (Git) do konfiguracji i skryptów związanych z całym procesem przetwarzania i analizy danych. Dlatego, gdzie było to możliwe i praktyczne, użycie GUI zostało ograniczone i zamiast tego zastosowane zostały skrypty w Bashu i AWS CLI. Cały kod (włącznie z kodem LaTeXużytym do napisania tej pracy) dostępny jest na moim repozytorium pw-big-data-final-project na GitHubie.

2 Ustawienie środowiska

2.1 AWS Academy Learner Lab i AWS CLI

Projekt wykonywany jest z użyciem AWS Academy Learner Lab [61569] i AWS CLI version 2. AWS CLI jest konfigurowane poprzez skopiowanie ustawień AWS Academy Learner Lab (dostępnych pod przyciskiem „AWS Details") do pliku:

`~/.aws/credentials`

AWS Academy Learner Lab generuje tymczasowe dane uwierzytelniające do AWS CLI za każdym razem, gdy jest uruchomiony, co powoduje, że plik `~/.aws/credentials` musi być aktualizowany.

Opis architektury Big Data i indywidualnych technologii AWS wykorzystanych podczas projektu znajduje się w sekcji 3.

2.2 Narzędzia używane lokalnie

Poniżej znajduje się lista głównych narzędzi używanych lokalnie podczas projektu:

- Python (wersja 3.9) — główny język programowania używany do przetwarzania i analizy danych.
- Bash — używany do automatyzacji zadań, głównie tych wykorzystujących AWS CLI.
- MacTeX i LaTeX— używane do napisania tej pracy końcowej.
- GitHub i Git — kontrola wersji. Cały kod dostępny na repozytorium pw-big-data-final-project.
- Miniconda — zarządzanie środowiskiem wirtualnym oraz bibliotekami programistycznymi (np. do Pythona). Środowisko może być załadowane używając:

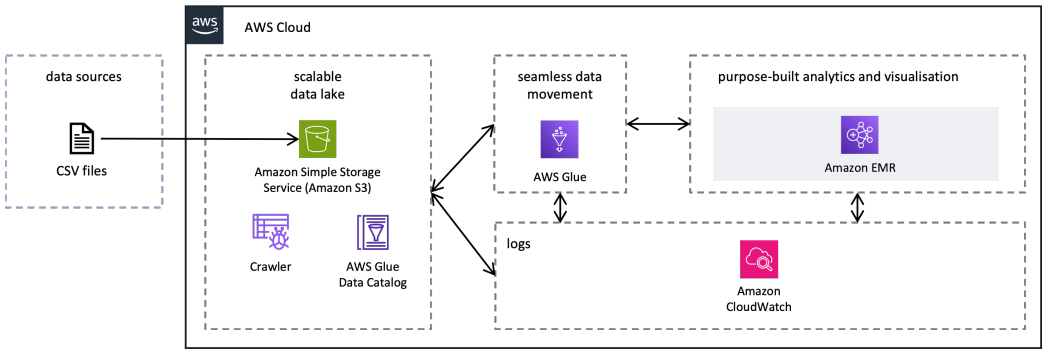
```
conda env create -f environment.yml
conda activate big-data-env
```

3 Architektura Big Data w AWS

Wykres 1 przedstawia ogólny zarys architektury Big Data w AWS użytej w projekcie.

Architektura ta składa się z następujących elementów:

1. **Źródła danych**: Surowe dane w formacie plików CSV.
2. **Skalowalny magazyn danych typu data lake**:
    - Amazon Single Storage Service (Amazon S3) używany do przechowywania surowych danych (bucket „tennis-stats-data”) oraz przetworzonych danych (bucket „processed-tennis-stats-data”).



Wykres 1: Architektura Big Data w AWS.

- AWS Glue Crawler używany do automatycznego aktualizowania metadanych w katalogu danych.
- AWS Glue Data Catalog — katalog danych używany do przechowywania metadanych o danych w Amazon S3.

3. **Zarządzanie przepływem danych**: AWS Glue używany do eksploracji i przetwarzania danych za pomocą Notebooków AWS Glue i PySparka (Python API do Apache Spark).

4. **Analityka i wizualizacja danych**: Amazon EMR Notebook używany do analizy danych i tworzenia wykresów. Notebook używał PySparka i był uruchamiany na wcześniej przygotowanym klastrze Amazon EMR.

5. **Logi**: Amazon CloudWatch używany do sprawdzania logów z Crawlera i procesów ETL w AWS Glue i klastrów Amazon EMR.

4 Zbieranie danych

4.1 Pozyskanie surowych danych

Surowe tenisowe dane pozyskane są z dwóch repozytoriów na GitHubie Jeffa Sackmanna.

- Dla profesjonalnego tenisa męskiego (ATP) [2].
- Dla profesjonalnego tenisa żeńskiego (WTA) [3].

Dane zostały pozyskane przez autora najprawdopodobniej (bazując na opisie repozytoriów) poprzez ekstrakcję danych (*web scraping*) z Wikidata i oficjalnych stron organizacji ATP i WTA. Dane są dostępne w formacie CSV. Dane dla ATP i WTA mają ten sam format i zawierają:

- Statystyki z singlowych meczów tenisowych z głównego cyklu turniejów ATP i WTA (np. Wielkie Szlemy, ATP/WTA 1000). Dostępne są dla każdego roku od 1968 (początek ery Open) do 2023. Statystyki dla każdego roku są zawarte w osobnym pliku CSV.

