|  |  |
| --- | --- |
| Icon  Description automatically generated | Data Analytics |

Can it be done? Text mining as a quality assurance method for the Orphanet nomenclature

Gemma Milman

March, 2023

Contents

[Definitions: 2](#_Toc129592680)

[Background: 2](#_Toc129592681)

[Business case: 2](#_Toc129592682)

[Objective 1 3](#_Toc129592683)

[Objective 2: 3](#_Toc129592684)

[Project plan 3](#_Toc129592685)

[Data collection 3](#_Toc129592686)

[Data cleaning and wrangling 3](#_Toc129592687)

[Removing missing data 4](#_Toc129592688)

[Removing duplicates 4](#_Toc129592689)

[Removal of irrelevant data 4](#_Toc129592690)

[Fixing structural errors 4](#_Toc129592691)

[Wrangling: extraction of clinical terms from the Orphanet disease definitions 4](#_Toc129592692)

[Wrangling: Align the extracted terms with the HPO terms 4](#_Toc129592693)

[Data exploration 5](#_Toc129592694)

[SQL vs no SQL databases 5](#_Toc129592695)

[Data import and creation of tables 6](#_Toc129592696)

[SQL Queries and insights 7](#_Toc129592697)

[Conclusions 10](#_Toc129592698)

## Glossary of terms

Below is a glossary of terms to help you navigate through the jargon used in this project.

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Clinical annotations** | A set of clinical phenotypes that are observed in specific disease. At orphanet,the phenotypes are listed for each disease in order of frequency that they are observed in the patient population. The HPO terms are used. |
| **Disease definition** | a short description the clinical signs and symptoms (phenotypes) that characterise the disease |
| **HPO** | Short hand for the Human Phenotype Ontology project, which project provides an ontology of medically relevant phenotypes, disease-phenotype annotations, and the algorithms that operate on these. |
| **HPO id** | each HPO term has a unique HPO identifier |
| **HPO term** | Standardised medical term to be used for each clinical phenotype. Each official term is accompanied by a definition and the synonyms |
| **Orphanet** | A knowledge base for rare diseases, assimilating lots of different information about each rare disease entity. At the centre of this is the orphanet nomenclature, which is an ontology which defines each disease, by name, orphacode, definition and it's synonymns. Accompanying each disease is clinically relavant informat, the associated orphan drugs, medical institutes/centres, and patient organisations |
| **Phenotype(s)** | Any observable characteristic or trait of a disease, such as morphology, development, biochemical or physiological properties, or behavior, without any implication of a mechanism |

## Background:

Orphanet is an international rare disease knowledge base, assimilating all aspects of data associated with all rare diseases. This ranges from oprhan drugs, expert resources, scientific and medical knowledge of each disease, and a nomenclature and classification system.

So what is a rare disease? A disease is considered rare if it affects less than 1 in 2,000 people. Whilst a rare disease on its own is uncommon, rare diseases as a whole are fairly frequent and the chances are that you know someone or have met someone with a rare disease (although you may not know they have a disease).

In addition, rare diseases field touches all organ systems: a rare disease may involve the heart, blood, or the brain for example. They can even involve multiple different systems in the same disease (e.g. Brittle cornea syndrome involves the eyes, ears, hip and joint mobility), these disorders are considered syndromic. The distinction between diseases and syndromes is not the focus here, and for ease, all diseases and syndromes are referred to as disease (but disease and disorder may be used interchangeably).

The Orphanet nomenclature is pivotal and consists of the disease name, orpha code (unique and stable identifier), synonyms and a disease definition. This nomenclatures is beginning to be implemented in clinical settings across the globe for the coding of rare disease patients. Having a code allows rare disease patients to be easily identifiable in the hospital systems, and should help improve their care by facilitating access to information on their particular rare disease. This is particularly important outside specialist centres where knowledge of rare diseases may be limited

As a former employee at Orphanet, I though it might be interesting to employ my new found coding and data analysis skills to address an issue that we had previously been unable to address.

## Business case:

Orphanet is committed to providing quality data. The data is currently manually curated and expert validated. Pre- and post-release procedures are in place to assure the quality of the data. Due to the technical constraints, no post-release, automated quality control is currently in place for the disease definitions.

