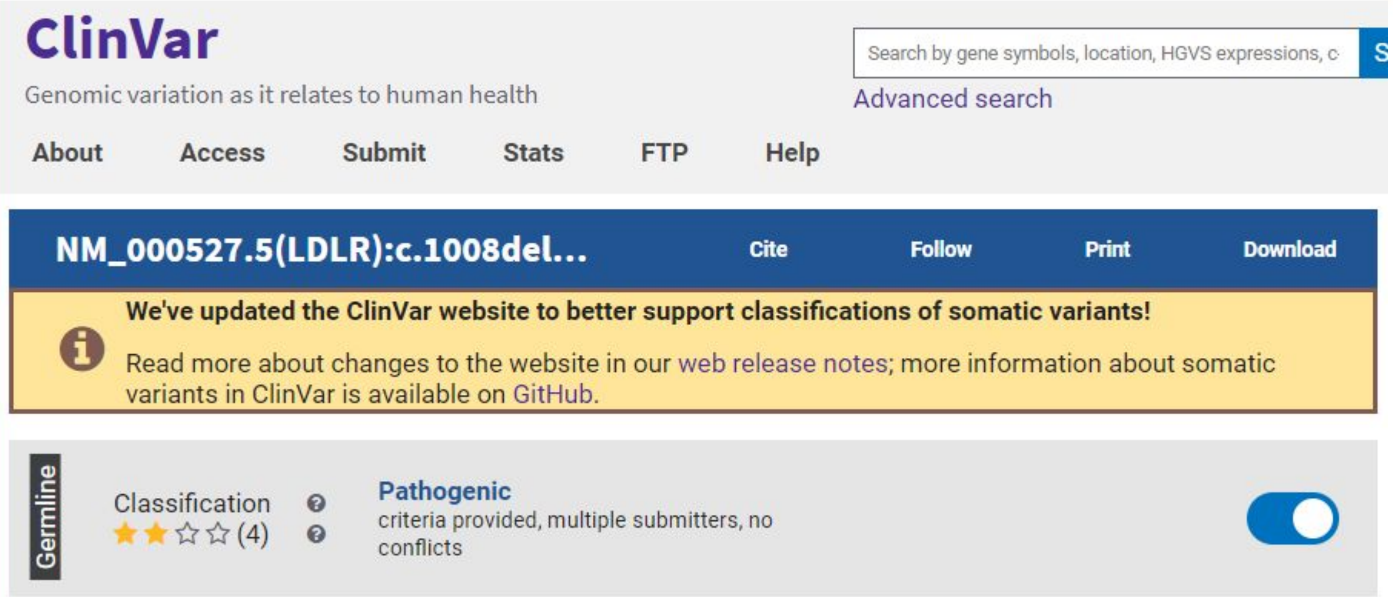
ClinCluster: Aggregating Disease Terms in ClinVar

NCBI's ML/AI Codeathon

Feb 26 - Mar 1, 2024

NIH National Library of Medicine
National Center for Biotechnology Information

ClinVar



ClinVar

Genomic variation as it relates to human health

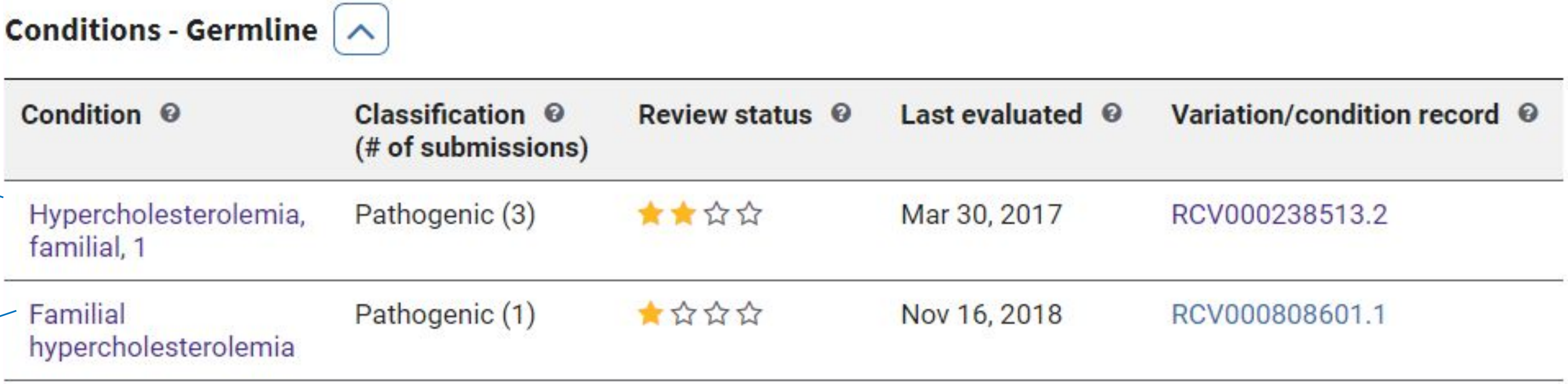
NM_000527.5(LDLR):c.1008del...

We've updated the ClinVar website to better support classifications of somatic variants!

Read more about changes to the website in our web release notes; more information about somatic variants in ClinVar is available on GitHub.

Classification ★★☆☆ (4)

Pathogenic
criteria provided, multiple submitters, no conflicts



Conditions - Germline

Condition	Classification (# of submissions)	Review status	Last evaluated	Variation/condition record
Hypercholesterolemia, familial, 1	Pathogenic (3)	★★☆☆	Mar 30, 2017	RCV000238513.2
Familial hypercholesterolemia	Pathogenic (1)	★☆☆☆	Nov 16, 2018	RCV000808601.1