- Statystyki z singlowych meczów tenisowych z etapów eliminacji oraz z niższych rangą turniejów ATP i WTA (np. Challengers, Futures). Te dostępne są od lat 1968/1978/1991 (w zależności od rodzaju turnieju) do 2023. Statystyki dla każdego roku są zawarte w osobnym pliku CSV.

- Lista graczy tenisa ziemnego, którzy grali w profesjonalnym turnieju ATP lub WTA wraz z podstawowymi informacjami na ich temat.

- Informacje o profesjonalnym rankingu ATP i WTA (aktualizowane co tydzień) od lat 70 do 2023. Statystyki dla każdej dekady i roku 2023 są zawarte w osobnym pliku CSV.

W sumie są to 262 pliki CSV o całkowitym rozmiarze 395,7 MiB. W sumie są to informacje o 1 581 121 meczach, 2 621 tygodniach rankingów, 64 675 zawodnikach i 65 607 zawodniczkach. Repozytoria Jeffa Sackmanna na GitHubie zawierają również statystyki z meczów deblowych i amatorskich, ale te zostały pominięte w tym projekcie. Bardziej szczegółowy opis surowych danych znajduje się w załączniku A.

Te dane zostały wybrane z poniższych powodów:

- Zawierają wystarczająco szczegółowe informacje, aby przeprowadzić sensowne tenisowe analizy.

- Dane są ogólnodostępne i darmowe. Autor danych wymaga jedynie, aby zostały odpowiednio zacytowane.

- Dane są w powszechnym formacie CSV, co ułatwia ich początkową interpretację i przetwarzanie. Jednocześnie nie są to dane oczyszczone i wymagają obróbki (co pozwala an demonstrację odpowiednich technik przetwarzania danych big data).

- Rozmiar danych (395,7 MiB) nie jest typowym przykładem big data (w dzisiejszych czasach są to zazwyczaj petabajty lub eksabajty danych), ale jest na tyle duży, że przetwarzanie ich lokalnie mogłoby być czasochłonne i problematyczne. Jednocześnie rozmiar danych jest na tyle mały, że ich obróbka i analiza w ramach AWS Academy Learner Lab ze studolarowym ($100) budżetem jest możliwa.

Inne źródła tenisowych danych również były rozważane. Między innymi było to użycie oficjalnych API od organizacji tenisowych, jednak takie nie były dostępne. Innym pomysłem była automatyczna ekstrakcja danych (*web scraping*) z użyciem istniejących skryptów [4], jednak skrypty nie działały po niedawnym przeprojektowaniu strony ATP i ograniczenia czasowe nie pozwoliły na tworzenie ich od nowa. Alternatywą było również pozyskanie niedawno udostępnionych oficjalnych danych od ATP [5] jednak były one dostępne tylko dla profesjonalnych graczy.

4.2 Przesyłanie danych do chmury AWS

Surowe pliki CSV opisane powyżej zostały przesłane do chmury AWS w krokach opisanych poniżej:

1. Trzy pliki CSV posiadające problem z kodowaniem UTF-8 (opisane bardziej szczegółowo w sekcji 5) zostały ręcznie oczyszczone.

2. Do przechowywania plików CSV stworzony został bucket S3 o nazwie „tennis-stats-data" z użyciem AWS CLI — skrypt:

   `aws_cli/create_s3_raw_data.sh`

3. Pliki CSV zostały przesłane do bucketu S3 z użyciem AWS CLI — skrypty:

   `aws_cli/upload_atp_files_to_s3.sh`
   `aws_cli/upload_wta_files_to_s3.sh`

4. Pliki zorganizowane w 6 folderach w buckecie S3 (decyzja o takim podziale była podjęta po wstępnej eksploracji danych): atp_matches, atp_players, atp_rankings, wta_matches, wta_players, wta_rankings. Skrypt:

   `aws_cli/reorganise_s3.sh`

5 Eksploracja danych

Kolejnym krokiem była eksploracja danych w celu zrozumienia struktury danych i ich jakości. Eksploracja danych została przeprowadzona w dwóch etapach:

- Lokalna eksploracja danych — z użyciem narzędzi takich jak Python, Pandas, Jupyter Notebook.
- Eksploracja danych w chmurze — z użyciem AWS Glue Studio Notebook i PySparka oraz AWS Glue Crawler i AWS Glue Data Catalog.

5.1 Lokalna eksploracja danych

Lokalnie dane były eksplorowane w Jupyter Notebooku poniżej:

`local/local_explore.ipynb`

Pojedyncze pliki CSV o różnych nazwach zostały wczytane do Pandas DataFrame, aby sprawdzić nazwy ich kolumn, typy danych oraz przykładowe dane w nich zawarte. Unikalne wartości niektórych kolumn (np. `round`) również zostały sprawdzone. Wczytane były:

- `atp_matches_2020.csv`
- `atp_matches_futures_2020.csv`
- `atp_matches_qual_chall_2020.csv`
- `atp_players.csv`
- `atp_rankings_20s.csv`
- `wta_matches_qual_itf_2015.csv`
- `wta_matches_qual_itf_2016.csv`
- `wta_matches_qual_itf_2017.csv`

Ostatnie trzy pliki CSV miały problem z kodowaniem UTF-8. Skrypt w Jupyter Notebooku sprawdził poprawność kodowania wszystkich plików CSV i wykazał, że tylko te 3 pliki miały problem. Następnie problematyczne wiersze zostały zidentyfikowane. Problematyczny był tylko jeden znak w każdym z tych plików. Problematyczny znak został ręcznie usunięty i nowe pliki CSV zapisane w folderze `tennis_wta_modified`.

