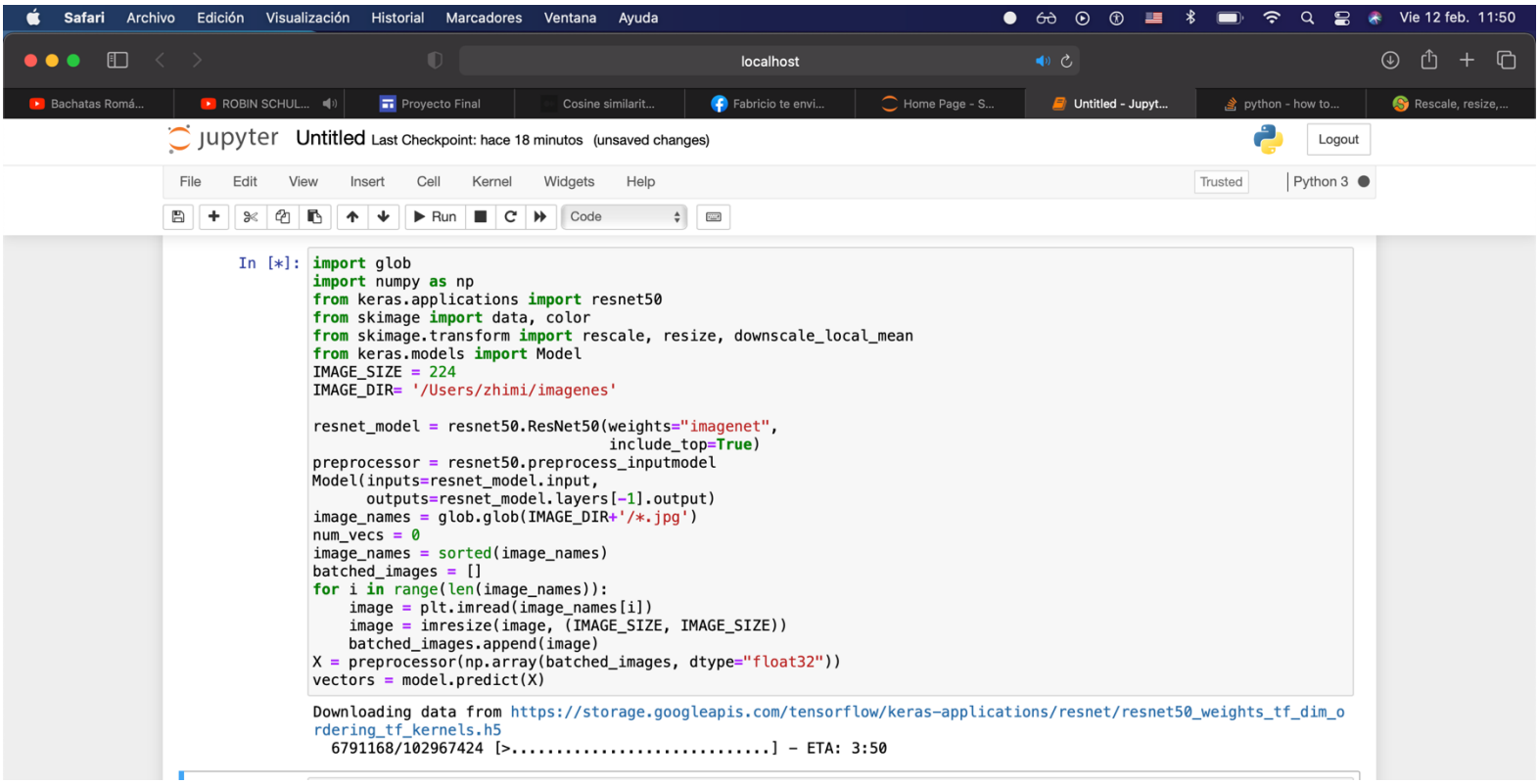
Similitud de coseno en Neo4J

1) Implementacion del Código: Descargar Datos resnet50



2) Mostramos las personas

```
match(p:Person) Return p.person_id, p.person_name, p.person_embedding
order by p.person_id
```

p.person_id	p.person_name	p.embedding
1	"Dave Brubeck"	[1, 1, 1, 1]
2	"Paul Desmond"	[12, 30, 14, 100]
3	null	[1, 1, 1, 2]
4	null	[2, 2, 2, 3]
5	null	[3, 3, 3, 3]
6	null	[1, 1, 1, 10]

3) Similitud

```
MATCH (p1:Person {name: 'Michael'})-[likes1:LIKES]→(cuisine)
MATCH (p2:Person {name: "Arya"})-[likes2:LIKES]→(cuisine)
RETURN p1.name AS from,
       p2.name AS to,
       algo.similarity.cosine(collect(likes1.score),
       collect(likes2.score)) AS similarity
```