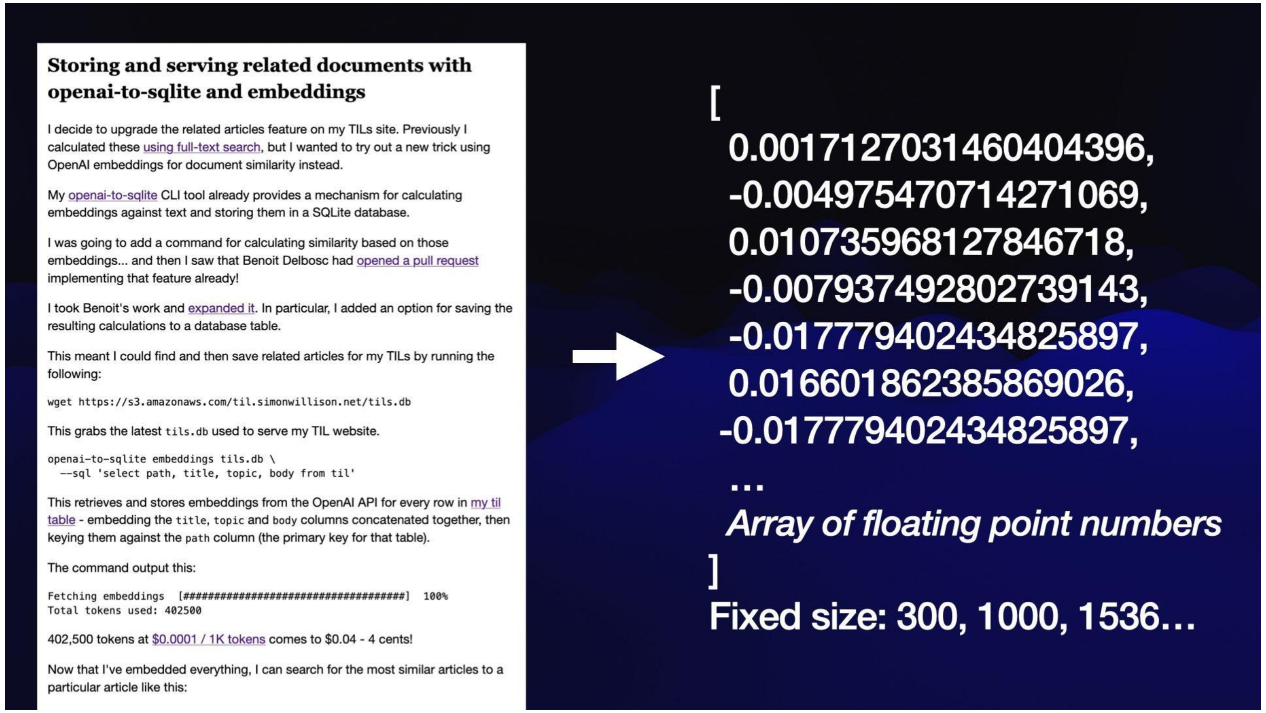
Hypercholesterolemia due to variants in the gene LDLR

Any inherited type of hypercholesterolemia

Problem statement: Diseases in ClinVar are very granular and result in many variant-disease records.

Can we use an ML/AI approach to aggregate disease terms in ClinVar to reduce the number of variant-disease records?

ClinCluster: Use LLM to aggregate these similar disease terms



Storing and serving related documents with openai-to-sqlite and embeddings

I decide to upgrade the related articles feature on my TILs site. Previously I calculated these using full-text search, but I wanted to try out a new trick using OpenAI embeddings for document similarity instead.

My openai-to-sqlite CLI tool already provides a mechanism for calculating embeddings against text and storing them in a SQLite database.

I was going to add a command for calculating similarity based on those embeddings... and then I saw that Benoit Delbosc had opened a pull request implementing that feature already!

I took Benoit's work and expanded it. In particular, I added an option for saving the resulting calculations to a database table.

This meant I could find and then save related articles for my TILs by running the following:

```
wget https://s3.amazonaws.com/til.simonwillison.net/tils.db
```

This grabs the latest `tils.db` used to serve my TIL website.

```
openai-to-sqlite embeddings tils.db \
  --sql 'select path, title, topic, body from til'
```

This retrieves and stores embeddings from the OpenAI API for every row in my til table - embedding the `title`, `topic` and `body` columns concatenated together, then keying them against the `path` column (the primary key for that table).

The command output this:

```
Fetching embeddings  [####################################]  100%
Total tokens used: 402500
```

402,500 tokens at $0.0001 / 1K tokens comes to $0.04 - 4 cents!

Now that I've embedded everything, I can search for the most similar articles to a particular article like this:

→

```
[
  0.0017127031460404396,
  -0.004975470714271069,
  0.010735968127846718,
  -0.007937492802739143,
  -0.017779402434825897,
  0.016601862385869026,
 -0.017779402434825897,
  ...
  Array of floating point numbers
]
Fixed size: 300, 1000, 1536…
```

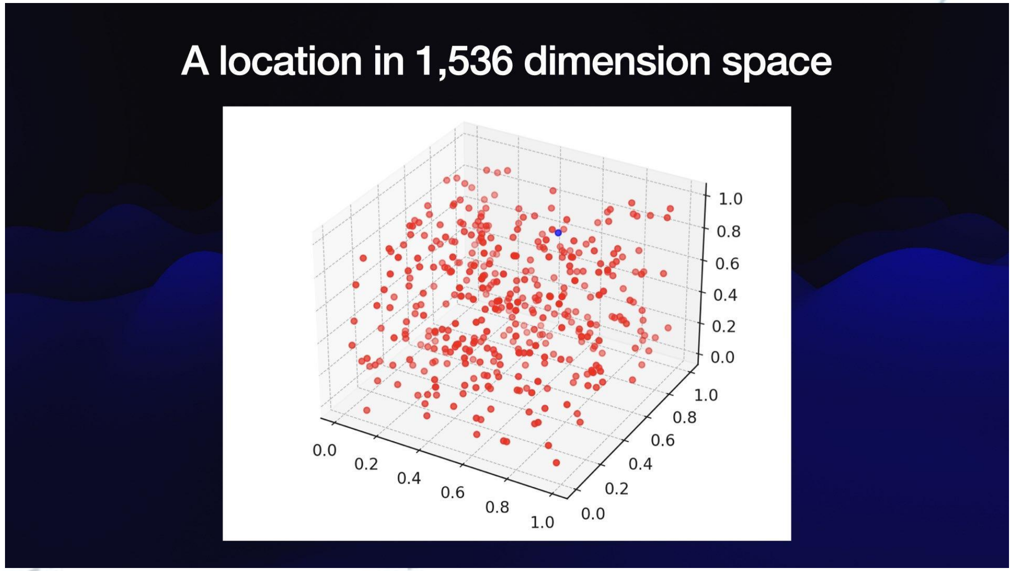
Same length, no matter how long the content is

Workflow

1. Feed the disease names to LLM
2. LLM gives a embedding (a array of floating point numbers) for each name
3. Use the embeddings to cluster the disease terms

Embeddings in LLM:

take a piece of content and turn that piece of content into an array of floating point numbers.