5.2 Eksploracja danych w chmurze

Po przesłaniu danych do chmury AWS Glue Crawler został skonfigurowany i uruchomiony za pomocą skryptów poniżej:

```
aws_cli/crawler_setup_raw.sh
aws_cli/crawler_run_raw.sh
```

Crawler automatycznie stworzył metadane (następnie zapisane w AWS Glue Data Catalog), które umożliwiają AWS Glue i innym usługom przeglądanie informacji w S3 w postaci bazy danych z tabelami. Zapewnia to uporządkowany i zorganizowany widok zasobów danych i spójne i dostępne metadane dla różnych narzędzi. Na podstawie bucketu S3 „processed-tennis-stats-data" Crawler wykrył 6 typów schematów i stworzył 6 tabeli w AWS Glue Data Catalog. Zawierają one informacje wyczytane z plików CSV takie jak tytuły kolumn i typ danych w kolumnach.

Następnie AWS Glue Studio Notebook i PySpark zostały użyte do kolejnej eksploracji danych (tym razem na całej bazie danych) oraz do zdefiniowania wstępnych transformacji. Notebook dostępny jest w pliku:

```
aws_glue/explore.ipynb
```

6 Przetwarzanie i czyszczenie danych

Na podstawie eksploracji danych lokalnie i w chmurze oraz wstępnych transformacji zdefiniowanych w AWS Glue Studio Notebook stworzony został ETL Job w AWS Glue — skrypt używający PySparka do oczyszczenia i transformacji surowych danych i zapisu danych przetworzonych. Są one opisane poniżej:

1. Rozwiązanie konfliktu w typie danych kolumny `draw_size` między danymi ATP i WTA.

2. Rozwiązanie konfliktu unikalnych ID graczy między danymi ATP i WTA. Niektóre unikalne identyfikatory się powtarzały.

3. Połączenie danych ATP i WTA — stworzenie 3 tabel (matches, rankings, players) z 6 tabel (atp_matches, atp_rankings, atp_players, wta_matches, wta_rankings, wta_players). Dodanie kolumny `tour` opisującej czy dane pochodzą z cyklu rozgrywek ATP czy WTA.

4. Zmiana typu danych kolumn opisujących daty z „bigint" na „date".

5. Interpretacja kolumny `score` opisującej wynik meczu tenisowego w gemach (np. 7-6 4-6 6-2), aby uzyskać informacje o gemach wygranych przez zwycięzcę i przegranego (odpowiednio 17 i 14 w tym przykładzie) i zapis tych informacji w nowych kolumnach.

6. Zapis przetworzonych danych w nowym buckecie S3 „processed-tennis-stats-data" w formacie Parquet zapewniającym wydajne przechowywanie i wyszukiwanie danych oraz wydajną kompresję danych. Skrypt do stworzenia nowego bucketu S3:

   `aws_cli/create_s3_processed_data.sh`

Cały skrypt do tego ETL Job w AWS Glue dostępny jest w pliku:

`aws_glue/etl.py`

Kolejnym krokiem było stworzenie i uruchomienie nowego Crawlera, który automatycznie zaktualizował metadane w AWS Glue Data Catalog na podstawie nowych danych:

`aws_cli/crawler_setup_processed.sh`
`aws_cli/crawler_run_processed.sh`

7 Analiza danych

Posiadając przetworzone dane w S3 oraz ich metadane w AWS Glue Data Catalog, następnym krokiem było przeprowadzenie analizy danych w celu wyciągnięcia wniosków, które przyczynią się do poprawy przygotowania i wyników zawodnika.

7.1 Narzędzia i konfiguracja

Aby skutecznie przeprowadzić analizę danych big data, odpowiednie narzędzia musiały być skonfigurowane. Tym narzędziem było Amazon EMR skonfigurowane w nowej konsoli AWS w następujący sposób (z pomocą instrukcji zawartych w [6]):

1. Stworzenie nowego klastra EMR z użyciem AWS CLI — skrypt:

   `aws_cli/create_emr_cluster.sh`

Domyślne ustawienia klastra z pakietem aplikacji „Spark Interactive" były odpowiednie. Nowe klastery musiały być tworzone po wygaśnięciu starych (np. po dłuższej bezczynności klastra).

2. Stworzenie EMR Studio — konieczne do stworzenia EMR Notebook w nowej konsoli AWS. Wykonane z użyciem GUI.

3. Stworzenie EMR Workspace (czyli nowego EMR Notebook) — interaktywne środowisko do analizy danych z użyciem PySparka. Do stworzonego EMR Workspace podłączany był istniejący i aktywny klaster EMR z ustawieniem „Launch in Jupyter" i domyślnymi grupami bezpieczeństwa. Wszystko wykonane z użyciem GUI.

4. Instalacja dodatkowych bibliotek Pythona (numpy, pandas, matplotlib) na klastrze EMR poprzez EMR Notebook według instrukcji w [7].

