**PhenClust, a standalone tool for identifying trends within sets of biological phenotypes using semantic similarity and the Unified Medical Language System Metathesaurus**

Jennifer L. Wilson1, Mike Wong2, Nicholas Stepanov3, Dragutin Petkovic2,3, Russ Altman4,5\*

1 Department of Chemical and Systems Biology, Stanford University, Stanford, CA 94305, USA

2 CoSE Computing for Life Science, San Francisco State University, San Francisco, CA 94312, USA

3 Department of Computer Science, San Francisco State University, San Francisco, CA 94132, USA

4 Department of Bioengineering, Stanford University, Stanford, CA 94305, USA

5 Department of Genetics, Stanford University, Stanford, CA 94305

\* Corresponding author: [rbaltman@stanford.edu](mailto:rbaltman@stanford.edu)

Department of Bioengineering

Shriram Center, 443 Via Ortega, Room 209

Altman Lab, MC: 4245

Stanford, California 94305-4145

(650) 725-3394

Keywords: interpretability, machine learning, accessibility, clustering application

Word Count: 1486

**Abstract**

**Objectives**

We sought to create an easy-to-install, stable, and reproducible implementation of Phenotype Clustering (PhenClust), a tool for clustering biological phenotypes.

**Materials and Methods**

We developed two PathFX+PhenClust Dockers with stable installations of either the 2017AA or 2020AA Unified Medical Language System (UMLS) metathesaurus versions. We applied PhenClust to two relevant biological analyses to demonstrate utility.

**Results**

The Dockerized PathFX+PhenClust generated easy-to-install access to the PhenClust tool, eliminating the requirement that the user to install the UMLS metathesaurus. PhenClust identified disease clusters for two biological analyses, demonstrating the tool’s utility.

**Discussion**

Docker containers can support dissemination and reproducibility of academic tools that are otherwise limited due to insufficient software support. PhenClust has applications to high-throughput biological analyses.

**Conclusion**

The Docker release of PhenClust achieved our objective of decreasing installation complexity and enabled demonstration of additional applications of the PhenClust tool.

**Lay Summary**

We created a Docker container to stably release an easy-to-install version of our tool, PhenClust. PhenClust is a computational tool that clusters biological phenotypes, such as diseases, based on their relatedness. PhenClust measures disease similarity using Unified Medical Language System (UMLS), however, installing this dependency is tricky and a failed installation prevents usage of PhenClust. Our docker system is a stable release of PhenClust with applications to multiple biological analyses.

**Background**

Tools that extract biological meaning from computational analyses advance biological research, yet a lack of stable and portable releases of these tools limit their utility and reproducibility. We developed such a tool and discovered limitations in the tool’s dissemination because of difficult-to-install dependencies. Our tool was motivated around the discovery that network methods identified tens to hundreds of phenotypes significantly associated with a drug’s protein-interaction network. The protein network generated for Metformin (published in [1]) was associated with “Diabetes mellitus, type 1”, “Diabetes mellitus, type 2”, “Pleural neoplasms”, and “Neoplasm of the rectum” among others. In this case we could manually identify disease groups (e.g. “oncology” or “metabolic disease”). However, some drug networks were associated with hundreds of phenotypes and we preferred a computational method for finding disease groups. We developed the Phenotype Clustering (PhenClust) tool to group diseases. At the time, we were unaware of other tools for completing this task.

We developed PhenClust around the Unified Medical Language System (UMLS) metathesaurus which has many advantages. The UMLS metathesaurus contains, to our knowledge, one of the largest biomedical ontologies, making it an attractive resource for understanding relationships between diseases. Additionally, we gathered our disease names from multiple databases and the UMLS meta-map[2] tools provided a robust method for mapping syntactically-distinct disease names to a common identifier system (the UMLS system uses the Concept Unique Identifier, or CUI term). Lastly, the umls-interface and umls-similarity[3] tools provided useful methods for interfacing with the UMLS database and calculating similarity between concepts. Calculating similarity between concepts was essential for understanding relationships between diseases and eventually discovering disease clusters, yet successfully installing a robust version of UMLS was difficult and this dependency was sensitive to routine operating system updates.