A location in 1,536 dimension space

Images from https://simonwillison.net/2023/Oct/23/embeddings/

DBSCAN: cluster the disease names

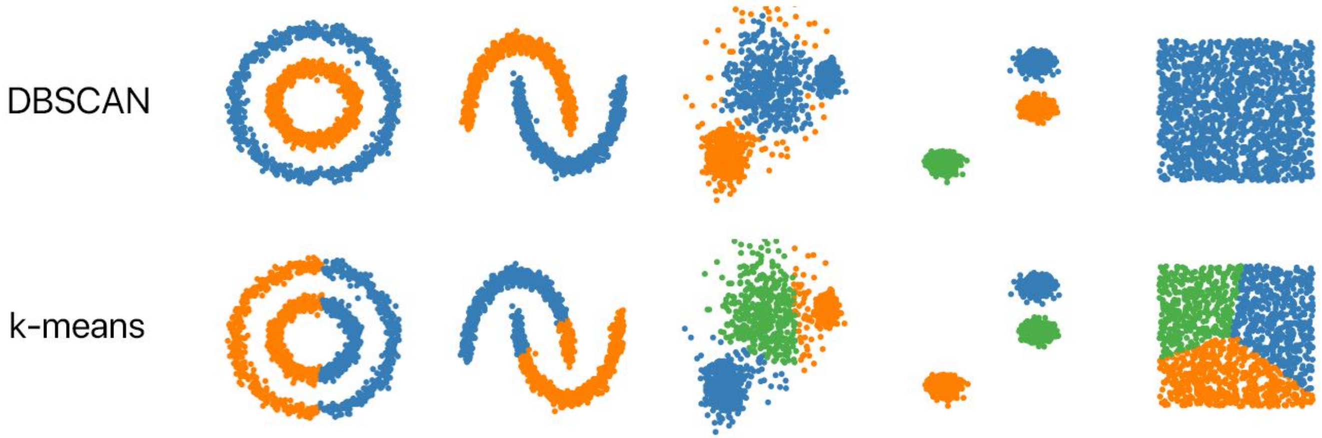
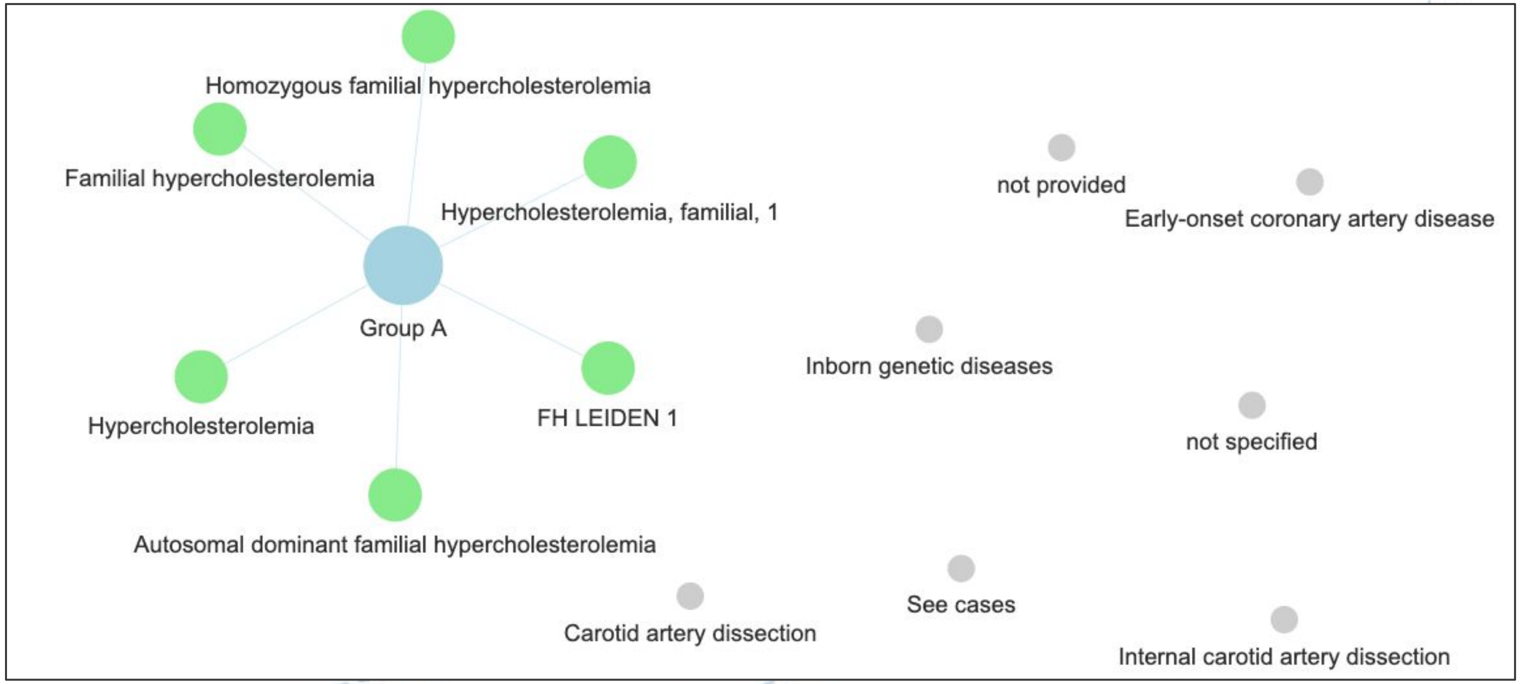


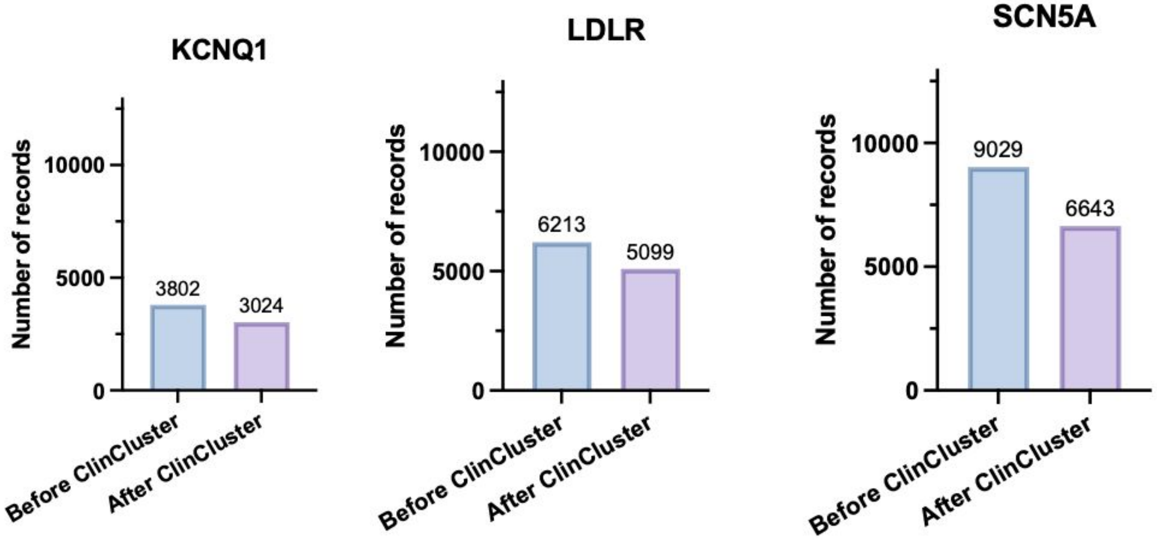
Image from https://github.com/NSHipster/DBSCAN

- We forked and modified llm-cluster package https://github.com/simonw/llm-cluster
- We change the k-means clustering algorithms to *unsupervised* DBSCAN algorithm for clustering the disease names.