Sama analiza danych wykonana została w EMR Notebook z użyciem PySparka oraz bibliotek Pythona: numpy, pandas i matplotlib. Notebook dostępny jest w pliku:

`aws_emr/analysis.ipynb`

7.2 Rozpoznanie najlepszych i najgorszych nawierzchni

Pierwszym prostym przykładem analizy, która może pomóc zawodnikom zidentyfikować obszar gry wymagający poprawy i lepiej zaplanować turniejowe starty jest rozpoznanie najlepszych i najgorszych nawierzchni dla zawodnika. W tenisie istnieją 4 główne nawierzchnie — kort twardy, kort ziemny (mączka), kort trawiasty i dywan (ten ostatni jest już rzadko spotykany w profesjonalnych rozgrywkach). Każda z nich ma swoje specyficzne cechy, które wpływają na styl gry. Na przykład kort trawiasty jest szybki i sprzyja grze serwisowej, a mączka jest wolna i sprzyja grze z głębi kortu. Zawodnicy często mają swoje ulubione nawierzchnie, na których osiągają najlepsze wyniki. Jednakże, aby osiągnąć sukces w profesjonalnym tenisie, zawodnik musi być w stanie grać na każdej nawierzchni. Dlatego, aby poprawić wyniki zawodnika, ważne jest, aby zidentyfikować nawierzchnie, na których zawodnik gra najlepiej i najgorzej i:

- Odpowiednio zaplanować treningi, aby poprawić grę na gorszych nawierzchniach lub

- Odpowiednio zaplanować starty turniejowe, aby grać więcej na nawierzchniach, które sprzyjają zawodnikowi.

Różnice w wynikach na różnych nawierzchniach zobrazowane są w tabeli 1 na podstawie danych z meczów profesjonalnego gracza Daniiła Miedwiediewa (nr 3 w rankingu ATP na 11 stycznia 2024). Na podstawie tych danych

Nawierzchnia	**Wygrane mecze**	**Przegrane mecze**	**Stosunek w/p**
Kort twardy	348	124	2.81
Dywan	12	5	2.4
Trawa	50	22	2.27
Mączka	71	43	1.65

Tabela 1: Wyniki Daniiła Miedwiediewa na różnych nawierzchniach.

zespół Daniiła mógłby zwiększyć częstotliwość treningów na jego najgorzej nawierzchni — mączce. Ewentualnie zespół mógłby zaplanować starty głównie na kortach twardych, aby zwiększyć szanse na wygrane i większe zarobki. Podobne analizy mogą być szczególnie przydatne dla graczy z niższych rankingów, którzy nie grają w głównych (często obowiązkowych) turniejach i przez to mają większą elastyczność w planowaniu turniejowych startów na różnych nawierzchniach oraz mają więcej czasu na treningi.

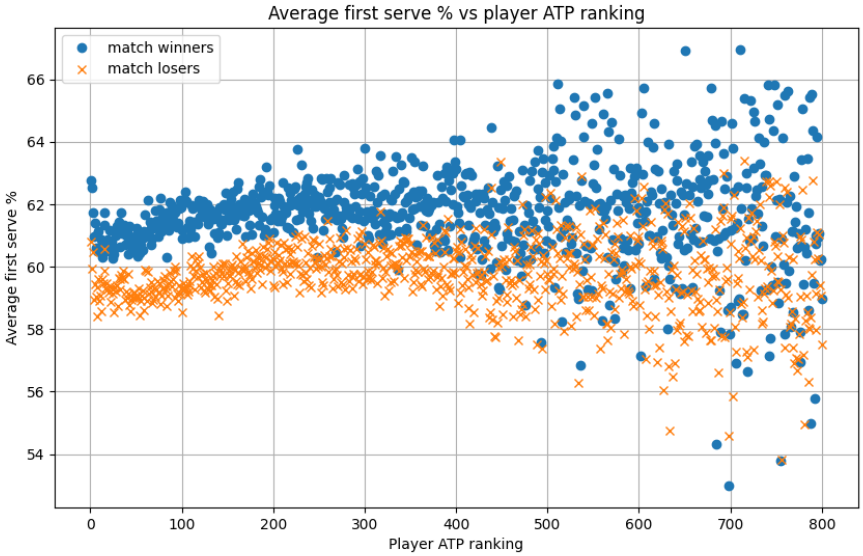
7.3 Analiza jakości serwisu na różnych poziomach rankingu

Serwis w tenisie ziemnym to uderzenie rozpoczynające każdy punkt. Jest to jeden z najważniejszych elementów gry i ma ogromne znaczenie w profesjonalnych rozgrywkach. Skuteczny, regularny i również różnorodny serwis daje serwującemu dużą przewagę w meczu. Zawodnicy spędzają dużo czasu nad doskonaleniem swojego serwisu na każdym etapie rozwoju, od juniorów do profesjonalnych zawodników w końcówce swojej kariery. W tenisie ziemnym zawodnik ma dwa serwisy:

1. **Pierwszy serwis**: Jego celem jest zazwyczaj zdobycie punktu bezpośrednio, zawodnik uderza piłkę mocno i precyzyjnie, ale ryzykując nietrafienie.

2. **Drugi serwis**: Jeśli pierwszy serwis jest nietrafiony, drugi serwis ma na celu zapewnienie drugiej szansy na rozpoczęcie wymiany, minimalizując ryzyko straty punktu.

