Open knowledge graph on clinical trials

Sandeep Kunkunuru[*] [†] [1,2]

[1]IIPH Hyderabad
[2]VaidhyaMegha Private Limited

[†] These authors contributed equally to this work.

[*] Correspondence: Sandeep Kunkunuru <hi18sk@iiphh.org>

Abstract

Using any trial id from across the globe find the associated diseases/interventions, research articles and genes. Also discover relationships b/w various medical topics through co-occurrences in articles. Query the graph using SparQL from cli or GraphQL using any API client tool ex: Postman or curl

Introduction

Sources

- WHO's ICTRP
  - Registries covered in ICTRP include :
- AACT Clinicaltrials.gov
- NLM MeSH
- NLM MRCOC
- NLM PubMed
- NLM PheGenI

Feature list

- Using GraphQL API knowledge graph can be queried using any API client tool ex: curl or Postman.
- Graph includes trials from across the globe. Data is sourced from WHO's ICTRP and clinicaltrials.gov
- Links from trial to MeSH vocabulary are added for conditions and interventions employed in the trial.
- Links from trial to PubMed articles are added. PubMed's experts curate this metadata information for each article.

- Added MRCOC to the graph for the selected articles linked to clinical trials.
- Added PheGenI links i.e. links from phenotype to genotype as links between MeSH DUI and GeneID.
- Added SparQL query execution feature. Adding CLI mode. Adding a count SparQL query for demo.
- 5 co-existing bi-partite graphs together comprise this knowledge graph. Bi-partite graphs are between
  - trial–> condition
  - trial–> intervention
  - trial –> articles
  - article –> MeSH DUIs
  - gene id –> MeSH DUIs

Demonstration

Querying knowledge graph using SparQL

```
$ java -jar -Xms4096M -Xmx8144M target/vaidhyamegha-knowledge-graphs-1.0-SNAPSHOT-jar-with-d
-m cli -q src/main/sparql/1_count_of_records.rq
...
Results:
--------
4766048^^http://www.w3.org/2001/XMLSchema#integer
```

Querying knowledge graph using GraphQL (via HyperGraphQL)

Start server

```
java -Dorg.slf4j.simpleLogger.defaultLogLevel=debug -jar lib/hypergraphql-3.0.1-exe.jar \
--config src/main/resources/hql-config.json
```

Start client

In a separate terminal execute GraphQL query using curl (alternatively use Postman)

```
$ curl --location --request POST 'http://localhost:8080/graphql' --header 'Accept: applicati
'Accept-Language: en-GB,en-US;q=0.9,en;q=0.8,kn;q=0.7' --header 'Content-Type: application/j
--data-raw '{"query":"{\n  trial_GET(limit: 30, offset: 1) {\n    label\n  }\n \n}","variabl

<https://www.who.int/clinical-trials-registry-platform/EUCTR2007-006072-11-SE> \<http://www.
<https://www.who.int/clinical-trials-registry-platform/EUCTR2007-006072-11-SE> <http://www.w
<https://clinicaltrials.gov/ct2/show/NCT02954757> <http://www.w3.org/1999/02/22-rdf-syntax-n
<https://clinicaltrials.gov/ct2/show/NCT02954757> <http://www.w3.org/2000/01/rdf-schema#labe
<https://www.who.int/clinical-trials-registry-platform/EUCTR2014-005525-13-FI> <http://www.w
```

```
<https://www.who.int/clinical-trials-registry-platform/EUCTR2014-005525-13-FI> <http://www.w
<https://clinicaltrials.gov/ct2/show/NCT02721914> <http://www.w3.org/1999/02/22-rdf-syntax-n
<https://clinicaltrials.gov/ct2/show/NCT02721914> <http://www.w3.org/2000/01/rdf-schema#labe
...
<http://hypergraphql.org/query> <http://hypergraphql.org/query/trial_GET> <https://www.who.i
<http://hypergraphql.org/query> <http://hypergraphql.org/query/trial_GET> <https://www.who.i
<http://hypergraphql.org/query> <http://hypergraphql.org/query/trial_GET> <https://www.who.i
```

Further reading

See the pandoc manual for more information on pandoc.

Authors struggling to fill this document with content are referred to Upper (1974).

Acknowledgements

References

Upper, D. (1974). The unsuccessful self-treatment of a case of "writer's block". *Journal of Applied Behavior Analysis*, *7*(3), 497–497. https://doi.org/10.1901/jaba.1974.7-497a