The definitions characterise a disease in terms of its defining clinical characteristics, and should be stable, withstanding changes in evolutions in knowledge or medical developments (e.g. treatments that increase life expectancy). It is thus important to assure the quality of these definitions, as the nomenclature is now being implemented for coding of rare disease patients.

Given the recent developments in text mining and named entity recognition, the aim here is to exploit these techniques in order to provide a quality indicator for the Orphanet disease definitions.

## Objective 1

Develop an indicator of quality for the Orphanet disease definitions by comparing with the clinical annotations for each disease which are also curated by Orphanet.

The clinical annotations are a list of standardised medical terms (HPO terms), listing the signs and symptoms for each disease, and categorising them by frequency. The principal idea for the quality indicator is to extract the clinical terms from the disease definition (via text mining) and compare them to the most frequent signs and symptoms listed in the Orphanet clinical annotations.

## Objective 2:

Develop a model to improve mapping between the extracted clinical terms and the standardised medical terms (HPO terms).

The standardised medical terms come from the Human Phenotype Ontology project. The extracted terms will need to be mapped to the HPO terms in order to subsequently compare with the clinical annotations.

It is not expected that the mapping will be very reliable, thus the need to train a model to improve mapping.

The first objective is the preliminary step to the modelling aspect of this IronHack Project and will be the focus of this report. Whilst the second objective is out of scope for this RNCP report, it will be covered in the presentation.

## Project plan

The plan for this project is available on Trello [here](https://trello.com/invite/b/fYztH2c2/ATTIc4738b28090a656af851a0ad8d329518E6B3AAD0/finalproject).

## Data collection

As a not-for-profit with a mission to Contribute to generating knowledge on rare diseases, Orphanet allows access to much of the core data via Orphadata.com. Thus, the orphanet data as been downloaded as XML files from this site. The XML files were imported and parsed using the python Elmtree module, and the resulting list of dictionaries were read into a Pandas dataframe. These files include:

* Clinical annotations for Orphanet diseases (112689 rows × 5 columns)
* Orphanet nomenclature (10,675 rows x 8 columns)
* Linearised Orphanet file (7,241 rows x 6 columns)

The Human Phenotype Ontology (HPO) project provides an ontology of medically relevant phenotypes, disease-phenotype annotations, and the algorithms that operate on these. These standardised medical terms (HPO terms) are used to annotate the diseases in the Orphanet knowledge base. The HPO project permits free access to their ontology via download of an obo file. This file was imported and parsed via the pyobo module in python. A list was created for each attribute (e.g. term name, id, definition, synonyms) which subsequently converted into a single Pandas dataframe.

* HPO onotology file (16874 rows x 5 columns)

A part of this project requires matching of the Orphanet terms used in the Orphanet disease definitions to the HPO terms. The HPO website has a search engine in which a HPO term can be searched for via keywords; this functionality is also accessible via an API. The API will be used via a python script to return a HPO term for each of the Orphanet terms. The HPO API documentation can be found here: <https://hpo.jax.org/api/hpo/docs>

## Data cleaning and wrangling

Data cleaning is an important first step towards data analysis and includes assessing and addressing missing and duplicate values, removing irrelevant data, fixing structural errors and assessing outliers. Since the data here consists of text and identifier fields, outliers were not as issue. Instead, I will briefly address the other data cleaning methods applied here.

### Removing missing data

There were no null values in the nomenclature, clinical annotations or linearized disease list. The HPO ontology had missing values in the definition and synonym fields, as the corresponding HPO IDs and HPO Term names are required, the data entries were kept, and the null values replaced with blanks.

### Removing duplicates

On checking duplicated rows, the clinical annotations data contained 33 duplicated entries which were removed. At the stage of setting up the database, it became apparent that there were still duplicates when assigning primary keys to my tables. On checking the tables, clinical annotations and extracted terms for duplicates on only the identifier columns (OrphaCode and HPO\_ID) approximately 33 and 34 duplicates were identified respecitively in each table. These were dropped.