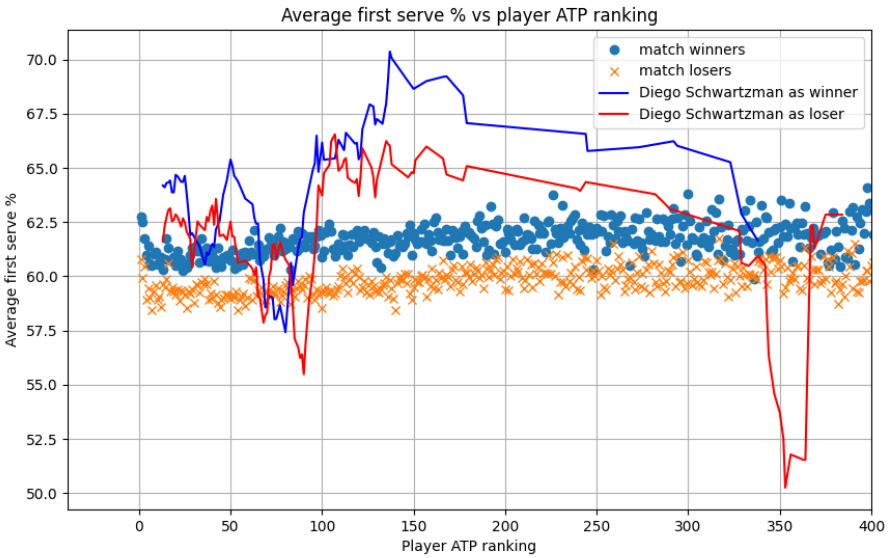
Znaczenie pierwszego serwisu na różnych poziomach rozgrywek zobrazowane jest na wykresie 2. Pierwszym jasnym wnioskiem jest to, że im większy procent trafionych serwisów, tym większa szansa na zwycięstwo na każdym poziomie rozgrywek. Ciekawą obserwacją jest także to, że najlepsi gracze



Wykres 2: Znaczenie pierwszego serwisu na różnym poziomie rozgrywek.

(top 100) nie trafiają w kort pierwszym serwisem częściej niż gracze z niższych poziomów. Wręcz przeciwnie, średnio trafiają trochę mniej. Może to być związane z tym, że grają przeciwko lepszym zawodnikom, co zmusza ich do podejmowania większego ryzyka i z tym że równie skutecznie potrafią wygrywać punkty po drugim serwisie. Natomiast wyraźna jest różnica w regularności pierwszego serwisu między top 300 a resztą rankingu. Procent trafionych pierwszych serwisów dla pierwszej grupy oscyluje między 58% a 64%, dla drugiej grupy jest to między 53% a 67%. Wynika z tego, że regularność pierwszego serwisu jest kluczowa, aby wygrywać turnieje i piąć się w rankingu ATP.

Podobne dane mogą być przeanalizowane dla konkretnego zawodnika i jego awansów lub spadków w światowym rankingu. Wykres 3 pokazuje jakość pierwszego serwisu stosunkowo niskiego (170 cm) gracza — Diego Schwartzmana. Wyraźnie widać, że rankingową wspinaczkę z poziomu 100-300 (turnieje Challenger) do poziomu top 100 (główny cykl ATP — Wielkie Szlemy i turnieje ATP 1000) Diego zawdzięcza bardzo wysokiej skuteczności pierwszego serwisu (mierzonej jako procent trafionych serwisów). Wysoki procent trafionych pierwszych serwisów może być szczególnie ważny dla niższych zawodników, których drugi serwis bywa słabszy i nie daje znaczącej przewagi przez uwarunkowania fizyczne (uderzanie piłki niżej). Analizując jakość i skuteczność serwisu graczy o podobnych warunkach fizycznych i osiągają-



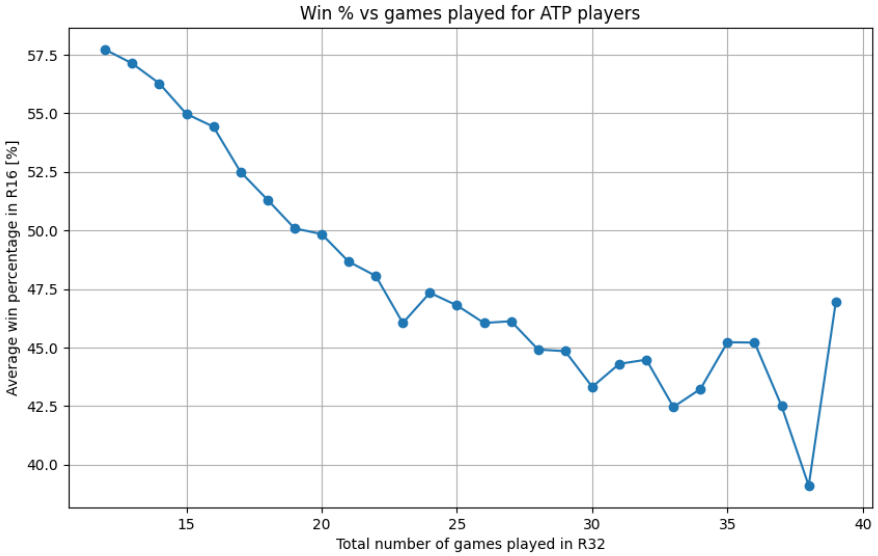
Wykres 3: Pierwszy serwis Diego Schwartzmana na tle innych zawodników na różnym poziomie rozgrywek.

cych sukcesy i monitorując swój własny serwis, rozwijający się gracze mogą zacząć rozwijać to uderzenie w odpowiednim kierunku i odpowiednio zaplanować serwisową strategię, np. zwiększając procent trafionych pierwszych serwisów kosztem siły lub precyzji.

