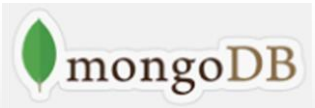


Práctica Base de Datos no Convencionales (Máster Data Science)

Beatriz Visitación, Natalia Alonso y Susana Albarrán Félix

Contenidos

1 Introducción

GESTIÓN DEL DOCUMENTO			
Autor	Beatriz Visitación, Natalia Alonso y Susana Albarrán	**Fecha**	04/05/2020
Nombre fichero	MDS_Memoria_Visitacion_Alonso_Y_Albarran_MongoDB.pdf	**Versión**	1.0
Descripción: En esta memoria, se incluyen todos los pasos que se han seguido para realizar la práctica de Bases de Datos no Convencionales con MongoDB. Para la realización de la práctica se ha usado el Lenguaje de programación Python (3.7) y MongoDB. Las consultas se han realizado en Robo 3T.			

En esta práctica se partirá de una fuente de datos real, exactamente de la base de datos **DBLP Computer Science Bibliography,** considerada como la mayor recopilación existente de referencias bibliográficas académicas centrada en la informática (en cualquiera de sus variantes). En particular, almacena los datos relativos a revistas científicas y congresos académicos sobre informática, en muchos casos remontándose hasta publicaciones de los años 60. Se caracteriza por un elevado grado de interdependencia interna. Además, el tamaño de esta fuente (en la forma de un único fichero) hará difícil su manejo, mostrando (aun a pequeña escala) algunas de las dificultades habituales en el procesamiento de esta información.

Consta de tres partes fundamentales: captura, procesamiento de datos y almacenamiento, y análisis de los datos.

El objetivo de este trabajo es utilizar esta información para realizar consultas en MongoDB y así poder obtener una serie de medidas que permitirán evaluar a los investigadores en informática.

2 Captura, procesamiento de los datos y carga de los datos en MongoDB.

La captura y procesamiento de datos consiste en descargar el fichero dbpl.xml https://dblp.uni-trier.de/xml/ y a continuación procesarlo para convertirlo a formato JSON.

En nuestro caso se ha procesado el fichero dbpl.xml desde Python. Estos son los pasos que se han seguido:

- Convertir el fichero dbpl.xml a formato .json (Código fuente: **1.Convertir xml_json.py**). Mediante este código se genera el fichero **'jsondata.json'.**

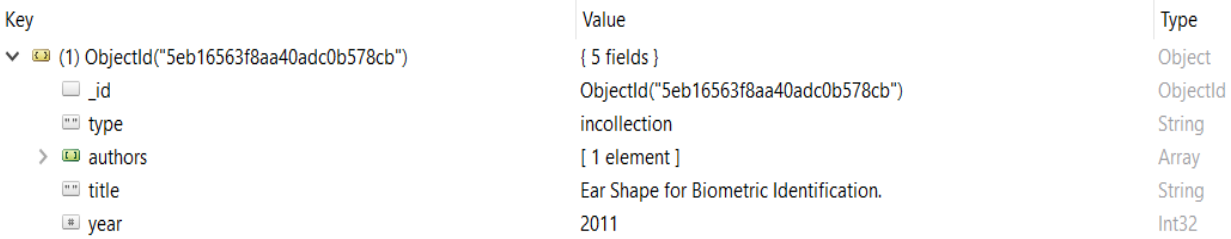
- Se realiza un filtrado sobre el fichero **'jsondata.json'** (Código fuente: **2.read_json.py**) para sólo trabajar con los tres tipos de publicaciones: artículos de revista (article), artículos de congresos (inproceedings) y artículos de libros (incollecction). Mediante este código se generan tres ficheros: **incollections.json, article.json, inproceedings.json.**

Una vez convertidos al formato necesario los datos se almacenan en una base de datos MongoDB **(db.collection_publication)**. Para realizar este paso se usa el código fuente **3.crear_db_inserMany.py**.

