TODO titulo

Trabalho elaborado no âmbito da Unidade Curricular de Armazenamento para Big Data do 2º ano da Licenciatura de Ciência de Dados do Instituto Universitário de Lisboa ISCTE

André Plancha; 105289 Another Author; Num

07/12/2022

Introdução

A Associação de Tenistas Profissionais (*ATP*) é um órgão de ténis profisional masculino, organizando torneios do desporto globalmente. A organização contém na sua base de dados um conjunto de jogos e jogadores que participaram em torneios pelo menos desde 1914, e incluem todos os grandes torneios do circuito masculino, incluindo os torneios de Grand Slam. O objetivo deste trabalho será limpar e preparar os dados de um modelo não-relacional para um modelo-relacional, para que possa ser utilizado em análises posteriores.

Importação dos dados

Para o nosso projeto voi-nos provisionado o ficheiro *atpplayers.json*, que contém os jogos feito pelos jogadores. Para importar este ficheiro, foi utilizado o comando *mongoimport*:

```
mongoimport `
    --db atp `
    --collection games `
    --file  "$pwd\data\atpplayers.json"
```

Isto significa que foi criada uma base de dados chamada *atp*, e uma coleção chamada *games*, que contém os dados do ficheiro *atpplayers.json*.

```
use atp;
db.games.find({}, {_id:0}).limit(5);
```

Born	Belgrade, Serbia	Belgrade, Serbia	Belgrade, Serbia	Belgrade, Serbia	Belgrade, Serbia
Date	2022.02.21 - 2022.02.26	2021.08.30 - 2021.09.12	2021.11.15 - 2021.11.21	2021.08.30 - 2021.09.12	2021.11.01 - 2021.11.07
GameRank	26	145	5	121	
GameRound	Round of 16	Round of 128	Round Robin	Round of 64	Round of 64
Ground	Hard	Hard	Hard	Hard	Hard

Hand	Right-Handed, Two-Handed Backhand	Right-Handed, Two-Handed Backhand	Right-Handed, Two-Handed Backhand	Right-Handed, Two-Handed Backhand	Right-Handed, Two-Handed Backhand
Height	188	188	188	188	188
Location	Dubai, U.A.E.	New York, NY, U.S.A.	Turin, Italy	New York, NY, U.S.A.	Paris, France
Oponent	Karen Khachanov	Holger Rune	Andrey Rublev	Tallon Griekspoor	bye
PlayerName	Novak Djokovic	Novak Djokovic	Novak Djokovic	Novak Djokovic	Novak Djokovic
Prize	$2,794,840	$27,200,000	$7,250,000	$27,200,000	5,207,405
Score	63 76	61 67, 62 61	63 62	62 63 62	null
Tournament	Dubai	US Open	Nitto ATP Finals	US Open	ATP Masters 1000 Paris
WL	W	W	W	W	

nota: Foi retirado a coluna LinkPlayer para melhor visualização

A coleção contém 15 colunas: * PlayerName: Nome do jogador do jogo * Born: Onde este jogador nasceu (cidade, pais) * Height: Altura deste jogador (cm) * Hand: A mão dominante do jogador, e o tipo de *backhand* que utiliza * LinkPlayer: link para a página do jogador em atptour.com * Tournament: nome do torneio do jogo * Location : A cidade e país onde o torneio foi realizado * Date: Periodo de tempo do torneio * GameRound: fase do jogo no torneio * GameRank: *ATP Rankings* do jogo * WL: Vitoria ou Derrota (W ou L) * Opponent: Nome do Oponente * Score: Sets do jogo

Preparar dos dados

Para preparar os dados, nós planeámos transformar a nossa coleção em coleções diferentes, de forma a representar o modelo relacionar, para facilitar a sua transição. Para isso, desenhámos o nosso diagrama do modelo relacional pretendido:

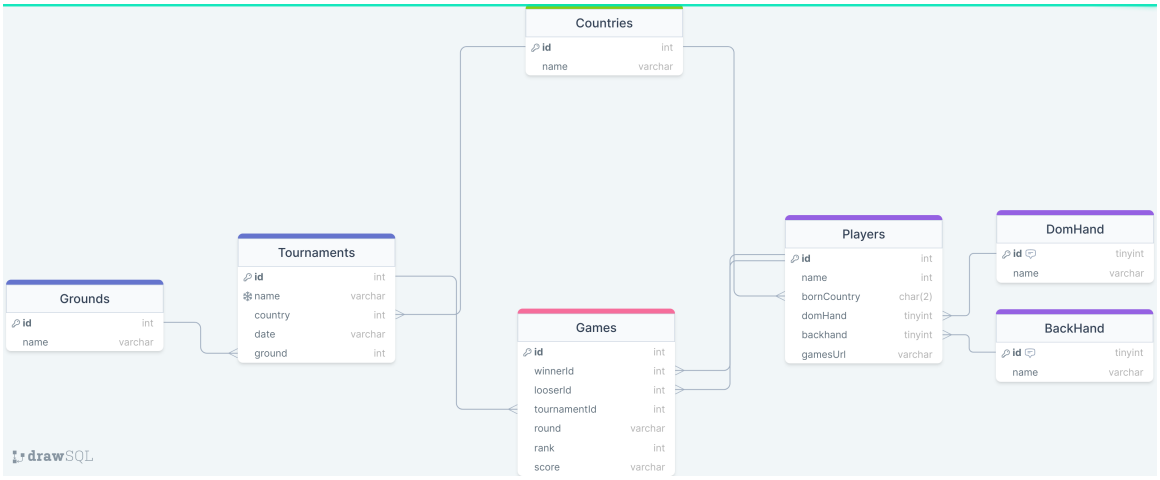

Figura 1: Diagrama do modelo relacional

Antes de começar a transformar os dados, foi necessário verificar a integridade deles.

Primeiro, verificámos se a coluna *Born* e *Location* mantinha o formato "cidade, pais" para todos os jogadores. Mais precisamente, como para os nóssos propósitos, apenas precisamos do país, verificámos se os países estavam sempre no final da string.

```
db.games.aggregate([
  {
    $match: {
      Born: {$not:/,/}
    }
  },
  {
    $group:{
      _id:"$Born"
    }
  },{
    $sample:{size:5}
  }
]);
```

_id
Jeju
Subiaco
Alatri
Verona
Cordoba

Perante os resultados, podemos verificar que existem jogadores cujo país não está no final da string. Para resolver este problema, foi-se adicionado manualmente os países destas cidades, de forma a

poder analisar o pais de origem dos jogadores. O mesmo é observável para a coluna *Location*. Adicionalmente, os países não encontravam consistência; por exemplo, "U.S.A." e "USA" eram usados para representar os Estados Unidos. Logo, foi necessário unificar os países, de forma a que todos os países fossem representados da mesma forma. Para isso, foi criado um ficheiro *countryAlias.csv*, o que associava o código do país com o nome do pais na base de dados. O ficheiro estará disponível no repositório do projeto, e em anexo na submissão.

```csv
alias,country,code
...
San Salvador,Bahamas,BS
Santa Cruz de la Sie,Bolivia,BO
Santiago,Chile,CL
Santo Domingo,Dominican Republic,DO
Sardinia,Italy,IT
SCG,Serbia,RS
Serbia &amp;amp; Montenegro,Serbia,RS
SErgia &amp;amp; M,Serbia,RS
Sharm El Sheikh,Egypt,EG
Slovak,Slovakia,SK
...
```

A partir deste ficheiro, cada pais pode ser associado o seu código usando a pipeline "$lookup" quando for feita a exportação destes dados, depois de importar o ficheiro para a base de dados.

```
mongoimport --db atp --collection countryAlias --type csv --headerline --file .\data\countryAlias.
```

De forma a gfarantir que todos os países sejam representados no resultado final, ao exportar as coleções, vai ser também importada para o modelo relacional países não representados na base de dados.