7.4 Analiza wyników zawodników w zależności od poziomu zmęczenia

Kondycja fizyczna jest ważnym atrybutem profesjonalnego tenisisty. W typowym turnieju tenisowym (drabinka na 32 graczy), aby wygrać turniej, gracz musi odnieść zwycięstwo w 5 meczach w ciągu paru dni. Wyznacznikiem ilości wysiłku włożonego w mecz (tj. zmęczenia) może być całkowita liczba rozegranych gemów w meczu. W meczu tenisowym do 2 wygranych setów (również nazywane *best of 3*) minimalna liczba rozegranych gemów to 12 (wynik 6-0 6-0), a maksymalna to 39 (wynik 7-6 6-7 7-6). Wykres 4 przedstawia procent wygranych meczów w drugiej rundzie (R16) typowego turnieju w zależności od liczby rozegranych gemów w rundzie pierwszej (R32).

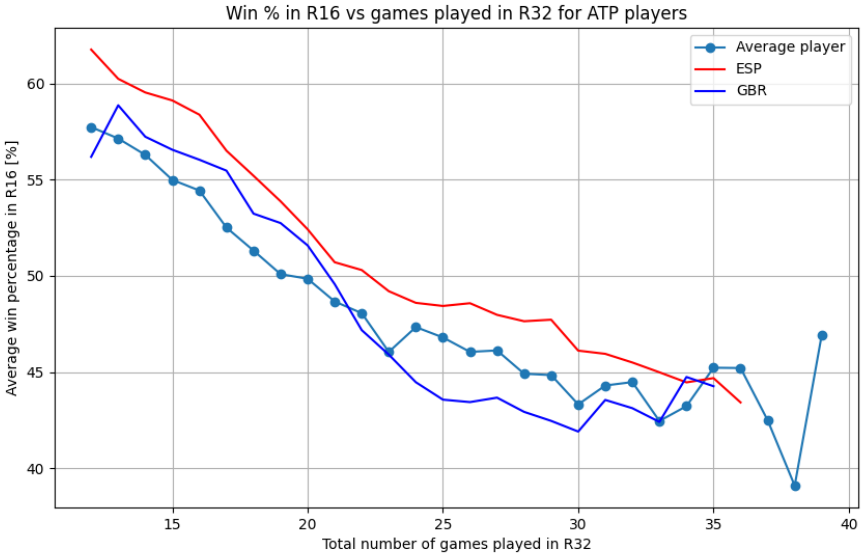
Jak widać powyżej, zmęczenie znacząco wpływa na wyniki zawodników. Zawodnik, który rozegrał w rundzie pierwszej tylko 12 gemów (wynik 6-0 6-0) ma średnio 57,7% szans na wygranie drugiej rundy. Natomiast zawod-



Wykres 4: Procent wygranych meczów w drugiej rundzie (R16) w zależności od rozegranych gemów w rundzie pierwszej (R32).

nik, który rozegrał długi 38-gemowy pojedynek pierwszej rundy (np. wynik 7-6 5-7 7-6) ma średnio tylko 39,1% szans na wygraną w kolejnej rundzie. Równocześnie warto zauważyć, że liczba rozegranych gemów nie jest idealnym miernikiem zmęczenia. Mecze bardzo wysokich graczy posiadających świetny serwis często kończą się wysokim wynikiem (np. 7-6 6-7 7-6) ale niekoniecznie są wyczerpujące fizycznie przez brak długich wymian (piłki są często wygrywane bezpośrednio po serwisie). To może być powodem wahań w procencie wygranych meczów w R16 dla liczby rozegranych gemów w R32 powyżej 30 (widocznych na wykresie 4).

Na wykresie 5 przedstawiono te same dane, ale z dodatkowym podziałem na pochodzenie zawodników — przedstawieni są Brytyjczycy (GBR) i Hiszpanie (ESP). Dla meczów pierwszej rundy, które mogą być uznane za długie i męczące (między 20 a 35 gemów) wyraźnie widać różnicę w procencie wygranych meczów drugiej rundy między Brytyjczykami i Hiszpanami. Co więcej, Brytyjski tenisista wypada również gorzej od średniego gracza. Za przygotowanie fizyczne profesjonalnych tenisistów z Wielkiej Brytanii często odpowiada brytyjski związek tenisowy (LTA). Powyższe dane powinny stanowić podstawę do zmiany brytyjskich treningów kondycyjnych, które mogłyby zostać oparte na modelu hiszpańskim.



Wykres 5: Porównanie kondycji zawodników brytyjskich (GBR) i hiszpańskich (ESP) na tle średniej.

