Open Targets
Software Developer Technical Test

Updated: 7 July 2022

Thank you for applying to join the Open Targets team!

Our two flagship digital tools - the Open Targets Platform and the Open Targets Genetics Portal -- rely on multiple ETL pipelines written in Scala and Python to aggregate, integrate, process, and analyse millions of objects.

In order to assess your technical skills and to give you a sense of the work we do, we would like you to complete a technical test.

Why do we ask candidates to complete a technical test?

The test will help us identify the skills that you bring to the team and will give us a sample of your work to discuss during the interview. During the interview, we will review your code and outputs and ask questions about your approach, any corner or edge cases you encountered, and improvements that you would make to your code.

About the Open Targets Platform

The Open Targets Platform is an open-source, comprehensive tool that supports systematic identification and prioritisation of potential therapeutic drug targets.

One of the key features of the Platform is our target-disease associations, which are unique target-disease pairs that are supported by one or more pieces of evidence (JSON/Parquet objects) that connect the two entities.

Each target-disease association is given an overall association score that ranges from 0 to 1. The overall association score is a harmonic sum of each of the individual evidence (JSON/Parquet objects) scores for each data source. A score of 0 corresponds to no evidence supporting an association and the closer the score is to 1, the stronger the association. To learn more, visit our association score and our evidence documentation.

Technical test: Parse a large dataset

The goal of this problem is to access data via FTP, quickly and efficiently parse a large dataset, and calculate statistics on the extracted data.

This work is a simple and very representative example of the type of Extract-Transform-Load work that we do at Open Targets.

Instructions

To begin the test, download our EVA evidence dataset in either JSON or Parquet formats. This dataset can be found in the `evidence/sourceId=eva/` directory. The files are available from http://ftp.ebi.ac.uk/pub/databases/opentargets/platform/21.11/output/etl/ . Please use the '21.11' release files.

Also download our targets and diseases datasets in either JSON or Parquet formats.

These files contain a series of evidence objects with their own individual evidence score. Our ETL pipeline takes these files and uses the individual evidence score to generate the overall association score for a given target-disease association.

Once you have downloaded the dataset, write a Python or Scala script that does the following:

1. Parse each evidence object to extract the `diseaseId`, `targetId`, and `score` fields.
2. For each `targetId` and `diseaseId` pair, calculate the median and 3 greatest `score` values.
3. Join the targets and diseases datasets on the `targetId` = `target.id` and `diseaseId` = `disease.id` fields.
4. Add the `target.approvedSymbol` and `disease.name` fields to your table
5. Output the resulting table in JSON format, sorted in ascending order by the median value of the `score`.

Tip: The resulting outputs should contain the following fields: 'targetId', 'diseaseId', 'median', 'top3', 'approvedSymbol', 'name'.

Each evidence object in the EVA evidence dataset defines a genetic association between a target and a disease. Since there are a few hundred diseases and thousands of targets, different targets will be connected to the same disease.

Using the same dataset, extend your Python or Scala script to:

1. Count how many target-target pairs share a connection to at least two diseases.

Submit your code and outputs by uploading them to your personal GitHub account and sending a link to the repository to administrator@opentargets.org by **Sunday, 17th July 2022 at 23:59 GMT**.

Tips and suggestions

1. Use either Python or Scala to complete the test -- our ETL pipelines are written using a mix of both languages.
2. Use any external libraries and provide clear instructions on how to install it: we encourage the use of libraries!
3. Working code, proper testing, useful README or documentation files, and code reusability are highly valued.
4. We will also be looking for solutions that take advantage of all available CPUs, finish in less time, and scale out the machine
5. Submit both code and outputs.
6. If you have any doubts about the data or your approach, describe your assumptions in your README file, and attempt to complete the test (even with pseudo code).
7. If you really get stuck you can email us at helpdesk@opentargets.org