Interpreting relationships among biological phenotypes has applications beyond protein interaction network analysis. For instance, selecting a gene target for further drug development may require analysis of whether that gene is associated with multiple biological processes or side-effects. A scientist may prefer a gene target with a narrow set of phenotypic associations or may also consider a gene’s association to both safety and efficacy phenotypes. In a recent search of the PheWAS catalogue[4] – a database that links genetic variants to clinical phenotypes – a search for the tumor necrosis factor alpha (TNFα) gene recovered associations to 279 phenotypes and a search for the proprotein convertase subtilisin/kexin type 9 (PCSK9) gene yielded associations to 62 phenotypes. For the case of drug target prioritization, we anticipate that the number of gene-phenotype relationships will increase, and that having tools like PhenClust to summarize those relationships will be of broad utility.

Because PhenClust was viable as a general research tool and because PhenClust required difficult-to-install dependencies, our goal with this application note was to release a stable, easy-to-install version of PhenClust and demonstrate the tool’s application to multiple biological analyses. Dockerized systems are increasingly popular for releasing software tools with required dependencies and environment variables through a container object [5]. Docker containers have the advantage of extensive software support, which is not always available for academic software tools. Indeed, installing the Docker software and PhenClust container is drastically less complex than installing PhenClust from GitHub and the UMLS metathesaurus. The Github release of PhenClust (via the PathFX repo[1]) required the user to install UMLS metathesaurus and umls-interface.pl and umls-similarity.pl[3] dependencies. Further, our original release of PhenClust used UMLS 2017AA, and we have updated and included a new version using UMLS 2020AA.

Here we present an introduction to the PhenClust tool and two case studies to orient new users. This application achieves our objective of releasing a stable, easy-to-install, and reproducible implementation of the PhenClust tool.

**Significance**

The PathFX+PhenClust Docker is significant because it contains a stable and reproducible implementation of the PhenClust tool and the container reduces the complexity of installation, thus making PhenClust broadly accessible for research applications. PhenClust is a valuable tool because it is designed to increase the biological understanding of high-throughput computational analyses.

**Methods**

We developed two Dockerized systems that contain either the 2017AA or 2020AA UMLS metathesaurus releases, umls-interface.pl, umls-similarity.pl, PathFX code, PhenClust, and all required python dependencies. The Docker containers allow users to run PhenClust as part of PathFX analysis or use the code in a standalone fashion. Because access to the UMLS metathesaurus requires registration, we leveraged the National Library of Medicine (NLM) UMLS Terminology Service (UTS) authorization API to restrict access to NLM-registered users using the UTS API.

We further engineered the Docker container for relatively easy use and installation. We created a copy of the mysql directory during build time which is then mounted in the correct folder at run-time, which decreased the initial database loading time to 5 minutes (the conventional loading method is >2 hours). Installation of the Docker takes about 30 minutes, but this load time is not incurred after the Docker container is installed. Users have the flexibility to use the Dockerized UMLS metathesaurus image separately or for further development. To ease communicating with Docker, we provided a shell script that includes a help command for available PathFX options and helps deploy and stop containers safely without running into memory leaks. The shell script can detect when the database is actively loading and will wait for the mysql folder to be mounted at run time before executing commands. These scripts currently install the UMLS-2017AA or UMLS-2020AA versions.

To access the Dockerized container, users can create an account on PathFX-web

[6] (https://www.pathfxweb.net/). After logging in, users navigate to the “Download” page. On this page, users are guided to install Docker Desktop. Users are also reminded of the requirement to have an account with the NLM UTS. After establishing an NLM account, users can login, and the container download becomes available (**Figure 1**).

The PhenClust code requires minimal inputs. If used as part of PathFX analysis, the user can specify PhenClust as an option to the algorithm. With this option, PathFX will pass network results to PhenClust. To use PhenClust as a standalone tool, the user only needs to specify a text file with a Concept Unique Identifier (CUI) term on a new line. To demonstrate use of the tool, we created two examples (contained in *run\_phenclust\_ex1.py* and *run\_stand\_alone\_phen\_clust\_ex2.py*, both examples explained in **Supplemental File 1,** and also included in the Docker container).