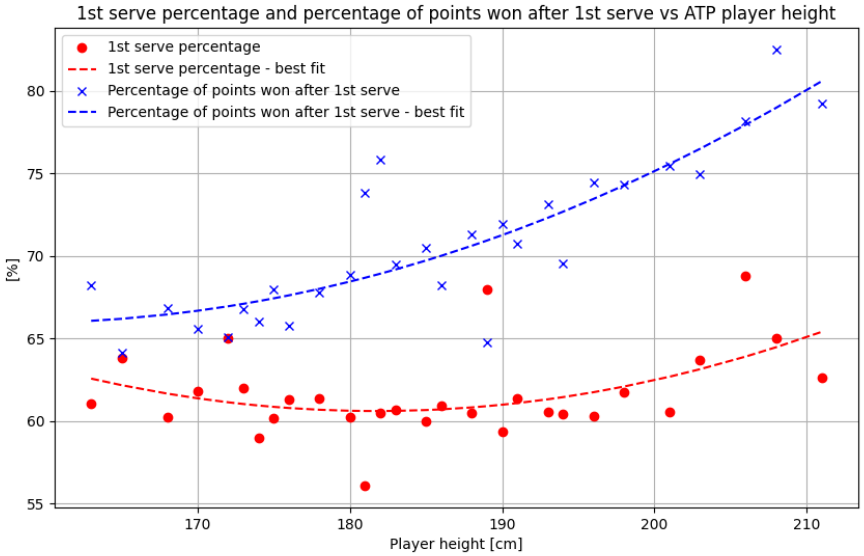
7.5 Analiza skuteczności pierwszego serwisu w zależności od wzrostu zawodnika

Znaczenie serwisu w tenisie i zależność jego skuteczności od wzrostu zawodnika zostały krótko wspomniane w poprzednich analizach. Wykres 6 pokazuje te zależności, bazując na danych. Skuteczność pierwszego serwisu wyrażona jest za pomocą dwóch metryk:

- Procent trafionych pierwszych serwisów (*1st serve percentage*)
- Procent wygranych punktów po trafionym pierwszym serwisie (*Percentage of points won after 1st serve*)

Wyraźnie widać, że trafianie pierwszym serwisem daje przewagę każdemu zawodnikowi (procent wygranych punktów $> 64\%$) jednak ta przewaga jest zdecydowanie większa dla graczy wyższych. Co ciekawe, procent trafionych pierwszych serwisów jest podobny dla graczy niższych i wyższych. Gracze średniego wzrostu trafiają pierwszym serwisem trochę mniej. Może to być związane z tym, jak warunki fizyczne kształtują styl gry, na przykład:

- Gracze niscy częściej trafiają pierwszym serwisem, aby uniknąć słabszego drugiego serwisu, który często jest atakowany.



Wykres 6: Skuteczność pierwszego serwisu w zależności od wzrostu gracza.

- Gracze wysocy częściej trafiają pierwszym serwisem, aby ograniczyć długość wymian ze względu na gorszą kondycję i mobilność.
- Gracze średniego wzrostu (na tle innych tenisistów) są bardziej wszechstronni i mogą sobie pozwolić na większe ryzyko i mniejszy procent trafień przy pierwszym serwisie.

Niestety brak bardziej szczegółowych danych w zbiorze danych analizowanym w tym projekcie nie pozwala na sprawdzenie wszystkich powyższych hipotez.

Niemniej jednak rozwijający się gracze powinni monitorować skuteczność swojego serwisu i swojej konkurencji w podobny sposób, aby upewnić się, że to uderzenie jest odpowiednio rozwijane, nie jest gorsze na tle aktualnej opozycji i zapewnia im przewagę w meczach.

8 Wnioski i zakończenie

Praca wykazała użyteczność analizy tenisowych danych w celu poprawy przygotowania i wyników zawodnika. Do przetwarzania i analizy danych skutecznie zostały użyte metody, narzędzia i aplikacje do przetwarzania dużych zbiorów danych (*big data*). Użyteczność analizy została wykazana pomimo użycia stosunkowo małego zbioru danych (395,7 MiB) w kontekście big data.

Rozszerzenie tego zbioru i przeprowadzenie analiz na większych i bardziej różnorodnych danych (np. historia meczu punkt po punkcie, nagrania wideo meczów, informacje z systemu Hawk-Eye) ma potencjał zrewolucjonizować podejście trenerów, organizacji i samych zawodników do ich tenisowego rozwoju.

8.1 Ulepszenia i dalsza praca

Potencjalne ulepszenia tej pracy i dalsza praca do wykonania:

- Rozszerzenie zbioru danych o:
  - Dane z automatycznej ekstrakcji danych ze stron organizacji ATP i WTA (*web scraping*).
  - Dane pozyskane z nagrań wideo meczów tenisowych (trajektoria piłki i ruch graczy w każdym punkcie) z użyciem biblioteki OpenCV. Ekstrakcja danych z nagrań wideo mogłaby być również przeprowadzona w czasie rzeczywistym z użyciem takich narzędzi jak YOLOv3 [8].

  Powyższe dane również mogłyby być przerabiane i analizowane za pomocą narzędzi big data.

- Użycie AWS Transfer Family to przesyłania plików CSV do S3. Zapewniłoby to solidniejsze i bardziej wszechstronne rozwiązanie w porównaniu z przesyłaniem bezpośrednim (np. zapewniłoby możliwość monitorowania przesyłu). Przysłanie wszystkich plików CSV do S3 zajmowało ponad 20 minut i mogło być podatne na błędy, których AWS Transfer Family pomaga uniknąć. Niestety AWS Transfer Family nie było dostępne w AWS Academy Learner Lab.

- Użycie AWS Lambda, aby reagować na nowe commity w repozytorium Jeffa Sackamanna (nowe dane) i automatycznie rozpoczynać cały proces ETL i analizy. AWS Lambda nie było użyte ze względu na ograniczenia czasowe.

