# 

# Exercise 4

Database Design/Implementation

16th May, 2022

línea corta

## **INTRODUCTION TO DATA SCIENCE AND DATA VISUALIZATION**

2027202-1/2022-1

**PROFESSOR**

Luis Fernando Niño, Ph. D

[lfninov@unal.edu.co](mailto:lfninov@unal.edu.co)

**PREPARED BY**

Leyla Rocío Becerra Barajas

[lrbecerrab@unal.edu.co](mailto:lrbecerrab@unal.edu.co)

Camilo Alfonso Mosquera Benavides

[camosquerab@unal.edu.co](mailto:camosquerab@unal.edu.co)

Joan Gabriel Bofill Barrera

[jgbofillb@unal.edu.co](mailto:jgbofillb@unal.edu.co)

**Exercise 4:**

Database Design/Implementation

PROBLEM DOMAIN

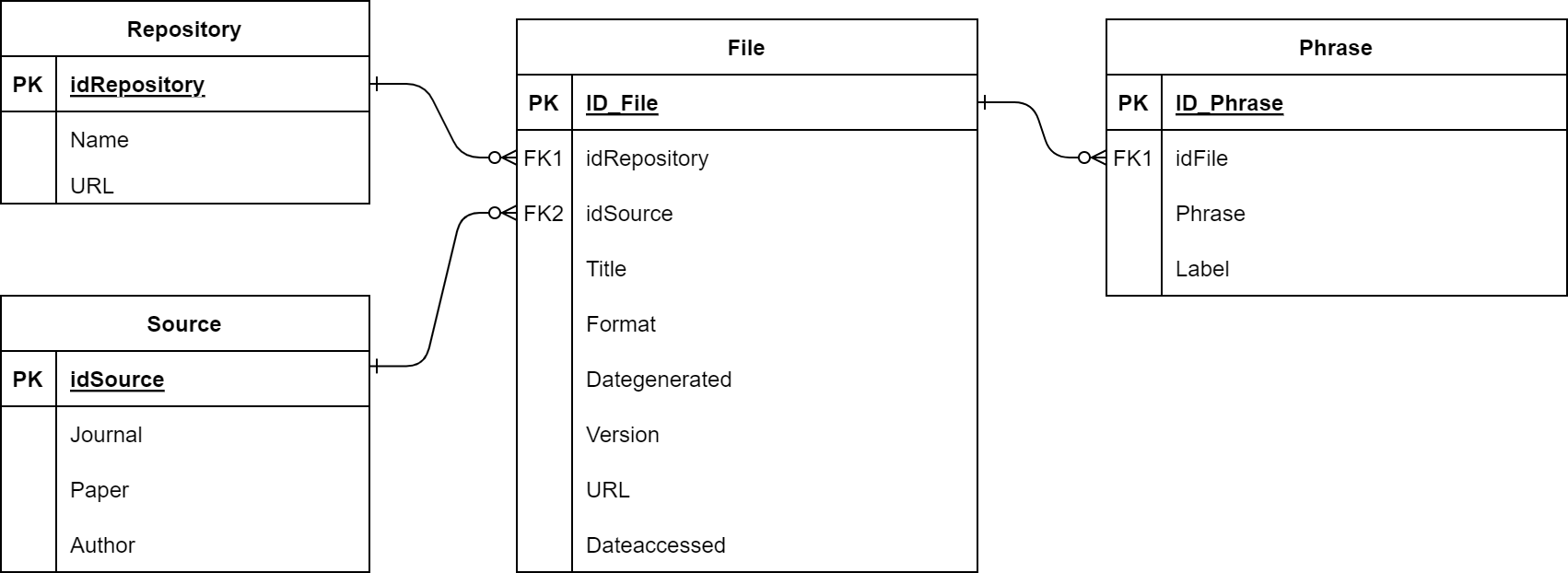
Our problem can be seen as a natural language issue, we want to be able to teach what humor is and what is not to a machine. For this purpose we have a particular dataset, so the focus would be on preprocessing/transforming the data extracting the information that is required to infer the characteristic of interest in the phrase. Accordingly, activities such as seeking other databases, joining or filter are not applicable. Nevertheless, in order to demonstrate that our model is able to generalize, some train/test splits will be performed in the data such as many other machine learning applications that can be found in the literature.