### Removal of irrelevant data

From the nomenclature files contains certain categories that are irrelevant. In particular the disorders are classed by type, e.g. disorder subtype, group of disorders and disorder. However there are a few other categories that pertain to the classification system and are not relevant to this project, these include the category ‘Category’, ‘Biological anomaly” and ‘Particular clinical situation’. In addition the data included some Non-Rare entities which were removed.

Some additional columns were removed such as Disorder ID (from the disease and definition table) and Expert link (from the definition data). The Disorder ID, is and internal, legacy ID for the diseases, but since the Orpha code is unique for each disease, keeping the Disorder ID is unnecessary. The Expert link provides a URL to the disease page on the Orphanet website, this has been kept in the Disease table for referencing purposes, but a duplication the definitions table is unnecessary.

### Fixing structural errors

The main problem to fix was the format of the HPO identifier which differed between the HPO ontology and the Clinical annotations. The official structure is ‘HP:#######’ where the hash tag represents a digit. The ‘HP:’ characters were removed in the clinical annotations, leaving only the digits. Having matching formats is essential for the relation database.

The definitions and Clinical Annotations both contained diseases (Orpha Codes) which were not in the main disease table. Thus, these extra entries (1356 for the definitions, 44 for the clinical annotations) were removed from the respective tables.

With regards to the HPO ontology data, the clinical annotations contained data for 122 HPO identifiers which were not in the HPO ontology file (possibly due to an update of the ontology since the clinical annotations were last made). These were equally dropped.

### Wrangling: extraction of clinical terms from the Orphanet disease definitions

Extraction of the clinical terms from the Orphanet definitions proved quite challenging. Clinical text has formulations and characteristics that are distinct to everyday prose (e.g. prose found in blogs). Initially, the plan was to use Named Entity Recognition (NER) to extract the terms, but this requires a pretrained model. Several options were explored (ClinicalBERT, MedCAT, SciSpacy, MetaMap[[1]](#footnote-1)) but due to time constraints this could not be explored any further. Instead, as a first iteration, the disease definitions were instead split using regex patterns and cleaned to remove stop words. Lemmatization of each term was tested but did not seem to impact the HPO matching later on.

### Wrangling: Align the extracted terms with the HPO terms

The next step was to match these terms to their corresponding HPO term. This was done using the module FuzzyWuzzy[[2]](#footnote-2), iterating over each clinical term and comparing with a list of all HPO terms. FuzzyWuzzy has four different ratio calculations that can be used for the comparison, these were tested to see which returned the best matches (the proportion of matches that were correct for matches scoring a ratio of 90 and above). Using my knowledge of the domain, the ratio cut-off of 90 was determined; the ratios below this threshold were very mixed, giving a mix of incorrect correct matches, and would need manual curation to determine correct matches.

Once the terms were matched based on the textual closeness, the corresponding HPO identifiers were mapped back to the terms using the Pandas map function.

## Data exploration

### Exploration of the Orphanet knowledge base

In the ‘linearized disease’ file, diseases are organized according to their primary classification (for the most part, the classification corresponds to the primary medical speciality for that disease). Given this information, let us start by looking at the repartition of diseases in the Orphanet knowledgebase according to the disease speciality. The tree map below (created in Tableau) shows the proportion of all diseases in each disease classification.

Chart, treemap chart

Description automatically generated

Figure .

The classification with the most diseases is the Developmental defect during embryogenesis. This classification is very broad, and includes diseases that could touch many different organs systems (e.g. intellectual disabilities are neurological by nature, and a cardiac malformation is cardiovascular by nature, but falls into this classification because it forms during development). With this information, it makes sense that this would be the largest classification. However, it is surprising that there is not equal repartition of diseases in other specialities. Neurological is the next biggest classification, and there is a gradual decrease in proportion for all other classes.

Now we know the repartition of diseases according to classification, let’s look at the proportion of diseases per classification that have a definition or clinical annotations. In figure 2, we can see that there is fairly complete coverage of definitions in each classification (i.e. almost every disease has a definitions). In contrast there in variable coverage of clinical annotations, the classifications with the most diseases annotated include Endocrine diseases (77% of diseases covered), Bone disease (71%), Development defect during embryogenesis (67%) and Neurological diseases (63%). Excluding Surgical cardiac disease and Genetic disease (for which there are very few diseases), the least clinically annotated diseases include Neoplastic diseases (24%), Maxillo-facial surgical disease (29%), Disorder due to toxic effects (32%) and Urogenital disease (33%). Based on this graph, I will focus on the developing an eventual model using terms from the most clinically annotated fields (at least as a first iteration).