Este último script para cargar los datos en MongoDB, lee los ficheros incollections.json, article.json e inproceedings.json, y para cada uno de ellos se recorre toda la lista de publicaciones extrayendo los campos **autor, title y year**. En algunos casos el campo autor es un array, ya que una publicación puede tener varios autores. Adicionalmente, se crea un nuevo campo **type** que identificará a los 3 tipos de publicaciones con los que se va a trabajar. También se ha observado que algunos de estos campos eran compuestos y tenían un subcampo "#**text**", en estos casos se ha seleccionado sólo el contenido de dicho campo. Estas son las instrucciones que se usan para la creación de la base de datos y la inserción de los distintos documentos:

```
conection = pymongo.MongoClient()
db = conection.dblp
collection = db.collection_publication
collection.insert_many(lista_incollections)
collection.insert_many(lista_inproceedings)
collection.insert_many(lista_articles)
conection.close()
```

La estructura de cada documento de la colección creada es la siguiente:



3 Análisis de los datos.

En esta etapa se realizan consultas sobre la base de datos almacenada en MongoDB **(db.collection_publication),** usando el lenguaje de consulta propio de esta Base de Datos.

Antes de realizar las consultas se ha creado un índice para reducir el tiempo de ejecución de algunas de ellas. Se ha decidido indexar el campo **autor**, ya que es uno de los más usados *(db.collection_publication.createIndex({authors:1})).* Adicionalmente, se probaron otros índices, pero se concluyó que el rendimiento de las consultas no mejoraba.

3.1 Consulta 1.

Listado de todas las publicaciones de un autor determinado.
En este caso se ha hecho un match por el autor: Robert Demolombe

```
db.collection_publication.aggregate([
{$match:{authors:"Robert Demolombe"}},
{$project:{_id:"$title"}}
])
```

```
db.collection_publication.aggregate([
    {$match:{authors:"Robert Demolombe"}},
    {$project:{_id:"$title"}}
    ])
```

collection_publication 13.7 sec.

Key	Value
(1) Temporal Logic Modeling of Biological Systems.	{ 1 field }
(2) Uncertainty in Intelligent Databases.	{ 1 field }
(3) How to Produce Information About a Given Entity Using Aut...	{ 1 field }
(4) The Language of SYNTEX-2, an Implemented Relational-lik...	{ 1 field }
(5) Answers About Validity and Completeness of Data: Formal ...	{ 1 field }
(6) Abstract Objects to Represent Large Answers to Queries in a...	{ 1 field }
(7) Answering Queries About Validity and Completeness of Dat...	{ 1 field }
(8) Normes, Pouvoirs et Rles: vers une formalisation en logique.	{ 1 field }
(9) Towards a Uniform Logical Representation of Different Kind...	{ 1 field }
(10) Trust and Agency in the Context of Communication.	{ 1 field }
(11) Reasoning About the Safety of Information: From Logical F...	{ 1 field }
(12) A Modal Logical Framework for Security Policies.	{ 1 field }

3.2 Consulta 2.

Número de publicaciones de un autor determinado.
Se usa el mismo autor que en la consulta 1: Robert Demolombe

```
db.collection_publication.aggregate([
   {$match:{authors:"Robert Demolombe"}},
   {$count:"Numero_publicaciones"},
   {$project:{_id:"$Numero_publicaciones"}}
   ])
```

```
db.collection_publication.aggregate([
    {$match:{authors:"Robert Demolombe"}},
    {$count:"Numero_publicaciones"},
    {$project:{_id:"$Numero_publicaciones"}}
    ])
```

3.3 Consulta 3.

Número de artículos en revista para el año 2018.