DATA DESCRIPTION

Our principal dataset for this work is the same that is proposed in the aforementioned case of Study paper and it is available on: <https://www.kaggle.com/datasets/deepcontractor/200k-short-texts-for-humor-detection>. It is a single file called ”dataset.csv” of about 15MB, it has two columns: ”text” and ”humor”. The former one is where the phrases potentially containing humor are, so its type is ”string”, on the contrary the latter is a boolean Column (where the possible values are ”True” or ”False”), indicating for each row if the sentence present in the text variable is humorous. In total there are 200.000 rows without repeated values. A major detail is that exactly 50% of the entries have True and the other half have False, so it is a balanced situation.

RELATIONAL DATABASE MODEL

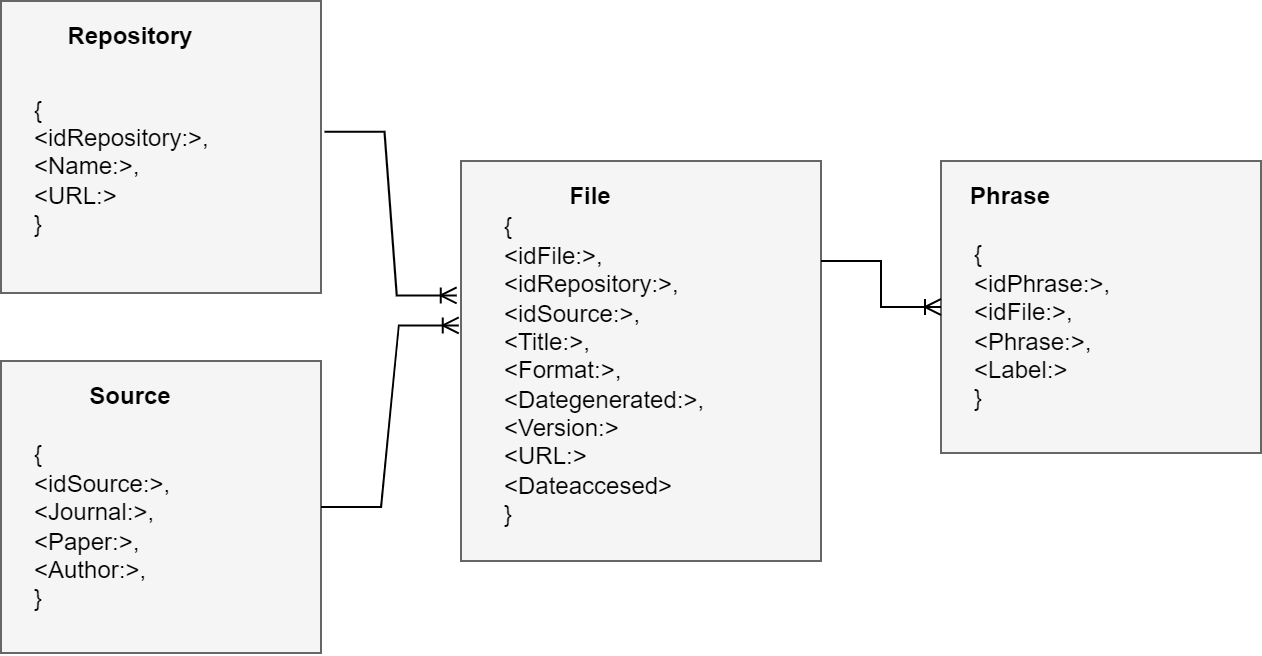
The data has been modeled using a Relational Database Model as is shown in Fig. 1. As already mentioned in the description of the data, the main dataset is a file with two columns: *Text* and *Mood*, which were proposed as the "*Phrase*" entity with *Phrase* and *Label* attributes respectively. Other entities such as Repository, Source and File are proposed to store metadata.