Chart, bar chart

Description automatically generated

Figure

### Orphanet medical terminology

The Orphanet clinical annotations use standardized medical terms call HPO terms. As this project aims to extract these terms from the disease definitions, lets look at the number of distinct HPO terms per classification (figure 3).

Chart, bar chart

Description automatically generated

Figure .

Unsurprisingly, the classification with the most distinct HPO terms is the Development defect during embryogenesis. This means that there is a wide range of terms used with almost 5,000 distinct terms used, which makes sense given the different medical specialties that can be involved. It also means, that this might be the most challenging for the initial development of a model since there is not much time to pull together the training data. Instead, the diseases from the Bone or Endocrine classification might be a good starting point for the model, as there are few unique terms used (2,063 and 1,437, respectively) and have good coverage in terms of the clinical annotations available (see figure 2).

Now let us look to see if certain terms correlate with specific specialities (figure 4). From initial data exploration looking at all the HPO terms used in the clinical annotations, it looked like there was no correlation between the terms and specialities. In reality, there were too many terms to see the detail. It is possible that many of the terms do not correlate with specific specialities, but I was certain that certain terms did.

Figure 4 takes the top 20 most frequent terms for each classification and looks at the correlation. There are certain terms that correlate specifically with a classification, not appearing in any other classification. Other terms do appear in other classifications but generally not in all other classifications. The same is still true if we look at the top 100 terms (figure 5), although there is obviously more overlap between classifications.

A picture containing chart

Description automatically generated

Figure . Heatmap showing the correlation between the top 20 terms for each speciality. For reasons of readability, note that not all terms are displayed on the heatmap. Grid lines were added for readability.

Chart, histogram

Description automatically generated

Figure . Heatmap for the top 100 terms, showing that the correlation is generalisable. Since there are two many terms to show, gridlines have not been added in this case. Again, not all terms are shown on the heatmap.

### Analysis of the text extraction methods

Looking at the number of distinct HPO terms extracted from the definitions versus the number of distinct HPO terms used in the clinical annotations, would suggest that we extracted roughly 25% of all the terms used , as 2000 unique terms were extracted versus the 8000 in the clinical annotations.

Timeline

Description automatically generated with medium confidence

Figure

SIMPLE CHARTS BUT WELL DONE

Correlations

* 1. Look at no. diseases per classification, number of definitions
  2. Average number of HPO terms used per definition.
  3. Word cloud for most common and least common terms per speciality.
  4. Compare the definition terms with the very frequent terms in the clinical annotations, see the proportion of very frequent terms that are describe in the definitions
  5. Assess the reliability the mapping between the clinical terms and HPO terms

## SQL vs no SQL databases

SQL databases have been around since the 70’s, and are structured, relational databases. There as been a recent growth in non-SQL databases which can take on many different formats such as graph, key-value, column store and document store. Here I am going to briefly describe the characteristics of each why I have chosen an SQL database, specifically mySQL which operates locally on my machine.

SQL databases:

1. Data in SQL databases are organised into structured tables, the structure and constraints of which is determined by the administrator. This ensures that data is consistent, and the new data follows this structure
2. Relationships can be established between different tables by implementing primary and foreign keys. This means that data does not need to be duplicated as the primary data sources can exploited via their key, and also that access to sensitive information can restricted whilst the related data can still be exploited.
3. SQL databases use a Structured query language that are fairly standard across different platforms and are very flexible.
4. SQL databases are typically held on a single server, and thus need to be scalled vertically (e.g. adding more computing power) when the computational demands are increased
5. For this above reason,SQL databases can be limiting when there are many users and queries operating at the same time

No-SQL databases:

1. Whilst they are extremely flexible, they are not structured and thus data consistency is not enforced.
2. There are no, or very few, relations between data structures
3. Data is typically nested or merged in a few collections, with a collection serving a particular puropose.
4. Both horizontal and vertical scaling are permitted, this offers great performance particularly when there is lots of data or users and queries operating at the same time.
5. The languages are platform specific and not necessarily as flexible as in SQL

For this project MySQL relational database management system was used because it is a small-scale project (only one user) and the relationships between tables permits exploitation of the 5 different related tables. The structured query language (SQL) permits flexible queries and is well documented, thus adapted for the purposes of the project.

