ProteinPrediction II

PPII Ex3 - Week 3

Jonathan Boidol, Rene Schoeffel, Yann Spöri

Mapping IDs

1. `http://www.uniprot.org/mapping/` maps 2813 Uniprot ACs to 2801 Entrez Gene IDs
2. python script `map.py`
   - reads mapping and annotation files
   - maps Uniprot AC to HPO term #or root node, but we have better annotations for everything
3. output:

```
P00441 HP:0003394,HP:0002314,HP:0003202,HP:001
P31749 HP:0000400,HP:0004322,HP:0004325
P31213 HP:0000028,HP:0008736
...
```

specific annotations

- annotations are redundant (node and parents of annotation tree)
- use function `has_children` to prune annotations

```
P00441 HP:0003394,HP:0002314
P31749 HP:0000400,HP:0004325
P31213 HP:0008736
...
```

Graph data structure

- Graph object and node (= hpo term) object
- Graph can (only) be instanced by hpo file
- Graph is represented by a python dictionary
- SubGraphs by hpo ids may be created

  `Note, that at subGraphs don't change childnodes`
- Functions (until now)
  - +: get a subgraph which contains the nodoes from both graphs
  - -: get a subgraph that contains only the nodes that had been in both graphs
  - in: str: item with id, else node in graph
  - getHpoTermById: get a term object by an id (None if not in subgraph, although it might be in graph)
  - getHpoSubGraph: initialize a subgraph by leaves
  - getLeaves: get all leaves of the subgraph
  - getChildrens: get the children of a node