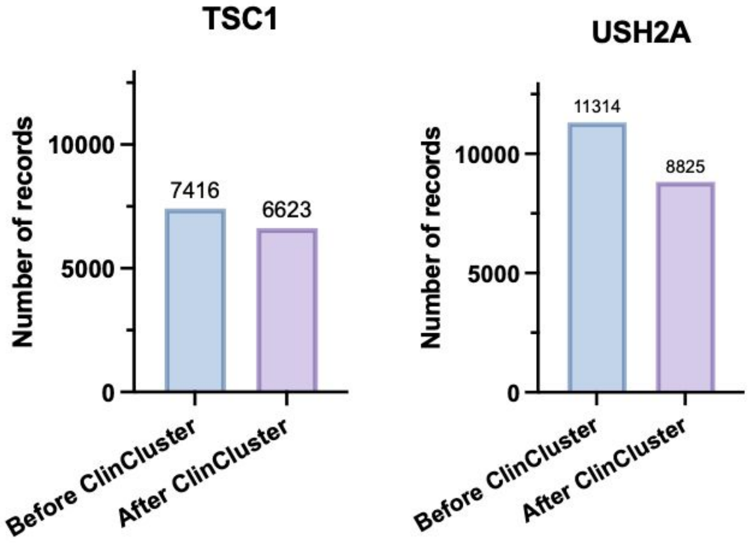
Result:


Homozygous familial hypercholesterolemia
Familial hypercholesterolemia
Hypercholesterolemia, familial, 1
Group A
Hypercholesterolemia
FH LEIDEN 1
Autosomal dominant familial hypercholesterolemia
not provided
Early-onset coronary artery disease
Inborn genetic diseases
not specified
See cases
Carotid artery dissection
Internal carotid artery dissection

ClinCluster: Performance metric



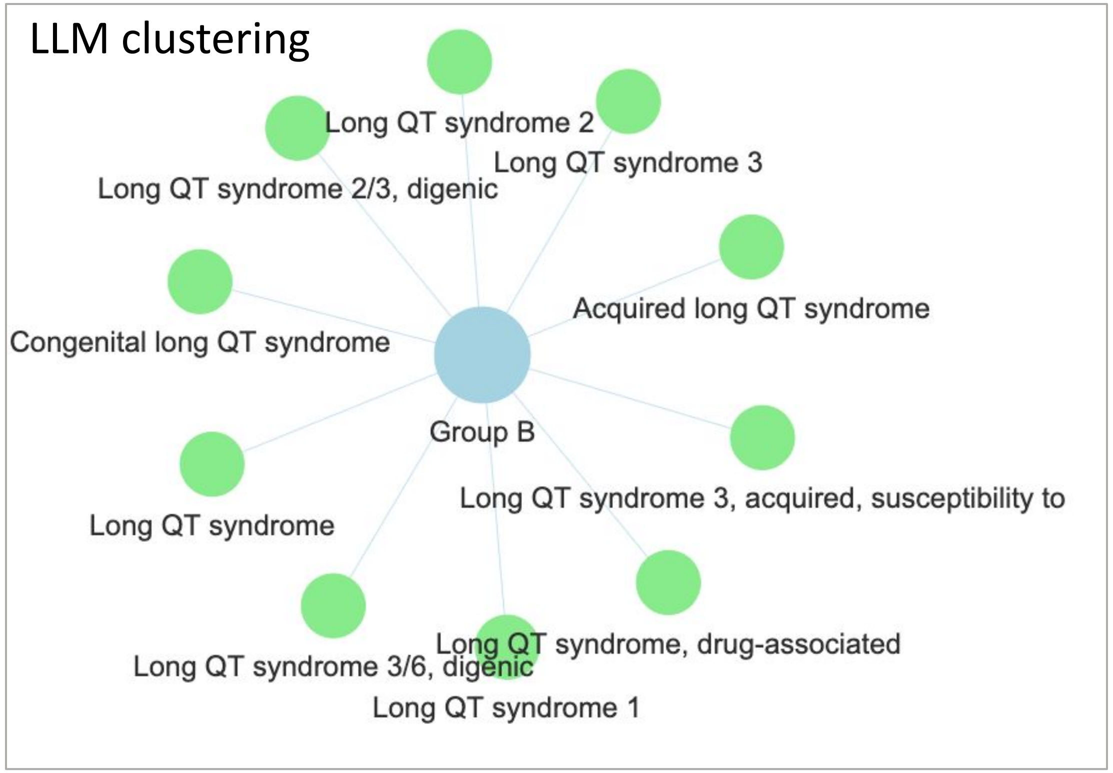
Significant reduction in the total number of RCV records