### Data import and creation of tables

Since the raw data had been cleaned and wrangled in python, it made sense it export the data directly to MySQL using python. The following libraries were used to do this import: pymysql.cursors, sqlalchemy (create\_engine, text) and pandas. In addition a config file was imported containing the MySql password. The function used to do this is below:

Graphical user interface, text, application

Description automatically generated

Whilst, you can specify the data formats specifically for each column, I chose to simply check the formats after importing. Since, the data is simply (text or integer), the default import options seem to work well.

Of course the table can be created in MySql used the CREATE TABLE commands, but the above method was much quicker.

The primary and foreign keys were also set using python. This permitted to easily debug some issues, specifically the incoherencies in disease in the definitions table that were not in the main diseases table, and terms that were in the clinical annotations table that were not in the HPO ontology. The primary keys were set with the following function:

Graphical user interface, text, application

Description automatically generated

Before the foreign keys could be set, the clinical annotations and definition dataframes need to be cleaned of entries that did not exist in the primary HPO ontology and Disease data, respectively. The function to do this is below (note that the incoherences were visualised before removing):

Graphical user interface, text, application

Description automatically generated

The foreign keys were determined for three tables (clinical annotations, orpha\_definitions and extracted\_hpo\_terms) using the following function:

Graphical user interface, text, application

Description automatically generated

### SQL Queries and insights

1. Let’s look at the repartition of diseases in the Orphanet database by speciality:

Whilst I would expect the most frequent classification to be Rare developmental defect during embryogenesis, as this covers all specialities from intellectual disability to physical malformations, I think it’s surprising to see that there is not and equal spread between all other specialities. However, the rare diseases are mutliclassed, so this does not mean that these other systems aren’t as involved (e.g. a syndromic disease could have neurology as the primary speciality but bone and endocrine systems could also be involved).

For this project, it gives me an idea for which speciality to tackle for the modelling part of the project, as there will not be time to address all terms for all specialities in this week long project.

Graphical user interface, text

Description automatically generated

Table

Description automatically generated

1. Query to look at the number of definitions by disorder type. It is no surprise that the highest number of definitions is for the disease level, followed by malformation syndromes and then the subtype level (clinical, etiological and histopathological). Clinical syndromes are very particular, and not very frequent in the Orphanet database, so it is not unexpected that there are only 45.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. Query to look the number of clinical annotations by disease type:

In contrast to the query above there are fewer different disorder types that have clinical annotations, specifically they are restricted to Disorder and Subtype of disorder. This means that the malformation syndromes are missing clinical annotations, and is something that could be addressed the Orphanet team responsible. In terms of this project, this means we can discard the morphological syndromes and the morphological anomalies, as well as any HPO terms specifically associated to these types of disorders.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. For each diseases, let’s look at the number of HPO terms extracted from the definitions, compared to the number that matches the clinical annotations and the total clinical annotations:

The results show that very few terms were extracted from the definitions using the method employed, and subsequently very few matches were made with the ‘very frequen’ HPO terms in the clinical annotations for each disease.