PhenClust consists of two scripts: first, there is a python wrapper (calc\_lin\_matrix\_umls\_SO.py) to umls-similarity.pl that takes an input file, calls the UMLS tools, and generates a matrix file of similarity values. The second script (plot\_and\_cluster\_phenotypes\_SO.py) uses the results from UMLS to cluster diseases and generates text and image outputs. For a full description of the scripts, and the parameters see **Supplemental File 1.** All code and example analyses are included in the Docker container.

**Results**

The Dockerized versions of PhenClust are stable and portable. Further, the containers are distinct from PathFXweb where PhenClust is not accessible as a stand-alone tool and from the PathFX GitHub repository that requires the user installs the UMLS metathesaurus(**Supplemental Table 1**)**.** We measured run times for these implementations and observed 10 minutes – 44 hours depending on the number of input CUI terms and the computing environment (**Supplemental Table 2**). However, changing certain parameters can greatly reduce run times to under 1 hour (**Supplemental Table 2**).

We endeavored to use PhenClust to identify phenotype clusters for diseases associated with a network analysis from PathFX and to understand diseases associated with two candidate gene targets for developing a novel therapy for systematic lupus erythematosus (SLE) using a gene signature published by Arasappan et al[7]. Running PhenClust with each example dataset created two images and a summary table of results (**Figure 2,** **Supplemental Figures 1-4**, **Supplemental Files 1,3**). The labeled dendrogram represents a summary of the clustered results and shows similar clusters proximal to each other (**Figure 2,** **Supplemental Figures 1-4**). Cluster labels represent the top words mentioned in all descriptor names for phenotypes associated with the cluster. PhenClust also generated a tabular output of all phenotypic descriptors in a file that has the extension ‘*cluster\_membership\_\*.txt* (**Supplemental** **Table 3, Supplemental File 3**). For a full description of the results files and example use cases see **Supplemental File 1.** We used PhenClust to compare IL1R2 and FCGR2B and candidate druggable targets for SLE, an auto-immune disease. FCGR2B had associations to more auto-immune diseases and fewer potential side-effects than IL1R2. PhenClust summary analysis could inform how gene targets are scrutinized before further development.

**Discussion**

Interpretability (e.g. visual and textual interpretation) is a major challenge for system-scale biological analyses and interpretability tools can drastically improve these efforts. In our own efforts, we discovered that protein networks around a drug’s targets were enriched for several, and sometime hundreds, of disease phenotypes. We were motivated to consolidate these lists into broad disease groupings. Motivated by this finding, we generated the PhenClust tool that produced visual and textual interpretation of the input data – a list of phenotypes – and afforded the user a summarized version of this data. We further applied PhenClust to understand the uniqueness of disease-associated genes and discovered that PhenClust provided novel insights to considering these genes as druggable targets. In the latter example, we considered gene targets ranked for their association to SLE through a meta-analysis of microarray studies[7]. Meta-analyses are one technique used in prioritizing potential drug targets. However, a gene’s association to other distinct or related diseases, or side effects may preclude a target from further development. We envisioned that PhenClust could be valuable to understanding a gene’s association to multiple phenotypes aggregated from multiple databases.

Docker containers, a well-developed and documented commercial software platform, presented a valuable opportunity for enhancing dissemination of software tools and increasing reproducibility of computational studies. Even the best intended software releases can contain assumptions or hidden dependencies that limit successful installation[8]. Further, although research publications make data and code available, it may not be sufficient for full replication of the analysis [5]. And despite the idea that computational code could be more portable, it remains difficult to completely replicate computational analyses [5]. The dependency of PhenClust on a working installation of UMLS motivated us to pursue an alternative solution. The Docker platform was ideal because it allowed us to create a portable computational environment that improved reproducibility of our analyses and enabled easier dissemination of the tool.