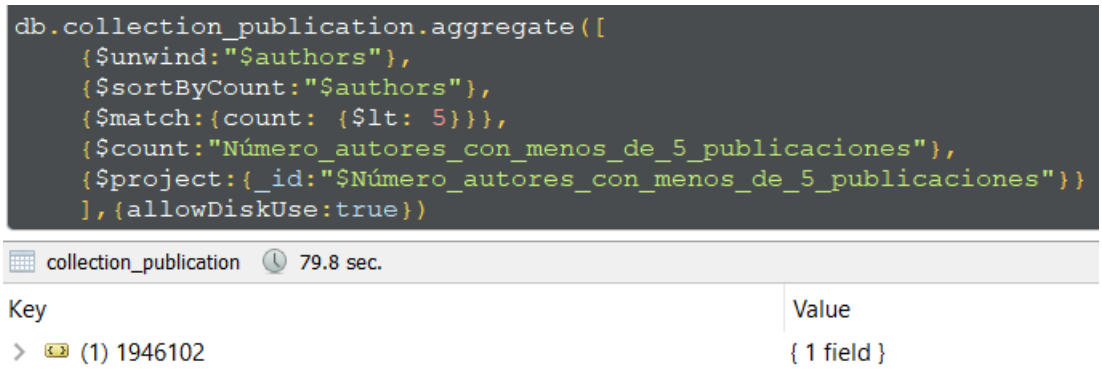
```
db.collection_publication.aggregate([
   {$match:{type:"article"}},
   {$match:{year:2018}},
   {$count:"Numero_articles_2018"},
   {$project:{_id:"$Numero_articles_2018"}}
   ])
```

```
db.collection_publication.aggregate([
    {$match:{type:"article"}},
    {$match:{year:2018}},
    {$count:"Numero_articles_2018"},
    {$project:{_id:"$Numero_articles_2018"}}
    ])
```

3.4 Consulta 4.

Número de autores ocasionales, es decir, que tengan menos de 5 publicaciones en total.

```
db.collection_publication.aggregate([
   {$unwind:"$authors"},
   {$sortByCount:"$authors"},
   {$match:{count: {$lt: 5}}},
   {$count:"Número_autores_con_menos_de_5_publicaciones"}
   {$project:{_id:"$Número_autores_con_menos_de_5_publicaciones"}}
   ],{allowDiskUse:true})
```



3.5 Consulta 5.

Número de artículos de revista (article) y número de artículos en congresos (inproceedings) de los diez autores con más publicaciones totales.

- Se crea una colección nueva **"Top10_autores":**

  ```
  db.collection_publication.aggregate([
  {$unwind:"$authors"},
  {$sortByCount:"$authors"},
  {$limit:10},
  {$out:"Top10_autores"}
  ],{allowDiskUse:true})
  ```

- Se crea otra colección **(“Autores_articles”)** donde se obtiene el número de autores para artículos de revista:

```
db.collection_publication.aggregate([
 {$match:{type:"article"}},
 {$unwind:"$authors"},
{$sortByCount:"$authors"},
{$out:"Autores_articles"}
 ],{allowDiskUse:true})
```

- Se unen las dos tablas anteriores:

```
db.Top10_autores.aggregate([
{$lookup:{
from : "Autores_articles",
 localField : "_id",
 foreignField : "_id",
 as : "Número_artículos"}},
 {$unwind:"$Número_artículos"}
  ],{allowDiskUse:true})
```

```
db.Top10_autores.aggregate([
    {$lookup:{
        from : "Autores_articles",
        localField : "_id",
        foreignField : "_id",
        as : "Número_artículos"}},
    {$unwind:"$Número_artículos"}
    ],{allowDiskUse:true})
```

Top10_autores 0.006 sec.

Key	Value
(1) H. Vincent Poor	{ 3 fields }
(2) Mohamed-Slim Alouini	{ 3 fields }
(3) Philip S. Yu	{ 3 fields }
(4) Wei Zhang	{ 3 fields }
(5) Wei Wang	{ 3 fields }
(6) Lajos Hanzo	{ 3 fields }
(7) Wei Li	{ 3 fields }
(8) Yu Zhang	{ 3 fields }
(9) Wen Gao 0001	{ 3 fields }
(10) Victor C. M. Leung	{ 3 fields }