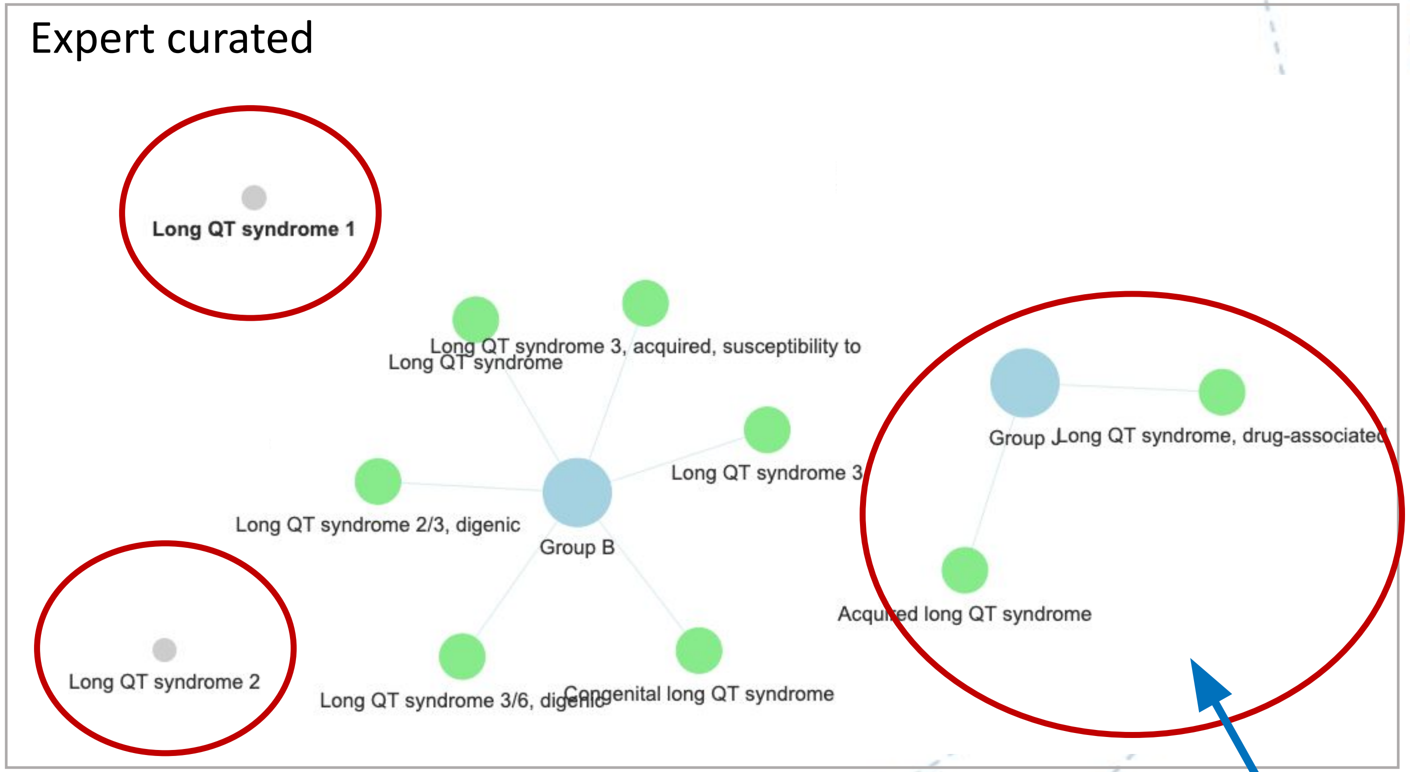


Disease term clustering is done by LLM clustering algorithm.

However, LLM clustering is not perfect…

Disease clusters for SCN5A gene


LLM clustering
Long QT syndrome 2
Long QT syndrome 3
Long QT syndrome 2/3, digenic
Acquired long QT syndrome
Congenital long QT syndrome
Group B
Long QT syndrome 3, acquired, susceptibility to
Long QT syndrome
Long QT syndrome, drug-associated
Long QT syndrome 3/6, digenic
Long QT syndrome 1


Expert curated
Long QT syndrome 1
Long QT syndrome 3, acquired, susceptibility to
Long QT syndrome
Group J
Long QT syndrome, drug-associated
Long QT syndrome 3
Long QT syndrome 2/3, digenic
Group B
Acquired long QT syndrome
Long QT syndrome 2
Long QT syndrome 3/6, digenic
Congenital long QT syndrome

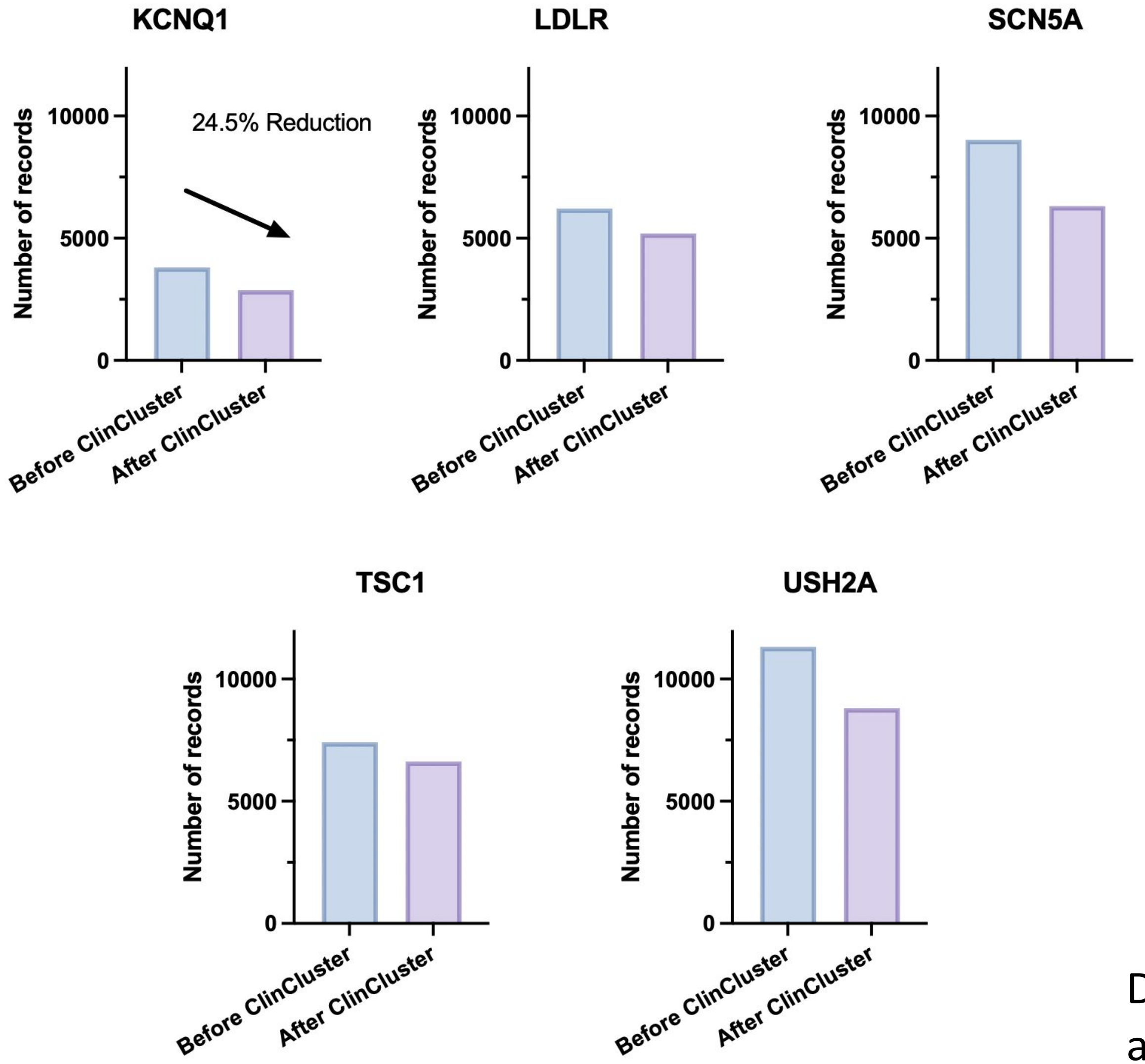
Circled diseases are caused by other underlying conditions

