# IronHack Final Project: Orphanet rare disease data

## Definitions:

|  |  |
| --- | --- |
| **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.

The orphanet nomenclature is pivotal and consists of the disease name, and orpha code (unique and stable identifier) synonyms and a disease definition. This nomenclatures and orpha codes are beginning to be implemented in clinical settings across the globe for the coding of rare disease patients.

## 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 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.

The aim here is to text mine the published definitions for the described clinical signs and symptoms and compare this with the clinical annotations of the disease. The annotations of the diseases is a separate activity performed by a medical doctor, who quantifies the frequency of each sign and symptom for each disease based on the scientific and medical literature.

## Objective 1)

develop an indicator of quality for the Orphanet disease definitions by comparing the the clinical annotations for each disease.

text mine the definitions to define the phenotypes

Compare with the clinical annotations in the database.

These are HPO annotations, thus it is not expected that these terms will match 1:1

We can use HPO API to map the terms in the definition to there corresponding HPO terms, however, it is not expected to have 100% fidelity

Thus requirement to train a model to map Orphanet terms to HPO terms, the training dataset will need to be manually curated.

## 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 HPO project also allows 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) were converted into a single Pandas dataframe.

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

The clinical annotations data contained 33 duplicated entries which were removed. No other date contained duplicates

### 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

SIMPLE CHARTS BUT WELL DONE

Correlations

* 1. Look at no. diseases per classification, number of definitions
  2. Look at overlap between different classifications (proportion of diseases multiclassified
  3. Number of different HPO terms used in each speciality
  4. Average number of HPO terms used per definition.
  5. Word cloud for most common and least common terms per speciality.
  6. 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
  7. Assess the reliability the mapping between the clinical terms and HPO terms

INCLUDE MANY TO MANY

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)