- Stworzenie dashboardu z dostępem do przetworzonych danych i udostępnienie go trenerom i zawodnikom jako końcowy produkt oparty na danych i komunikacja wyników. Dashboard mógłby być stworzony z użyciem AWS QuickSight (oryginalny plan zakładał użycie tego narzędzia, ale nie było ono dostępne w AWS Academy Learner Lab) lub

innego narzędzia do wizualizacji danych i udostępniony w aplikacji webowej. Dashboard pozwalałby na filtrowanie danych, np. wybór zawodnika i przegląd jego statystyk.

- Większy zakres oczyszczania danych w procesie ETL. Podczas analizy w EMR Notebook, że więcej danych wymagało oczyszczenia, np. niektóre dane WTA dotyczące pierwszego serwisu były nieprawidłowe (więcej wygranych punktów serwisowych niż zagranych). Podobnie było z danymi dotyczącymi wzrostu zawodniczek WTA.

- Większy zakres zastosowania AWS Glue Data Catalog. Poza przeglądaniem informacji o bazie danych zorganizowanych w tabelkach, zastosowanie AWS Glue Data Catalog w tym projekcie było ograniczone. Zalety AWS Glue Data Catalog takie jak przechowywanie scentralizowanych i zorganizowanych metadanych, zapewnienie spójnego interfejsu między narzędziami, czy optymalizacja zapytań mogłyby zostać wykorzystane lepiej przy większej ilości użytych narzędzi AWS.

- Uprzątniecie kodu w EMR Notebook. Niektóre fragmenty kodu powtarzały się, ale miały różne parametry. Mogłyby zostać zastąpione funkcjami.

- Poza tenisem męskim (ATP), dodatkowe przeanalizowanie podobnych zależności w profesjonalnym tenisie żeńskim (WTA). Jakość dostępnych danych WTA była nieco gorsza niż danych ATP.

- Automatyczne testowanie kodu (*unit testing*).

Bibliografia

[1] S. Briggs, "The Telegrah: Men's tennis to introduce new 'IQ' data analysis to expose players' patterns mid-match," 2023. [Online]. Available: https://www.telegraph.co.uk/tennis/2023/09/25/mens-tennis-hawk-eye-iq-data-analysis-coaches-mid-match-atp/

[2] J. Sackmann, "Git repository 'tennis_atp' - ATP tennis rankings, results, and stats," 2023, all data from this source is attributed to Jeff Sackmann and is licensed under the terms of Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. [Online]. Available: https://github.com/JeffSackmann/tennis_atp.git

[3] ——, "Git repository 'tennis_wta' - WTA tennis rankings, results, and stats," 2023, all data from this source is attributed to Jeff Sackmann and is licensed under the terms of Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. [Online]. Available: https://github.com/JeffSackmann/tennis_wta.git

[4] K. Lin, "Git repository 'atp-world-tour-tennis-data' - ATP World Tour tennis data," 2023. [Online]. Available: https://github.com/serve-and-volley/atp-world-tour-tennis-data

[5] ATP, "Elevating Tennis Performance: ATP & TDI Unveil Tennis IQ Analytics Platform," 2023. [Online]. Available: https://www.atptour.com/en/news/atp-tdi-unveil-tennis-iq-analytics-platform

[6] AWS, "Tutorial: Getting started with Amazon EMR," 2023. [Online]. Available: https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-gs.html

[7] P. Chaudhari, "Install Python libraries on a running cluster with EMR Notebooks," 2022. [Online]. Available: https://aws.amazon.com/blogs/big-data/install-python-libraries-on-a-running-cluster-with-emr-notebooks/

[8] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018. [Online]. Available: https://arxiv.org/pdf/1804.02767v1.pdf

Załącznik A Opis surowych danych

Oryginalny opis poszczególnych kolumn w plikach CSV dostępny jest w jednym z repozytoriów Jeffa Sackmanna [2] pod tym linkiem: LINK

Kolumny w surowych plikach CSV ze statystykami meczów, których dane zostały użyte w tym projekcie, opisane są poniżej:

- `tourney_id`: unikalny identyfikator turnieju tenisowego.
- `surface`: nawierzchnia kortu, na której został rozegrany mecz.
- `winner_id`: unikalny identyfikator gracza, który wygrał mecz.
- `winner_name`: imię i nazwisko gracza, który wygrał mecz.
- `winner_ht`: wzrost gracza, który wygrał mecz.
- `winner_ioc`: narodowość gracza, który wygrał mecz.

- `loser_id`: unikalny identyfikator gracza, który przegrał mecz.
- `loser_name`: imię i nazwisko gracza, który przegrał mecz.
- `loser_ht`: wzrost gracza, który przegrał mecz.
- `loser_ioc`: narodowość gracza, który przegrał mecz.
- `score`: wynik meczu zapisany jako tekst (np. „6-3 1-6 7-5").
- `best_of`: format meczu (maksymalnie 3 sety lub maksymalnie 5 setów).
- `round`: turniejowa runda, w której został rozegrany mecz.
- `w_svpt`: liczba punktów serwisowych gracza, który wygrał mecz.
- `w_1stIn`: liczba trafionych pierwszych serwisów przez gracza, który wygrał mecz.
- `w_1stWon`: liczba punktów wygranych po trafionym pierwszym serwisie przez gracza, który wygrał mecz.
- `l_svpt`: liczba punktów serwisowych gracza, który przegrał mecz.
- `l_1stIn`: liczba trafionych pierwszych serwisów przez gracza, który przegrał mecz.
- `l_1stWon`: liczba punktów wygranych po trafionym pierwszym serwisie przez gracza, który przegrał mecz.