Adecuación de datos astronómicos para importación a Neo4J

Ing. Martín G. Casatti
e-mail: mcasatti@frc.utn.edu.ar

21 de octubre de 2022

Descripción de las herramientas y métodos utilizados para el procesamiento de los datos astronómicos para su incorporación en la base de datos ONGDB.

1. Introducción

Como primer paso para el proyecto de detección de patrones en datos astronómicos por medio de bases de datos de grafos, se deben importar dichos datos al almacenamiento elegido.
Los datos vienen en la forma de archivos separados por coma que cuentan con los siguientes datos:

- Ascensión recta.
- Declinación
- Lectura del Filtro F555W
- Error en la Lectura del Filtro F555W
- Lectura del Filtro F814W
- Error en la Lectura del Filtro F814W
- Diferencia entre F555 y F814 (F555-F814)
- Error en la diferencia entre F555 y F814
- Identificador del registro

Los datos son de utilización directa excepto el identificador de registro. Esto es así debido a que el identificador se genera por archivo, por lo que en diferentes archivos se repiten los mismos indicadores (entero, autoincremental)
Como además las lecturas no cuentan con un identificador único absoluto en todo el dominio de datos, es de suma importancia generar un identificador que no sea dependiente del archivo de datos sino de la lectura del instrumento.

2. Métodos

Inicialmente se descargaron y descomprimieron los archivos disponibles en el repositorio DropBox[1] los cuales contienen un conjunto de archivos con lecturas de Hubble, para los filtros mencionados anteriormente y para las galaxias NGC300 y NGC2366 (ambos cuentan con los mismos campos de datos).
Una vez descargados y descomprimidos los archivos se obtuvieron varios archivos con formato CSV, los que se procesaron utilizando la herramienta TOPCAT[2].
TOPCAT tiene la posibilidad de generar columnas "sintéticas" o sea elaboradas a partir de los datos existentes en el archivo original. Esta funcionalidad resultó muy útil a la hora de generar información necesaria para la importación a la base de datos.
Estas columnas se pueden generar utilizando una gran cantidad de funciones disponibles, para procesar los datos existentes y generar nueva información, la que se puede grabar posteriormente como una tabla adecuada para su importación.
El principal dato que se debe generar para poder realizar la importación es el identificador único que permitirá individualizar cada una de las lecturas de los instrumentos.
Actualmente cada lectura se identifica mediante los datos de Ascensión Recta y Declinación, pero dichos valores tienen el inconveniente de ser de tipo flotante con múltiples decimales de precisión. Las claves primarias de este tipo presentan siempre un problema de rendimiento y además no son indexables, o lo son solamente en algunos motores de base de datos.
Utilizando las herramientas de generación de TOPCAT se generaron dos columnas intermedias, para adecuar los tipos y formatos de datos, utilizando las siguientes funciones:

```
1  RA_id = formatDecimal(RA*100000,"#########");
2  DEC_id = formatDecimal(DEC*100000,"#########");
```

Listado 1: Funciones para generación de columnas intermedias

Estas dos funciones generan dos columnas temporales con el siguiente formato:

- Dato original: `RA=112.11976209349766`, Dato generado: `RA_id=11211976`
- Dato original: `DEC=69.18853071324`, Dato generado: `DEC_id=6918853`

Una vez generadas estas dos columnas se concatenan con un separador, para generar el identificador final que se va a utilizar para identificar las lecturas individuales.

```
1  id = RA_id+":"+DEC_id;
```

Listado 2: Generación de identificador final

lo que genera un identificador final con el siguiente formato:

- `id=11211976:6918853`

2.1. Columnas adicionales

Por medio del mismo mecanismo mencionado anteriormente se generaron algunas columnas adicionales destinadas a facilitar el procesamiento de los datos una vez que los mismos se encuentren importados a la base de datos.
Las mismas son:

```
galaxy = trim(concat('NGC300',''));
field = 1
```

Listado 3: Identificación de galaxia y campo de observacion

Estos datos salen directamente de los archivos de datos a procesar (el formato del nombre del archivo de datos original es campo1_NGC300.csv)
Se incluyeron además dos columnas que permitiran agrupar las zonas de las observaciones a partir de la elevación recta y la declinación:

```
RA_group = toInteger(RA);
DEC_group = toInteger(DEC);
```

Listado 4: Creación de los grupos de elevación y declinación

2.2. Automatización de la adecuación de datos

Si bien la herramienta TOPCAT, mencionada anteriormente, dispone de una gran variedad de funciones para la adecuación de la estructura de las tablas CVS, las cuales se utilizaron principalmente en la faz exploratoria del presente trabajo, dichas funciones se ejecutan de manera interactiva con el usuario y no son adecuadas para procesos masivos o automatizados.
TOPCAT utiliza la herramienta STILS[3] en background para realizar las operaciones por lo que se planteó la posibilidad de transformar todos los comandos utilizados de manera interactiva en una única secuencia de comandos que se pudiera ejecutar mediante STILTS sobre cualquier archivo de datos para generar un archivos de datos directamente importable.
De ésta manera se generó el siguiente archivo de comandos de Windows (.CMD) el cual recibe como parámetro el nombre de un archivo CSV en formato original y genera un archivo con extensión .import.csv que tiene el formato esperado por la herramienta de importación de la base de datos.

```
@echo off
setlocal
set _filename=%~n1
set _extension=%~x1
set _outextension=.import%_extension%
set _inputfile=%_filename%%_extension%
set _outputfile=%_filename%%_outextension%
set _campo=%_filename:~5,1%
set _galaxy=%_filename:~7,8%

echo Generando ^<%_outputfile%^> a partir de ^<%_inputfile%^>

@java -jar "../tools/stilts/stilts.jar" ^
   -verbose ^
   tpipe ^
   ifmt=csv ^
   cmd="progress" ^
   cmd="delcols id" ^
```

```
19    cmd="addcol -after RA RA_id 'formatDecimal(RA*100000,\"
          #########\")'" ^
20    cmd="addcol -after DEC DEC_id 'formatDecimal(DEC*100000,\"
          #########\")'" ^
21    cmd="addcol -before RA id 'RA_id+\":\"+DEC_id'" ^
22    cmd="addcol -after id galaxy '\"%_galaxy%\"'" ^
23    cmd="addcol -after galaxy field '\"%_campo%\"'" ^
24    cmd="addcol -after field RA_group 'toInteger(RA)'" ^
25    cmd="addcol -after RA_group DEC_group 'toInteger(DEC)'" ^
26    out=%_outputfile% ^
27    ofmt=csv ^
28    in=%_inputfile%
29 endlocal
```

Listado 5: Archivo batch para la conversión de fuentes de datos

3. Resultados

4. Discusión

5. Referencias y bibliografía

Referencias

[1] DropBox para descarga de archivos, `https://www.dropbox.com/sh/kniy70dwlx5s1ow/AACySwi4pJsctAstrZf-mK3Wa?dl=0`

[2] TOPCAT (herramienta de gestión de tablas de texto), `http://www.star.bristol.ac.uk/~mbt/topcat/#starjava`

[3] STILTS (Starlink Tables Infrastructure Library Tool Set) `http://www.star.bris.ac.uk/~mbt/stilts/`

Pendientes

Todo list