4) Incrustaciones como una propiedad de nodo

```
MERGE (french:Cuisine {name:'French'})
  SET french.embedding = [0.71, 0.33, 0.81, 0.52, 0.41]
MERGE (italian:Cuisine {name:'Italian'})
  SET italian.embedding = [0.31, 0.72, 0.58, 0.67, 0.31]
MERGE (indian:Cuisine {name:'Indian'})
  SET indian.embedding = [0.43, 0.26, 0.98, 0.51, 0.76]
MERGE (lebanese:Cuisine {name:'Lebanese'})
  SET lebanese.embedding = [0.12, 0.23, 0.35, 0.31, 0.39]
MERGE (portuguese:Cuisine {name:'Portuguese'})
  SET portuguese.embedding = [0.47, 0.98, 0.81, 0.72, 0.89]
MERGE (british:Cuisine {name:'British'})
  SET british.embedding = [0.94, 0.12, 0.23, 0.4, 0.71]
MERGE (mauritian:Cuisine {name:'Mauritian'})
  SET mauritian.embedding = [0.31, 0.56, 0.98, 0.21, 0.62]
MATCH (c:Cuisine)
  WITH {item:id(c), weights: c.embedding} as userData
  WITH collect(userData) as data
CALL algo.similarity.cosine.stream(data, {skipValue: null})
YIELD item1, item2, count1, count2, similarity
RETURN algo.getNodeById(item1).name AS from,
       algo.getNodeById(item2).name AS to, similarity
ORDER BY similarity DESC
```

5) Filtro para que coincida con un nodo específico

```
MATCH (c: Cuisine{name:"French"})
```

6) La agregación de datos

```
WITH {item:id(c), weights: c.embedding} as userData
WITH collect(userData) as data
```

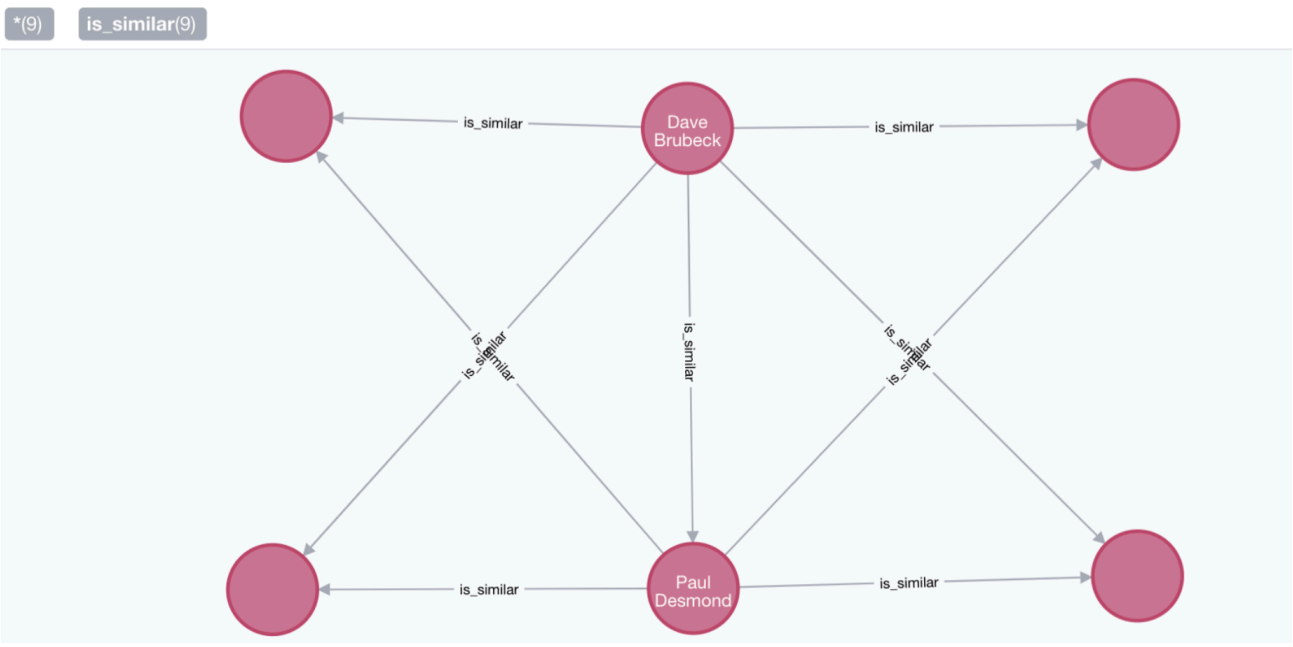
7) consulta deseada que calcula la similitud

```
WITH 1 AS startId
MATCH (p1:Person{person_id:startId}),(p2:Person)
WHERE p2 <> p1
WITH p1, p2,
algo.similarity.cosine(p1.embedding,p2.embedding) as similarity
MERGE (p1)-[r1:is_similar{score: similarity}]-(p2)
RETURN p1,p2,r1
WITH 2 AS startId
MATCH (p1:Person{person_id:startId}),(p2:Person)
WHERE p2 <> p1
WITH p1, p2,
algo.similarity.cosine(p1.embedding,p2.embedding) as similarity
MERGE (p1)-[r1:is_similar{score: similarity}]-(p2)
RETURN p1,p2,r1
```

8) Resultados Obtenido Grafico:

```
MATCH (n:Person) Return n limit 25
```

```
TCH (n:Person) Return n limit 25
```



9) Resultados Obtenido Similitud:

```
MATCH (p1: Person{person_name:"Dave Brubeck"})-[r:is_similar]-(p2:Person)
WHERE r.score > 0.8
RETURN p1.person_id, r.score, p2.person_id
```

p1.person_id	r.score	p2.person_id
1	0.944911182523068	3
1	1.0	5
1	0.9819805060619657	4