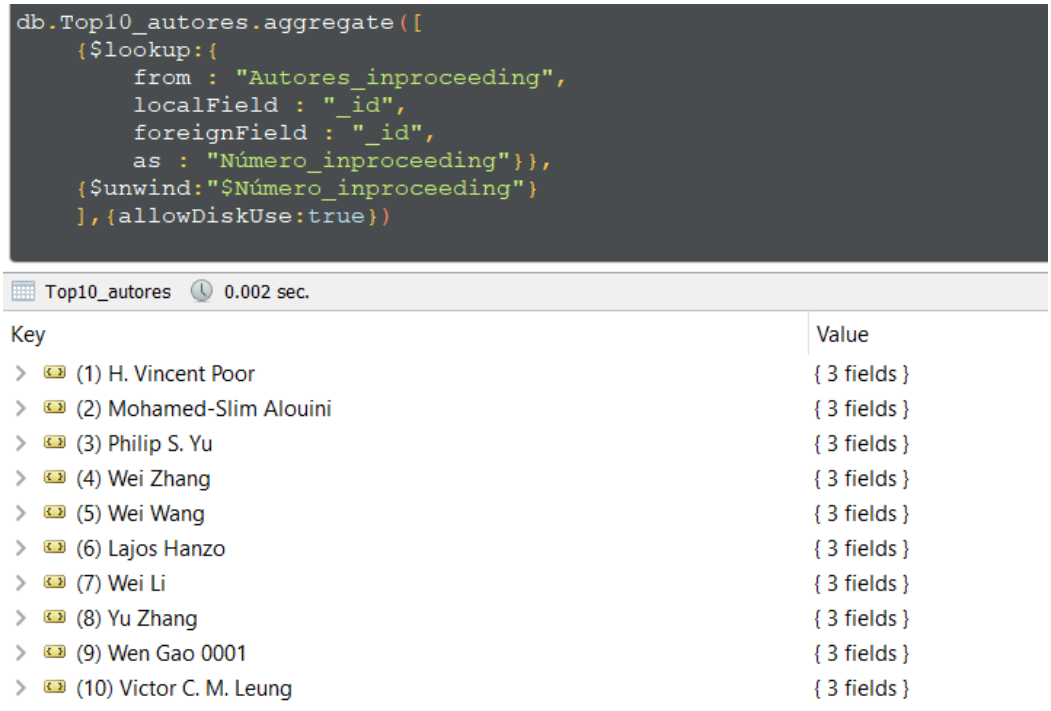
/* y número de artículos en congresos (inproceedings) de los diez autores con más publicaciones totales. */

- Se crea otra colección **(“Autores_inproceeding”)** donde se obtiene el número de autores para artículos de congresos:

```
db.collection_publication.aggregate([
   {$match:{type:"inproceeding"}},
   {$unwind:"$authors"},
   {$sortByCount:"$authors"},
   {$out:"Autores_inproceeding"}
   ],{allowDiskUse:true})
```

- Se unen las dos tablas anteriores:

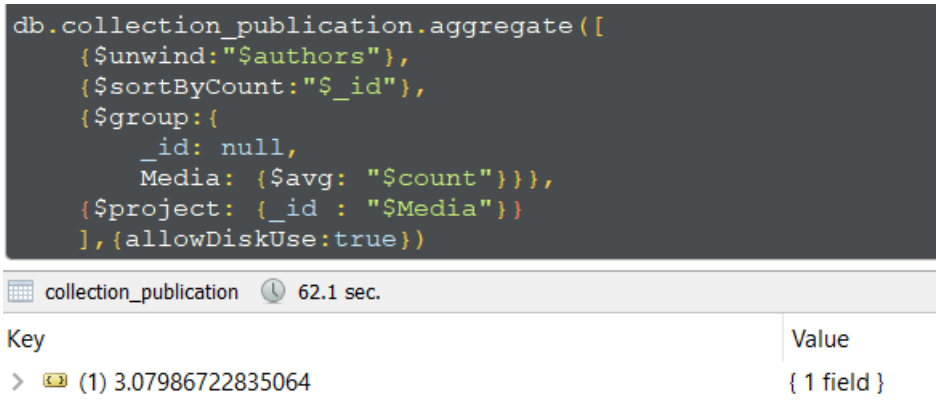
```
db.Top10_autores.aggregate([
   {$lookup:{
      from : "Autores_inproceeding",
      localField : "_id",
      foreignField : "_id",
      as : "Número_inproceeding"}},
   {$unwind:"$Número_inproceeding"}
   ],{allowDiskUse:true})
```