Graphical user interface, text, application, email

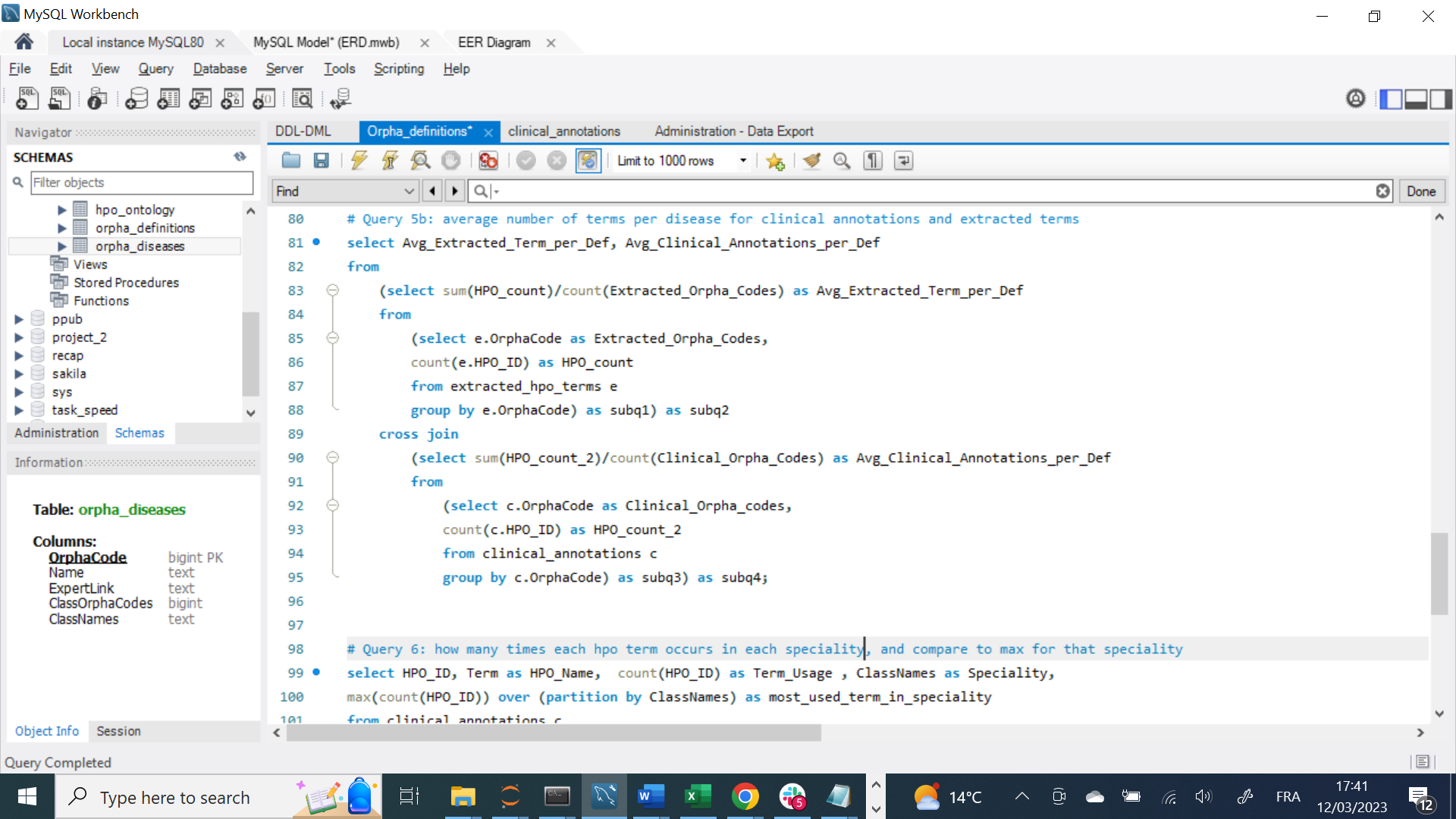
Description automatically generated

Graphical user interface, application

Description automatically generated

…

1. Following on from the above query, lets look at the average number of HPO terms per definition in the clinical annotations and the extracted terms:



Graphical user interface, text, application, email

Description automatically generated

# Conclusions

1. Create model to improve the phenotype HPO mapping
   1. Training data to be based on original HPO mapped terms, but correct the wrong mappings (by looking at where they didn’t match in the clinical terms). This needs to be done manually before fitting to a model.

1. Useful information : https://gweissman.github.io/post/using-metamap-with-python-to-access-the-umls-metathesaurus-a-quick-start-guide/ [↑](#footnote-ref-1)
2. Documentation: https://pypi.org/project/fuzzywuzzy/) [↑](#footnote-ref-2)