Rare conditions

State-of-art LLM might help

ClinCluster: Performance metric

Significant reduction in the total number of RCV records



KCNQ1

LDLR

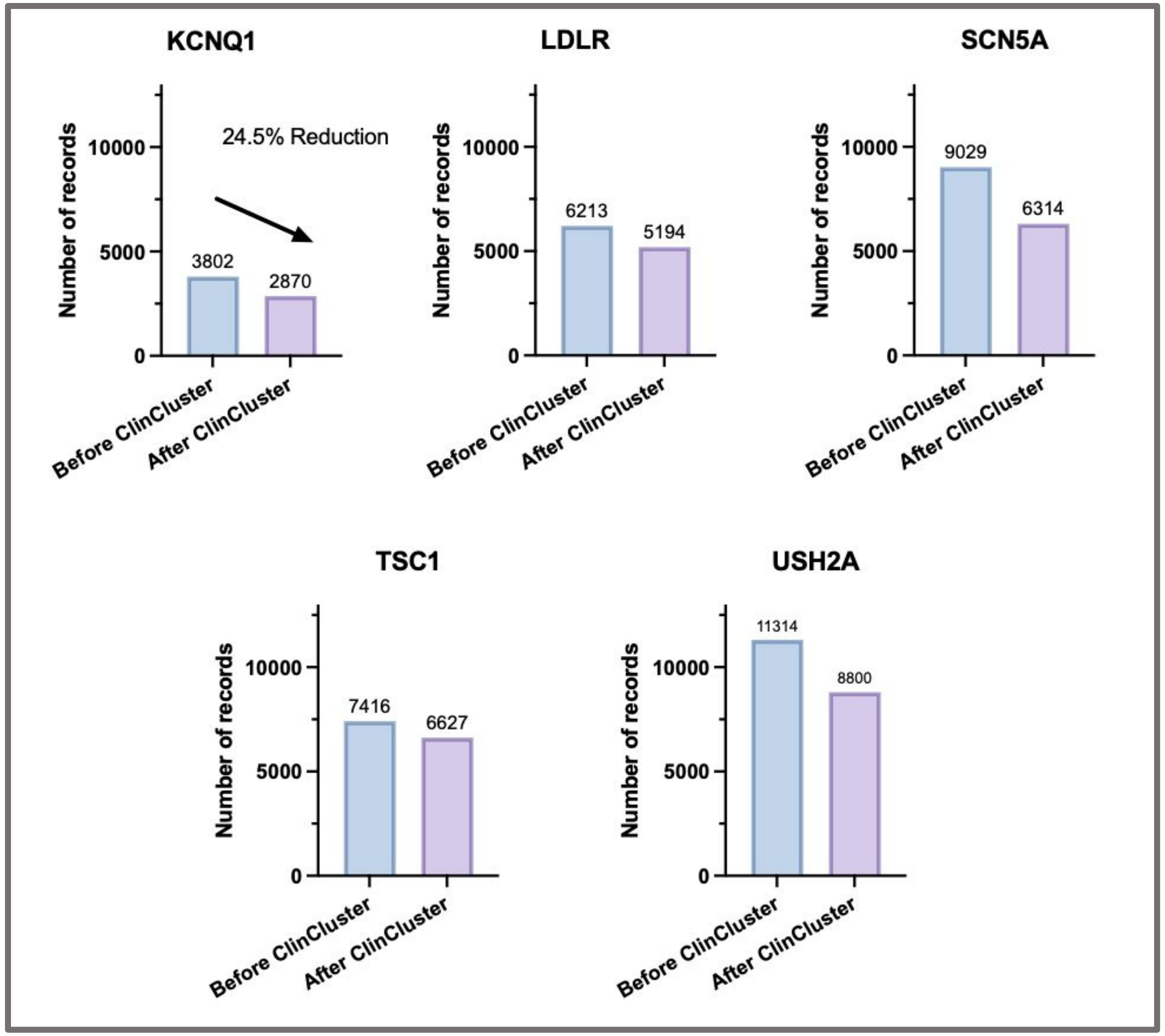
SCN5A

TSC1

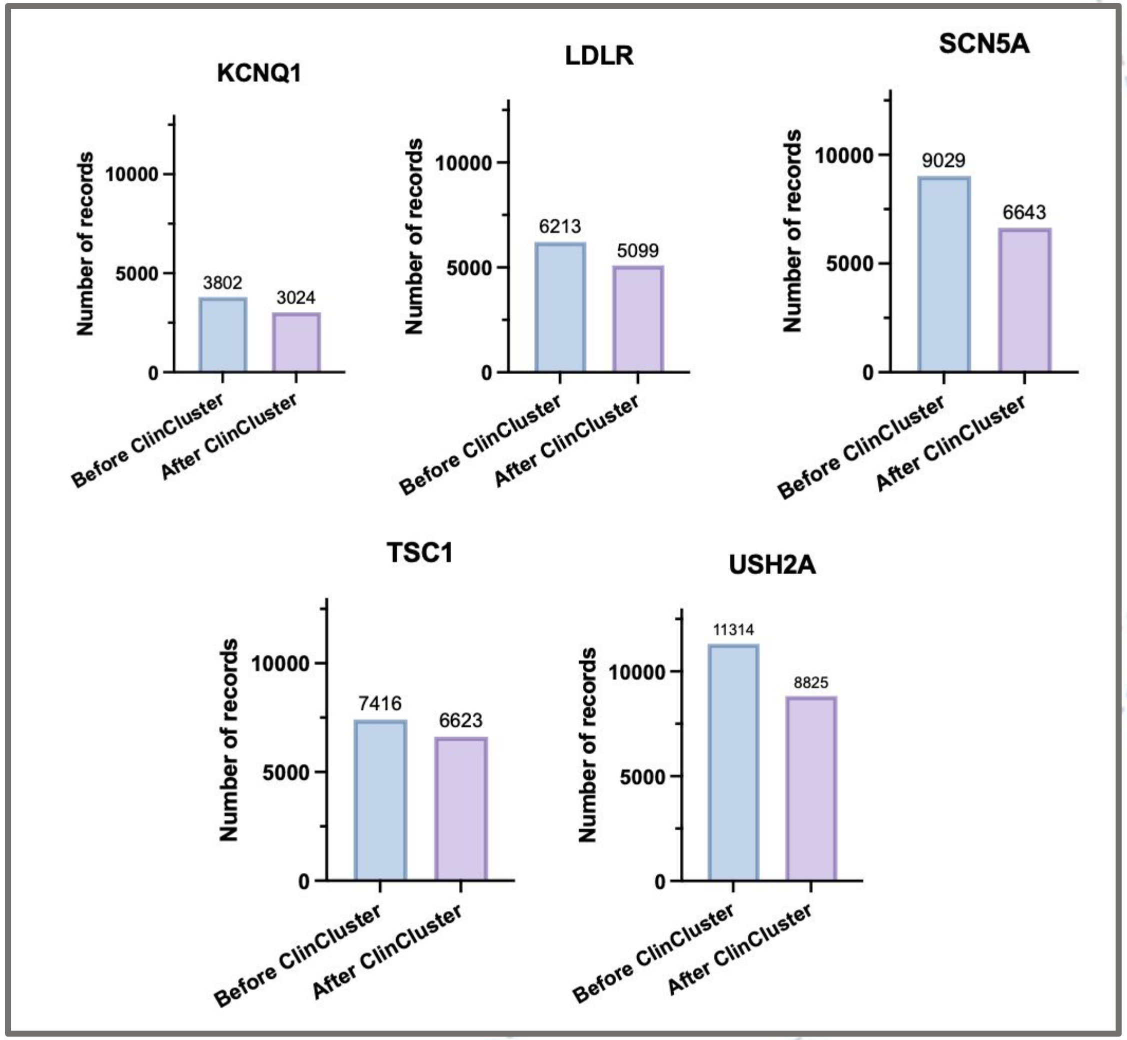