The UTS is one of the largest biomedical ontologies and the UMLS platform is amendable to development and deployment of tools. The UMLS metathesaurus contains extensive resources, such as Meta-map[9], for mapping plain text terms from several other biomedical ontologies and sources to UMLS CUI identifiers. Further, tools such as umls-interface.pl and umls-similarity.pl have enhanced access to the UMLS ontology[3]. Our project directly benefitted from the UTS resources and allowed us to actively engage with this valuable resource. The Docker system removed hurdles to installing the metathesaurus and thus better enables scientists to focus on the interpretation and use of our tool, as well as the UMLS metathesaurus. Further, the platform was amenable to releasing a stable version of the original PhenClust tool (using UMLS version 2017AA) to support reproducibility of research and was amenable to data updates (using UMLS version 2020AA) to support evolving data sources.

**Conclusions**

We produced a standalone version of PhenClust, a tool that is valuable for grouping biological phenotypes based on relatedness. The tool is also broadly applicable to experiments where phenotypes can be mapped to UMLS identifiers. We extended the applications of the UMLS metathesaurus and anticipate that the tool will bring further users to the UMLS community. Lastly, packaging PhenClust with the UMLS metathesaurus in a docker contain increases access to our tool by making it simpler to disseminate and install and manage UMLS dependencies.

**References**

[1] J. L. Wilson, R. Racz, T. Liu, et al. PathFX provides mechanistic insights into drug efficacy and safety for regulatory review and therapeutic development. *PLoS Comput Biol* 2018; 14: e1006614–27.

[2] A. R. Aronson and F.-M. Lang. An overview of MetaMap: historical perspective and recent advances. *J Am Med Inform Assoc* 2010; 17: 229–236.

[3] B. T. McInnes, T. Pedersen, and S. V. S. Pakhomov. UMLS-Interface and UMLS-Similarity: open source software for measuring paths and semantic similarity. *AMIA Annu Symp* 2009;431–435.

[4] J. C. Denny, M. D. Ritchie, M. A. Basford, et al. PheWAS: demonstrating the feasibility of a phenome-wide scan to discover gene-disease associations. *Bioinformatics* 2010; 26:1205–1210.

[5] C. Boettiger. An introduction to Docker for reproducible research. *ACM SIGOPS Operating Systems Review* 2015; 49: 71–79.

[6] J. L. Wilson, M. Wong, A. Chalke, et al. PathFXweb: a web application for identifying drug safety and efficacy phenotypes. *Bioinformatics* 2019; 11:265-4506.

[7] D. Arasappan, W. Tong, P. Mummaneni, et al. Meta-analysis of microarray data using a pathway-based approach identifies a 37-gene expression signature for systemic lupus erythematosus in human peripheral blood mononuclear cells. *BMC Med* 2011; 9:65–10.

[8] J. Cito, V. Ferme, and H. C. Gall. Using Docker Containers to Improve Reproducibility in Software and Web Engineering Research. *Web Engineering* 2016; 609–612.

[9] D. Demner-Fushman, W. J. Rogers, and A. R. Aronson. MetaMap Lite: an evaluation of a new Java implementation of MetaMap. *J Am Med Inform Assoc*, 2017; 24: 841–844.

**Acknowledgements**

JLW was supported by UA, SPARK at Stanford, and a Sanofi iDEA Award.

**Figure Captions**

**Figure 1.** How to download PhenClust & PathFX Docker container via PathFXweb**.** Users are guided to install Docker desktop and reminded of the requirement to have an NLM account. The docker container is available for download after user authentication via the UMLS terminology service. The user navigates to the download page, selects “Login to NIH”, is redirected to the NLM page. On the NLM page the user selects the identity provider of their choice. After the license is verified, the user is redirected to PathFXweb and the docker container is available for download.

**Figure 2.** Clustering dendrograms from analysis of metformin-associated CUI terms with and without descriptive labels. PhenClust selects up to five of the most-frequent words from disease names in the cluster as a label for the dendrogram (left) or provides the number of terms grouped into a cluster or the CUI identifier for un-clustered CUI terms (right). The x-axis represents the between-cluster Euclidean distance as calculated by the linkage function from the fastcluster Python module. The first line of the figure title is an analysis name parameter specified by the user. The branch colors are assigned by default from the scipy.cluster.hierarchy.dendrogram function. The color threshold is set with 0.7\*max(linkage\_distance) to approximate clusters in the data.