Suggesting Reasonable Phenotypes to Clinicians

Laura Slaughter and Dag Hovland[0000−0002−3569−8838]

Dept. of Informatics, University of Oslo, Norway
{laurasla,hovland}@ifi.uio.no

Abstract. We describe a method for aiding clinicians in the process of reporting clinical features (abnormalities and symptoms) in newborns with suspected rare genetic diseases. The Human Phenotype Ontology (HPO) is well suited for such characterization, but impractical to navigate in a clinical situation. We enable different entry points depending on clinical specialization, detect certain types of errors that may occur, and suggest related phenotypes and candidate diseases to check for further clinical workup. Our main improvements are: (1) incompatible entries are highlighted and/or denied; (2) relevant related phenotypes are suggested; and (3) diseases are proposed in cases where the clinical report contains a complete profile and includes highly specific phenotypes.

Keywords: Human Phenotype Ontology · Reasoner · Rare Diseases

1 Introduction

The process of differential diagnosis is a systematic method used by clinicians to eliminate disease candidates, narrowing the possibilities, until the correct diagnosis is made. In rare disease diagnostics of newborns, this process can take years and involves genetic sequencing. Correct diagnosis can be a lengthy process that involves an entire healthcare team, including clinicians who communicate observations of abnormalities and geneticists who search within the gene sequence for variants, leading to a molecular diagnosis. It is an iterative process, and in newborns can be challenging, since some abnormalities will not manifest until specific developmental stages [11].

It is important to make a conclusive diagnosis for newborns as quickly as possible so that treatment can begin and it is known whether or not the condition is immediately life-threatening. It is also helpful for the parents to have less uncertainty surrounding the needs of the child. Having a full picture of symptoms, signs, family history and other clinical features is essential to the geneticists who need to narrow down the possibilities and match clinical features to specific genetic variants.

Tools for helping clinicians specify and communicate clinical features have been developed to be used within the differential diagnostic process. These tools are based on use of the Human Phenotype Ontology (HPO), a standardized vocabulary of phenotypic abnormalities structured as a directed acyclic graph

(DAG). Each term in the HPO describes a phenotypic abnormality, such as "Atrial septal defect" and terms can have parent terms, in the case of "Atrial septal defect", its parent is "Abnormal atrial septum morphology" and it is also the parent of several more specific child terms, including "Secundum atrial septal defect" and "Unroofed coronary sinus".

In addition to openly available entry tools using HPO codes, some hospitals may have other means for specifying clinical features and communicating these to geneticists. In our case, hospitals have implemented the use of a genetic testing requisition form within the Electronic Health Records (EHR) containing a list of phenotypical abnormalities, organized in categories related to bodily systems, anatomy or processes (e.g. neurological, developmental, muscle/skeletal) [6]. The observations are conveyed to the geneticist when the clinician requests genetic testing by checking off terms on the requisition form.

There are both legal and design-related issues associated with implementing the openly available existing tools for specifying clinical features of patients. In addition, the hospital requisition form is an inadequate means to communicate needed information with the geneticist. Problems observed when using existing open source HPO-related phenotyping tools include: (1) an overwhelming set of HPO codes to select in the user interface, allowing inconsistency in responses; (2) unclear meaning behind a "no" response for a term, and (3) limited prompts or suggestions for further patient workup. The requistion form, while more simplistic contains too few clinical feature observational terms, is not well-connected to HPO which has become a standard tool for geneticists, and provides no feedback whatsoever to the clinician.

Our aim is therefore to improve the specificity of the clinical features communicated to the geneticist by providing suggestions to clinicians to reduce inconsistencies, prompt for potentially missed observations, and help with deciding on further needed workup (e.g. additional lab tests, imaging). We developed a proof-of-concept web application[1] addressing these goals. The application uses semantic technology to leverage the HPO DAG and parent-child hierarchical structures. We use HPO, biomedical ontologies, and other knowledge resources to provide additional information to clinicians that increases the quality of information communicated further to the geneticist and helps with speeding up the diagnostic process in newborns. In this work, we address both the 1) suggestion service and 2) user "information interaction" issues that are connected to the user interface.

2   Related Work

Supporting the process of teamwork in the diagnostic process is a challenge [2]. The geneticist relies on the clinician to gather as much information as possible during clinical observations in order to inform the task of searching the genetic sequence for variants. These are used to search databases containing clinical features-disease associations.

---

[1] Available at https://gitlab.com/hovlanddag/rekvisisjon

Phenotips [3], has been developed for gathering a complete picture of the patient's clinical features (i.e. phenotype), including demographics, medical history, family history, physical and laboratory measurements, and physical findings. Phenotips uses a form-based data entry mechanism and the UI is said to closely mirror the clinician workflow. Phenotips allows for predictive search, when the user enters a term, it works to correct misspellings and suggests terms according to semantic relations. The tool provides some feedback to the user on the completeness of the phenotype, through "star ratings" which is a "sufficiency rating". They state that the goal is to make a profile "specific enough to exclude similar diseases and to identify model organisms with similar phenotypes that may have mutations in relevant genes or pathways." The "star rating" is, in some ways, a type of suggestion.