USH2A

Disease term clustering is augmented by human expert

ClinCluster: Performance metric



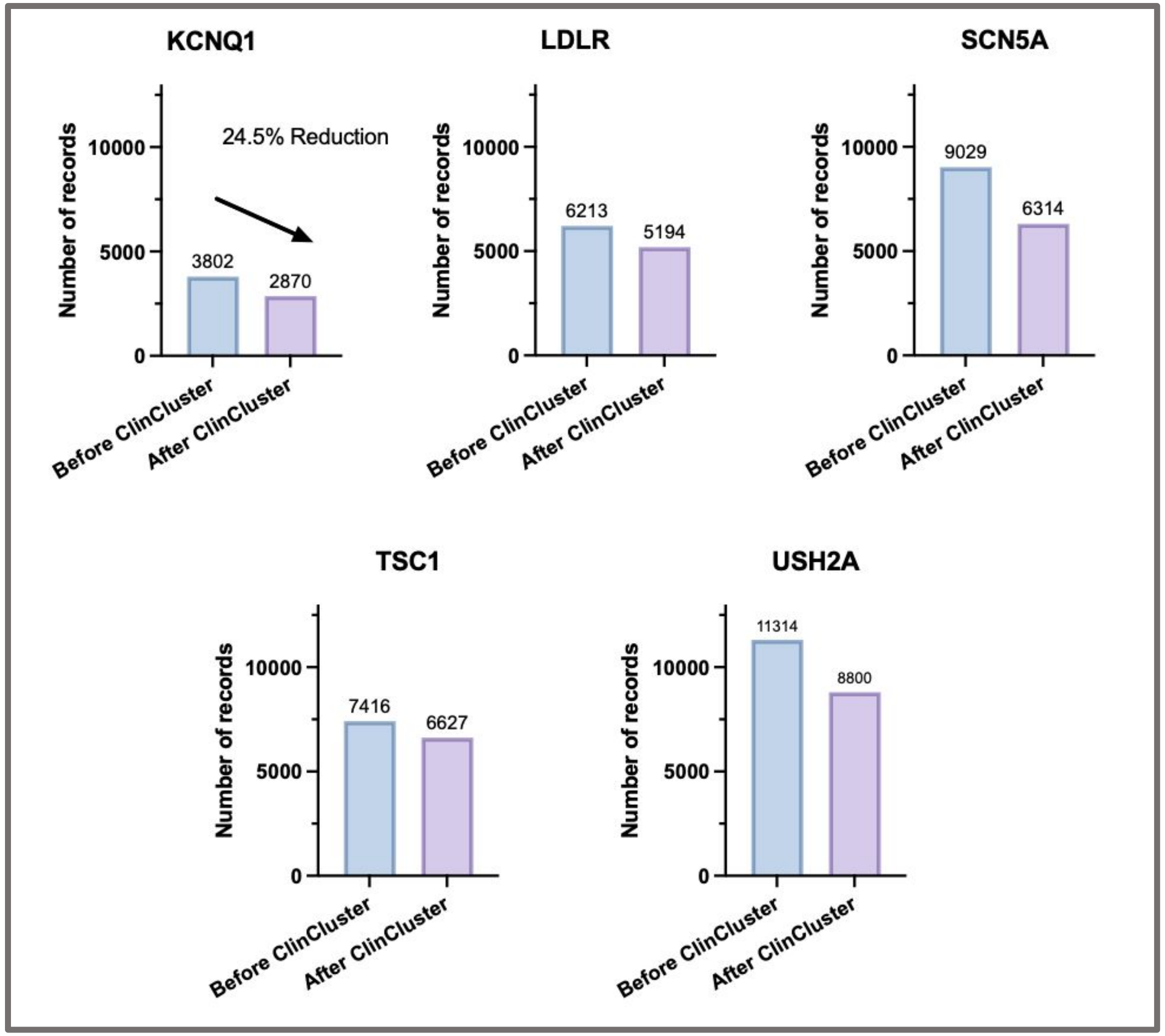
Disease term clustering is augmented by human expert after LLM clustering.