Outra verivicação que fizemos foi verificar os vários grupos: as colunas *Ground*, *Hand*, e *WL*.

```
db.games.distinct("Ground");
```

result
Carpet
Clay
Grass
Hard

```
db.games.distinct("Hand");
```

result
Ambidextrous, Two-Handed Backhand
Left-Handed, One-Handed Backhand
Left-Handed, Two-Handed Backhand

result
Left-Handed, Unknown Backhand
Right-Handed, One-Handed Backhand
Right-Handed, Two-Handed Backhand
Right-Handed, Unknown Backhand
null

```
db.games.distinct("WL");
```

result
L
W

Como podemos observar, a coluna *Ground* e *WL* têm valores nulos, e a coluna *Hand* tem valores "null". Verificou-se que nestas na primeira e última colunas, a base de dados não contem informação suficiente para verificar poder completar estas colunas, mas não devem interferir com os resultados devido ao pequeno número de casos. Na coluna *WL*, os valores são nulos sem oponentes, ou seja, jogos não jogados. Foi assim decidido não tratar estes casos e filtrar estes ao exportar a coleção.

Ainda assim, decimos separar a *Hand* em duas colunas, porque vai ser útil para a análise futura.

Criação das coleções

Para a nossa criação das coleções, vai ser usada primariamente a função *aggregate* do MongoDB. Esta função permite executar várias *pipelines* de operações sobre os dados, e é muito útil para a criação de coleções, devido à sua flexibilidade e ao seu desempenho. Em cada uma destas, a *pipeline* "$out" é usada para exportar os dados da *query* para uma nova coleção.

Criação da coleção *Player*

```
db.games.aggregate([
  {
    $group: {
      _id:{
        hand: {$split: ["$Hand", ", "]},
        born: {$split: ["$Born", ", "]},
        height: "$Height",
        linkPlayer: "$LinkPlayer",
        playerName: "$PlayerName"
      }
    }
  },{
    $project:{
      playerName: "$_id.playerName",
      country: {$arrayElemAt: ["$_id.born", -1]},
```

```
      height: "$_id.height",
      linkPlayer: "$_id.linkPlayer",
      domHand: {$arrayElemAt: ["$_id.hand", 0]},
      backhand: {$arrayElemAt: ["$_id.hand", 1]}
    }
  },{
    $out: "players"
  }
]);
```

Para a criação da coleção *Player*, foi usada a pipeline "*group*"*paraparaagruparosdadosdecadajogador*, *eapipeline*"projec para tornar os dados mais legíveis e exportáveis. Para o pais do jogador, como o país está sempre no fim da string, ou após uma virgula ou sozinha, foi usada a pipeline "*split*"*e*"arrayElemAt" para selecionar o último elemento do array. Para a mão dominante e o tipo de backhand, foi usada a mesma técnica, separando a mão dominante da backhand.

```
db.players.find({}, {_id:0, linkPlayer: 0}).limit(5)
```

backhand	country	domHand	height	playerName
Two-Handed Backhand	South Africa	Left-Handed	188	Cameron Norrie
Unknown Backhand	Germany	Left-Handed	196	Andreas Lesch
Unknown Backhand		Right-Handed	NA	Ran Xu
null		null	NA	Olivier Le Jeune
null		null	NA	Luka Todorovic

Nota-se que alguns jogadores não têm país, mão dominante ou backhand, e que alguns jogadores não têm altura registada. Estes casos estão consistentes com a coleção original.

Esta coleção não inclui os oponentes, porque estes vão ser tratados ao exportar os dados, com base na coleção *matches*.