3.6 Consulta 6.

Número medio de autores de todas las publicaciones que tenga en su conjunto de datos.

```
db.collection_publication.aggregate([
  {$unwind:"$authors"},
  {$sortByCount:"$_id"},
  {$group:{
     _id: null,
     Media: {$avg: "$count"}}},
  {$project: {_id : "$Media"}}
  ],{allowDiskUse:true})
```

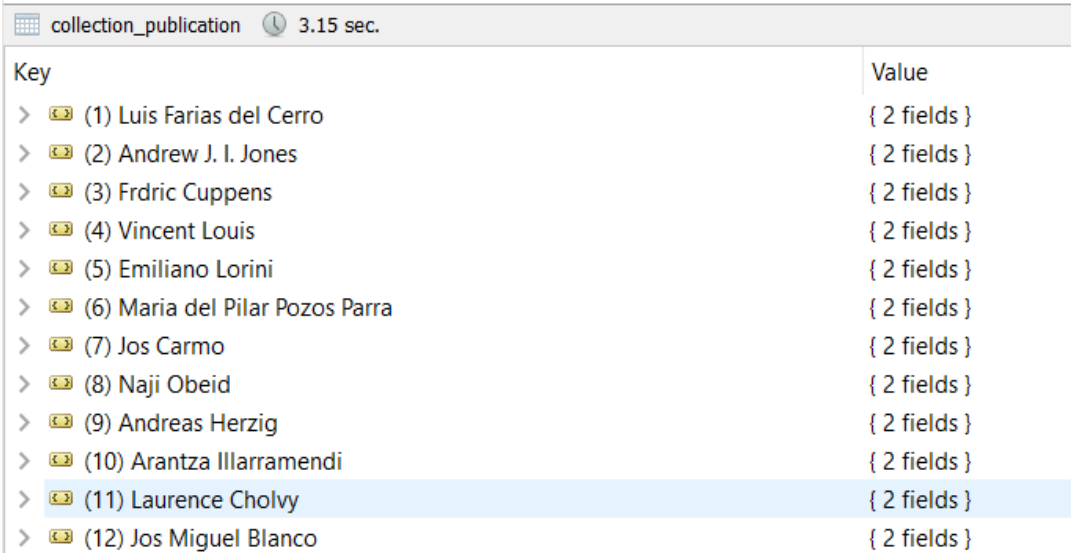


3.7 Consulta 7.

Listado de coautores de un autor (Se denomina coautor a cualquier persona que haya firmado una publicación).

```
db.collection_publication.aggregate([
  {$match:{authors:"Robert Demolombe"}},
  {$project: {"Co-autores" : "$authors"}},
  {$unwind:"$Co-autores"},
  {$sortByCount:"$Co-autores"},
  {$match:{_id : {$ne : "Robert Demolombe"}}}
  ])
```

```
db.collection_publication.aggregate([
    {$match:{authors:"Robert Demolombe"}},
    {$project: {"Co-autores" : "$authors"}},
    {$unwind:"$Co-autores"},
    {$sortByCount:"$Co-autores"},
    {$match:{_id : {$ne : "Robert Demolombe"}}},
    ])
```