**Fig. 1** SQL model

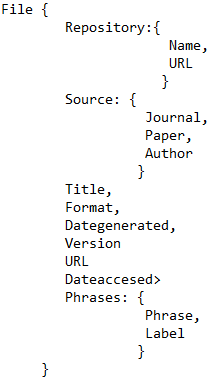
NOSQL MODEL

Similar to the SQL model it was proposed with similar collections as is shown in Fig. 2.



**Fig. 2** NoSQL model

This model store the information using collections in specific structure as is shown in Fig.3



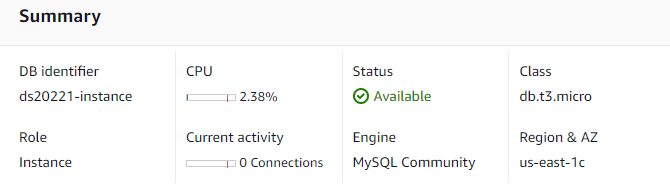
**Fig. 3** NoSQL structure

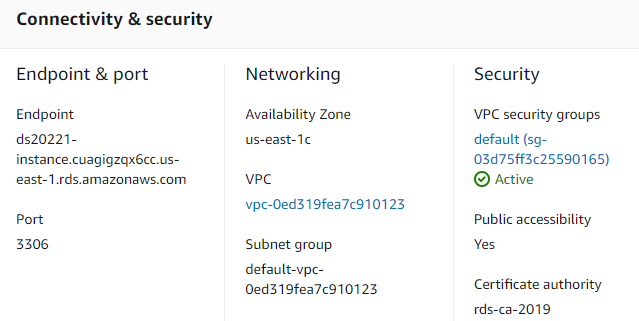
RELATIONAL DATABASE IMPLEMENTATION:

The deployment was done in AWS using the RDS service. It was created on MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly, embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

The deployment has the following features:



In order to create the database and tables according to the model, it was necessary to install mySQL Workbench.

Secondly, It was prepared a connection to the instance using the connection parameters:

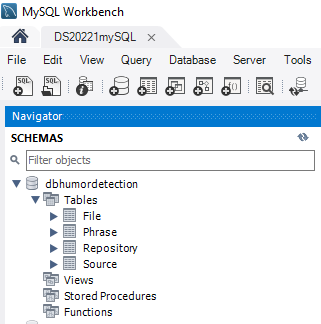
Hostname: ds20221-instance.cuagigzqx6cc.us-east-1.rds.amazonaws.com

Port\_ 3306

user: admin

password \*\*\*\*\*\*\*\*\*\*\*\*\*

After that, the database schema “dbhumordetection” was created and the tables according with the proposed model Repository, Source, File, Phrase:



Finally the process of populating the database was established and on each table as is shown by the following printscreens:

Table Repository

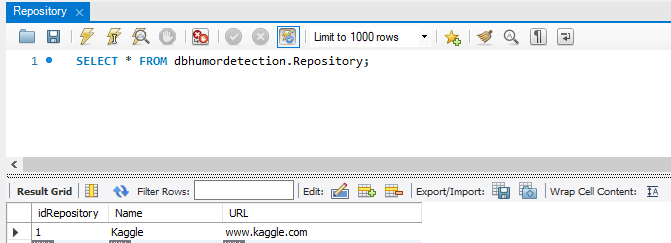


Table Source:

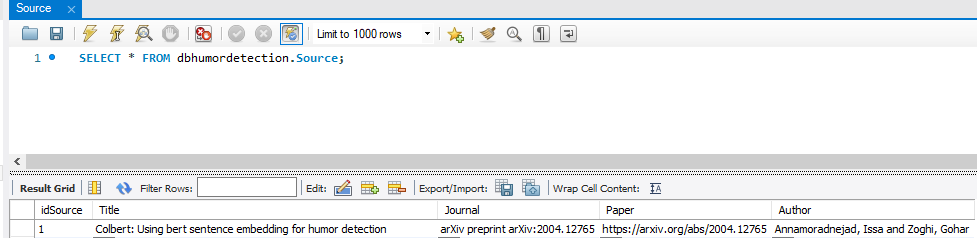


Table File:

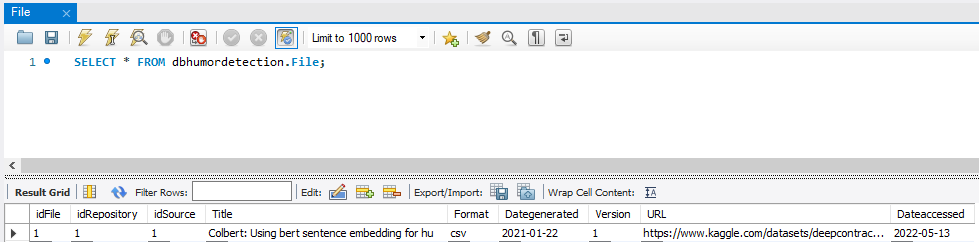
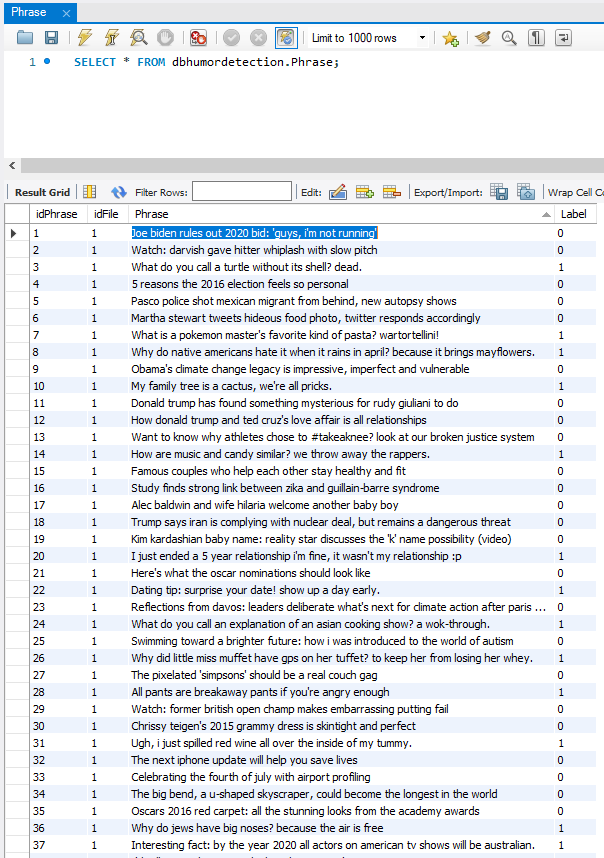


Table Phrase:



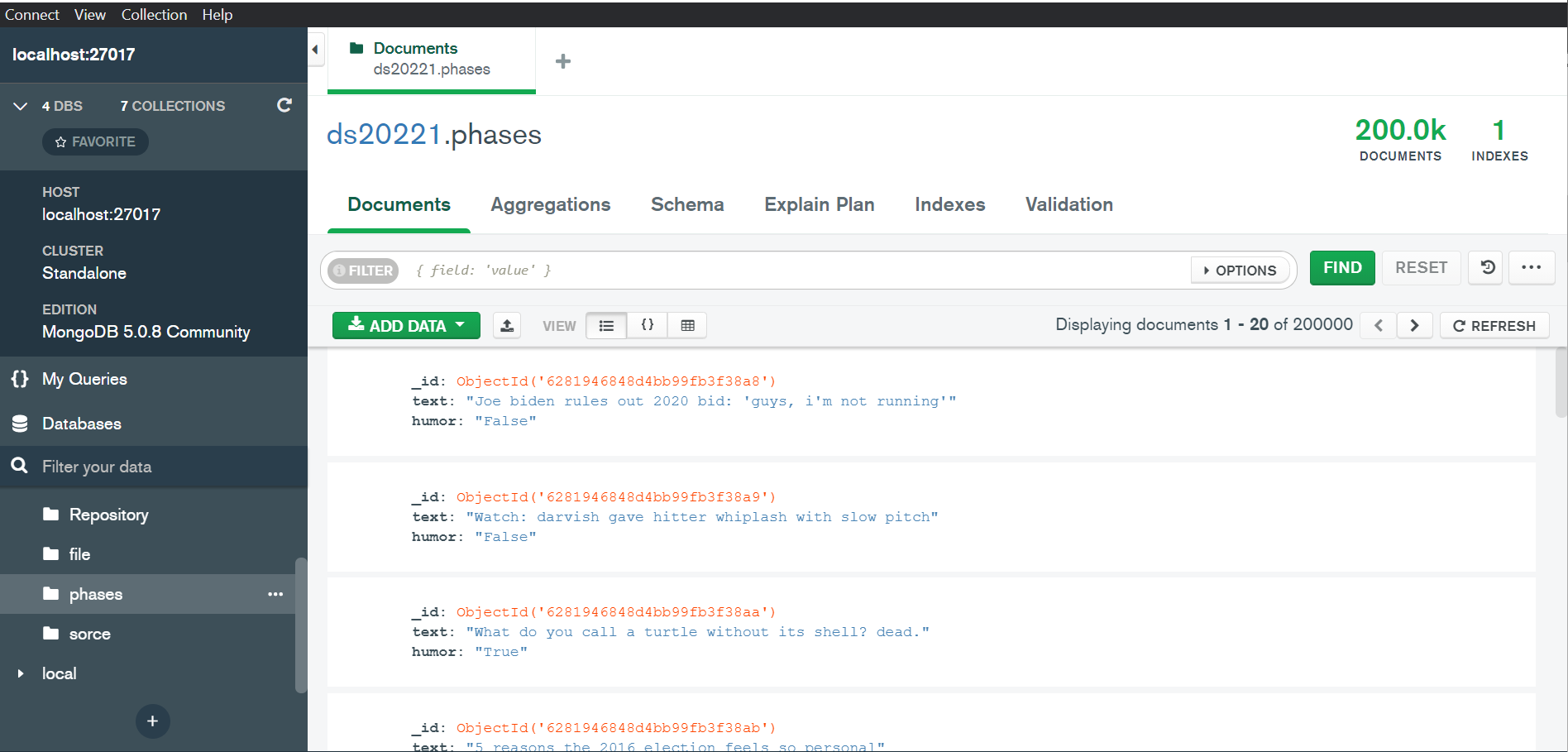
NoSQL implementation:

The deployment was done in MongoDB. MongoDB is a schema-free, document-oriented database written in C++. The choice of encoded format in MongoDB is JSON. This means that even if the data is nested inside JSON documents, it will still be queryable and indexable. As a document store based, it stores values (referred to as documents) in the form of encoded data MongoDB has a flexible storage system, which means stored objects are not necessarily required to have the same structure or fields. MongoDB also has some optimization features, which distributes the data collections across, being overall a more balanced and performance focused system.

It was created in MongoDBCompass, instance with the following features:



After that, the database schema “ds20221” was created and the tables according with the proposed model Repository, Source, File, Phrase:



e. A comparison of the relational and NoSQL databases implemented needs to be included, stating which of the two models is more suitable to handle your data.

Given the nature of our problem, we do not have a huge dataset that cannot be accessed efficiently in a relational way, actually it is not a Big Data problem nowadays. We have 200 thousand entries and our feature is really the text that is in a single column named “humor”. Additionally, in terms of computational complexity our target is simple, being our target a boolean value that says if the given phrase in “humor” is funny or not. The complexity of our problem lies in the language analysis that could understand such a complex matter in the human relationship as humor is. On the other hand, considering that sql databases are more mainstream and easier to understand, the project can benefit from that fact in order to have better data management to whatever process may appear in the development of the project.

**REFERENCES**

NoSQL Database: The Definitive Guide to NoSQL Databases. Accessed: 2022-05-16. <https://pandorafms.com/blog/nosql-databases-the-definitive-guide/>

MySQL :: MySQL 8.0 Reference Manual :: 1.2.1 What is MySQL? Accessed: 2022-05-16.

https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html

Meier, A., & Kaufmann, M. (2019). SQL & NoSQL databases. Springer Fachmedien Wiesbaden.

I. Annamoradnejad and G. Zoghi, “ColBERT: Using BERT Sentence Embedding for Humor Detection,” Apr. 2020, Accessed: Apr. 17, 2022. [Online]. Available: http://arxiv.org/abs/2004.12765.