Criação da coleção *Tournament*

```
db.games.aggregate([
  {
    $group: {
      _id: {
        tournament: "$Tournament",
        location: {$split: ["$Location", ", "]},
        date: "$Date",
        ground: "$Ground",
        prize: "$Prize"
      }
    }
  },{
```

```
    $project: {
      tournament: "$_id.tournament",
      country: {$arrayElemAt: ["$_id.location", -1]},
      date: "$_id.date",
      ground: "$_id.ground",
      prize: "$_id.prize"
    }
  },{
    $out: "tournaments"
  }
])
```

A coleção *tournaments* é criada de forma semelhante ao *players*, com um semelhante resultado.

```
db.tournaments.find({}, {_id:0}).limit(5)
```

country	date	ground	prize	tournament
Scotland	2000.05.15 - 2000.05.21	Clay	$25,000	Edinburgh
Croatia	2013.08.19 - 2013.08.25	Clay	$10,000	Croatia F8
England	1999.09.13 - 1999.09.19	Clay	$375,000	Bournemouth
Belgium	1997.07.21 - 1997.07.27	Clay	$75,000	Ostend
Japan	2017.09.11 - 2017.09.17	Hard		JPN vs. BRA WG Play-Off

Criação da coleção *matches*

```
db.games.aggregate([
  {
    $match: {
      Oponent: {$ne: "bye"}
    }
  },
  {
    $set: {
      winner: {$cond: [{$eq: ["$WL", "W"]}, "$PlayerName", "$Oponent"]},
      loser: {$cond: [{$eq: ["$WL", "W"]}, "$Oponent", "$PlayerName"]}
    }
  },
  {
    $group: {
      _id:["$Tournament", "$GameRound", "$Date", "$winner", "$loser", ],
      count: {$sum: 1},
      sets: {$push:"$Score"}
    }
  },{
    $project: {
      _id: false,
      tournament: {$arrayElemAt: ["$_id", 0]},
      gameRound: {$arrayElemAt: ["$_id", 1]},
```

```
      date: {$arrayElemAt: ["$_id", 2]},
      winner: {$arrayElemAt: ["$_id", 3]},
      loser: {$arrayElemAt: ["$_id", 4]},
      count: true,
      sets: true
    }
  }, {
    $out: "matches"
  }
])
```

Este processo continua com a semelhança, adicionando a pipeline "*match*"*para filtrar os jogos se mo ponentes*, *eapipeline*" para criar as colunas *winner* e *loser*. É adicionado a pipeline "*match*"*usada para filtrar os jogos no jogados*, *eapipeline*"co o que nos deixa escrever uma condição que decide sobre o vencedor e o perdedor. A pipeline "$sets" garante que os nossos jogos repitidos são realmente repetidos, sendo que um score de, por exemplo, "67 72 25" seria igual a um score de "76 27 52", devido à natureza da coluna; esta vai incluir as varias formas como a coluna se encontra nos vários jogos repetidos.

```
db.matches.find({}, {_id:0}).limit(5)
```

count	2	2	2	2	2
date	2007.06.18 - 2007.06.24	2007.06.18 - 2007.06.24	2007.06.18 - 2007.06.24	2007.06.18 - 2007.06.24	2007.06.18 - 2007.06.24
gameRound	1st Round Qualifying	1st Round Qualifying	1st Round Qualifying	1st Round Qualifying	1st Round Qualifying
loser	Sven Swinnen	Xander Spong	Martijn Van Haasteren	Patrick Eichenberger	Dustin Brown
sets	["36 46", "63 64"]	["57 06", "75 60"]	["57 63 26", "75 36 62"]	["63 67, 63", "36 76, 36"]	["67, 26", "76, 62"]
tournament	's-Hertogenbosch	's-Hertogenbosch	's-Hertogenbosch	's-Hertogenbosch	's-Hertogenbosch
winner	Alexander Nonnekes	Bart Beks	Gilles Elseneer	Jordan Kerr	Marcin Matkowski

Criação da coleção *countryAliases*

Para a criação desta coleção, vai ser importada o csv previamente referido.

```
mongoimport `
    --db atp `
    --collection countryAliases `
    --type csv `
    --headerline `
    --file "$pwd\data\countryAlias.csv"
```

```
db.countryAliases.find({}, {_id:0}).limit(8)
```

alias	code	country
Mexico	MX	Acapulco
CÃ´te d'Ivoire	CI	Abidjan
France	FR	Ajaccio
United Kingdom	GB	AL
Italy	IT	Alatri
Kazachstan	KZ	Aktau
Canada	CA	ALberta
USA	US	Alexandria