

Práctica 5

Vamos a trabajar con la base de datos Neo4J ver: <https://neo4j.com/>

Antes de empezar tienen que tener instalada la versión de escritorio o creado un sandbox en el sitio de neo4j como vimos en los videos de la clase 5.

Como un primer ejercicio se recomienda hacer el tutorial de la base de datos de películas si aún no lo hicieron.

Redes de palabras

Vamos a trabajar con un grafo de asociación de palabras extraído del proyecto [Wordnet](#). Dentro de ese proyecto hay un dataset sobre palabras evocadas que explora el grado de asociación que existe entre palabras comunes (en este caso en inglés). En esa red cada par de palabras tiene un puntaje asociado entre 0 y 100 que resulta de promediar las respuestas de los participantes del estudio. Para evitar un poco el trabajo previo de curar y limpiar los datos preparamos ya dos archivos listos para cargar en la base [nodos.csv](#) y [links.csv](#). Nodos contiene la lista de palabras usadas en el estudio y links contiene los pares de palabras estudiados junto con el puntaje asignado para la asociación. Noten que valen autoloops. También tienen disponible la clase de palabra que se trata: verbo, adjetivo, sustantivo

Creen un sandbox o una base de datos local vacía para empezar a trabajar.

Ejercicio 1 - Cargar los nodos en la base de datos

Para eso vamos a usar la sentencia [LOAD CSV](#). Intenten crear la sentencia adecuada usando la documentación.

NOTA: Si trabajan en el sandbox pueden usar los links de los archivos de entrada directos en la sentencia LOAD CSV. Si trabajan en la versión desktop tiene que hacer click en los 3 puntos al lado del nombre de la base y poner los archivos en la carpeta *import folder*.

AYUDA:

```
LOAD CSV WITH HEADERS FROM "file:///nodos.csv" as line
CREATE (w1:word {id: line.id, name: line.word, type: line.type})
```

Ejercicio 2 - Cargar las relaciones

Fijense que van a tener que combinar sentencias MATCH para recuperar los nodos que van a conectar con una sentencia CREATE para crear la relación. No se olviden de convertir el puntaje de la asociación a formato float al cargarlo.

AYUDA:

```
LOAD CSV WITH HEADERS FROM "file:///word_associations_links.csv" as line
MATCH (w1:word {id: line.word_1_id})
MATCH (w2:word {id: line.word_2_id})
CREATE (w1)-[:EVOCA {strength: toFloat(line.association_rating)}]-(w2)
```

Ejercicio 2 -

- 1) Encuentren la palabra con la mayor cantidad de relaciones.
- 2) Encuentren la palabra con la menor cantidad de relaciones
- 3) Expandan esa palabra a sus vecinos ¿Cuántos son?

Ejercicio 3 -

- 1) ¿Cuál es el nivel de asociación promedio?
- 2) ¿Cuál es el nivel de asociación máximo?
- 3) ¿Cuál es el nivel de asociación mínimo?
- 4) ¿Qué tipo de vínculo observan para las relaciones de puntaje máximo?

Ejercicio 4 - Caminos mínimos

- 1) Elijan dos palabras que no estén vinculadas directamente y calculen su camino mínimo (usen la función [shortestPath](#))
- 2) ¿Cuántos nodos están a distancia 2 de la palabra "film"?
- 3) Elijan dos palabras que no estén vinculadas directamente, ¿cuántos vecinos en común tienen?

OPCIONAL

Instalen la librería de ciencia de datos. <https://neo4j.com/product/graph-data-science-library/>

- 1) ¿Cuál es el diámetro de la red (es decir el camino mínimo promedio)?
- 2) ¿Cuáles son los nodos más centrales?
- 3) Busquen comunidades de palabras muy vinculadas entre sí.