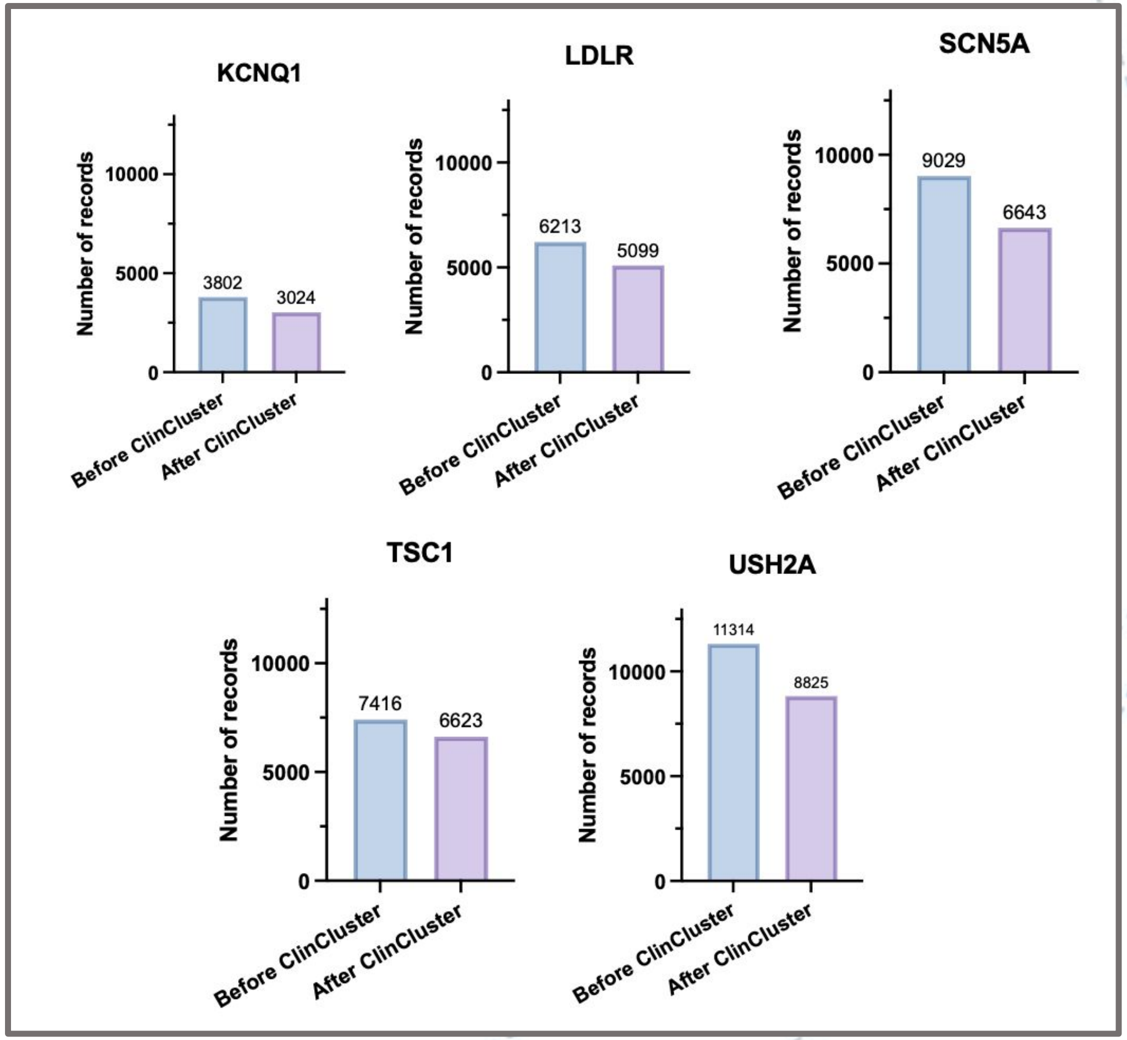


Disease term clustering is done by LLM clustering algorithm.

NIH National Library of Medicine
National Center for Biotechnology Information

ClinCluster: Performance metric

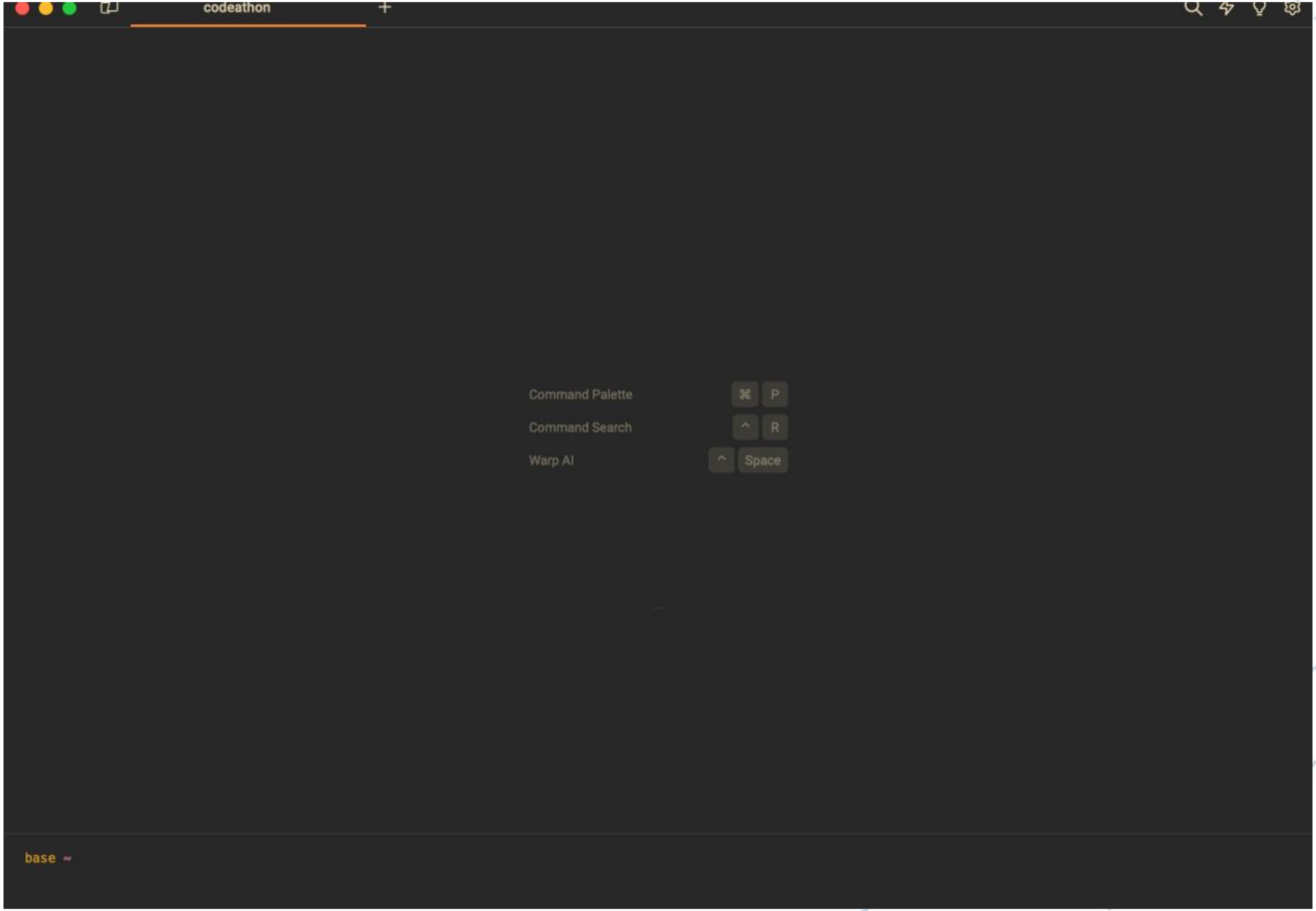




Reproducible implementation

nextflow

docker®



```
base ~
```

Acknowledgements

Lauren Edgar, NIH/NHGRI

Benjamin Kesler, Vanderbilt University

Nicholas Minor, University of Wisconsin

Michael Muchow, Unaffiliated

Rebecca Orris, NIH/NCBI

Wengang Zhang, NIH/NCI

Guangfeng Song, NIH/NCBI (Co-Team Leader)

Melissa Landrum, NIH/NCBI (Co-Team Leader)