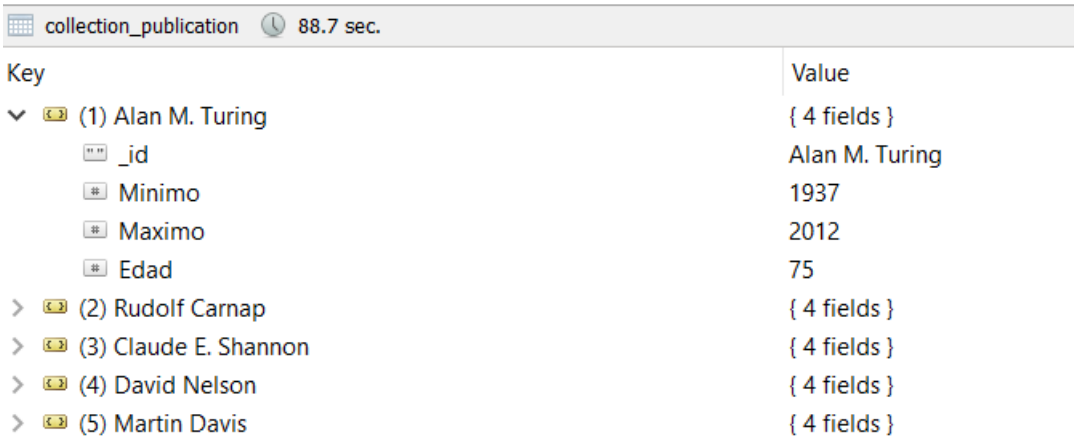


3.8 Consulta 8.

Edad de los 5 autores con un periodo de publicaciones más largo (Se considera la edad de un autor el número de años transcurridos desde la fecha de su primera publicación hasta la última registrada).

```
db.collection_publication.aggregate([
   {$unwind:"$authors"},
   {$group: {
      _id: "$authors",
      Minimo: { $min: "$year" },
      Maximo: { $max: "$year" }}},
    {$addFields:{Edad: { $subtract: ["$Maximo", "$Minimo"] }}},
    {$sort: {"Edad" : -1}},
    {$limit: 5}
    ],{allowDiskUse:true})
```

```
db.collection_publication.aggregate([
    {$unwind:"$authors"},
    {$group: {
        _id: "$authors",
        Minimo: { $min: "$year" },
        Maximo: { $max: "$year" }}},
     {$addFields:{Edad: { $subtract: ["$Maximo", "$Minimo"] }}},
     {$sort: {"Edad" : -1}},
     {$limit: 5}
     ],{allowDiskUse:true})
```



3.9 Consulta 9.

Número de autores novatos, es decir, que tengan una Edad menor de 5 años. (Se considera la edad de un autor el número de años transcurridos desde la fecha de su primera publicación hasta la última registrada).

```
db.collection_publication.aggregate([
   {$unwind:"$authors"},
   {$group: {
      _id: "$authors",
      Minimo: { $min: "$year" },
      Maximo: { $max: "$year" }}},
    {$addFields:{ Edad: { $subtract: ["$Maximo", "$Minimo"] }}},
    {$match:{Edad: {$lt: 5}}},
    {$count : "Número_autores_novatos"},
    {$project : {_id: "$Número_autores_novatos"}}
    ],{allowDiskUse:true})
```

```
db.collection_publication.aggregate([
    {$unwind:"$authors"},
    {$group: {
        _id: "$authors",
        Minimo: { $min: "$year" },
        Maximo: { $max: "$year" }}},
     {$addFields:{ Edad: { $subtract: ["$Maximo", "$Minim
     {$match:{Edad: {$lt: 5}}},
     {$count : "Número_autores_novatos"},
     {$project : {_id: "$Número_autores_novatos"}}
     ],{allowDiskUse:true})
```

collection_publication 90.1 sec.

Key	Value
(1) 1858893	{ 1 field }

3.10 Consulta 10.

Porcentaje de publicaciones en revistas con respecto al total de publicaciones.

```
db.collection_publication.aggregate([
    {$facet:
        {
            "Numero_articulos": [{$match:{type:"article"}}, {$count:"Num_articles"}],
            "Numero_total": [ {$count : "Num_total"}]
        }
    },
    {$unwind:"$Numero_articulos"},
    {$unwind:"$Numero_total"},
    {$addFields:{ Porcentaje: { $divide: ["$Numero_articulos.Num_articles",
"$Numero_total.Num_total"] }}}
    ],{allowDiskUse:true})
```

```
db.collection_publication.aggregate([
    {$facet:
        {
            "Numero_articulos": [{$match:{type:"article"}}, {$count:"Num_articles"}],
            "Numero_total": [ {$count : "Num_total"}]
        }
    },
    {$unwind:"$Numero_articulos"},
    {$unwind:"$Numero_total"},
    {$addFields:{ Porcentaje: { $divide: ["$Numero_articulos.Num_articles", "$Numero_total.Num_total"] }}}
    ],{allowDiskUse:true})
```

collection_publication 13.1 sec.

Key	Value	Type
(1)	{ 3 fields }	Object
Numero_articulos	{ 1 field }	Object
Num_articles	2228904	Int32
Numero_total	{ 1 field }	Object
Num_total	4857061	Int32
Porcentaje	0.45889973381022	Double