The user interface (UI) for Phenotips allows the clinician to select various HPO codes and also specify "NA/Yes/No". The forms used are not ontology-driven and do not make use of the HPO hierarchy. One could assume that if a patient showed "cachexia" and the clinician selected "yes" indicating presence, then it would be an inconsistent selection if the clinician also selected "no" to its parent term "weight loss". If a patient has "cachexia" which is defined as severe weight loss, then it would be inconsistent to also say that the patient does not have "weight loss". However, the Phenotips UI does not provide feedback to the clinician when such inconsistencies occur. Phenotips has opted for entering the upper navigable layer of HPO into the code. There may of course be good performance reasons for this, but this made it impossible to experiment with leveraging a reasoner to improve the HPO entry process live.

Another tool is Phenotero, an annotation tool that adds HPO codes + MONDO (Monarch Initiative Disease Ontology) to clinical notes as the user writes [5]. Phenotero is an extension of the Zotero tool, a citation management software, and is also available for use within Google Docs, LibreOffice, and Microsoft Word. The user writes clinical text and then manually adds a "citation", which is the HPO code, with the help of lexical matching search. Phenotero does not provide any completeness of phenotype, no analysis of the annotated coding, and no suggestions for related phenotypes.

Natural language entry tools also have been developed to help the clinician identify relevant HPO codes. The doc2HPO tool automatically recognizes HPO codes in clinical notes [8]. The tool allows the user to choose between different parsing engines, extracting concepts from the text and mapping to HPO codes. The system makes use of the Unified Medical Language System (UMLS) to map between identified UMLS concepts and HPO codes. The system is an HPO code curation/entry system and does not help clinicians with the process of deciding on possible further work-up or listing possible disease candidates.

Text mining tools such as NCBO Annotator+ [10] also can be used to annotate clinical texts for HPO codes, but there are known performance issues. There are problems with disambiguation of annotations (too many polysemic terms decrease precision) and for clinical text mainly, cleaning and reformatting of the text (abbreviations, spelling mistakes, unconventional sentence structures,

decrease recall). The Phenomizer [7] is a tool that is focused on differential diagnosis. The resulting output from the tool is a ranked list of diagnoses, however, it does also generate some related HPO codes (clinical features) to be considered for further work-up. These suggestions are not the main purpose of the tool, however, and are not well-displayed or considered within the user interface.

3   Suggestions

Our main goal is to suggest HPO codes, diseases and subsequent further patient work-up to improve the quality of the clinical profile provided to the geneticists. This has a dual purpose. The clinician becomes aware of HPO codes that are informative for geneticists and is able to provide a more complete picture of the patient. In addition, the second purpose is to motivate clinician-geneticist communication where both participate in the diagnostic process as a team. The completeness of the clinical profile of the patient is essential for a molecular diagnosis. The ability to diagnose rare monogenetic diseases through identification of gene variants requires this communication of phenotypes and patient characteristics with the cooperation of the clinician.

We provide suggested diseases and HPO codes to clinicians based on a simple lookup mechanism with information about disease-HPO code associations. These are found in the HPO ontology annotation file[2]. However, even a few HPO terms (phenotype codes) may lead to a long list of potential disease candidates and therefore gene variant candidates become too numerous. An ordering or prioritization is necessary. Because we are constructing a tool that will aid the clinician in diagnostic work, suggestions come in the form of HPO codes that point to further workup needed in the diagnostic process. The tool does not function as a differential diagnosis "clinician-replacement" machine. Rather we aim at suggesting diseases that are very specific and/or rare, thus augmenting the knowledge of the trained professional clinician. We also use disease information to inform HPO code priority and ordering. We aim for measuring the specificity of the disease given the currently entered phenotype. We measure the specificity by percent coverage: Let X be the set of phenotypes (HPO Codes) which are given a "Yes" click. For a disease i, let Φ_i be the set of phenotypes linked to it by the annotation files. The specificity of disease i is then $|\Phi_i \cap X|/|X|$, a number from 0 to 1, with 1 meaning that all features are present in the patient, and 0 no features present.

Unfortunately, this ordering gives a bias to diseases with few registered features, especially as the first few entries are made. Therefore, other heuristics are needed to order the many entries with a specificity of 1. We order entries (with the same specificity) in decreasing order of the number of patient features that are *not* registered on the disease. That is, by $|X - \Phi_i|$. This order works only if these assumptions are correct:

– All entered features are *pathologic*, in the sense that a diagnosis is wanted

―――――――――
[2] https://hpo.jax.org/app/download/annotation

- All the features stem from the same disease

This is clearly not the case in a general patient population, but in newborn patients with genetic disease, it is unlikely that two such diseases should occur. Also, newborns do not have the accumulated damage and injury that occur over a lifetime and may lead to other *pathological* features.

We plan to allow for additional orderings that the user can control. Another ordering that can be selected is:

- $|\Phi_i \cap X|$. That is, by how many disease-relevant features are seen in the patient. The problem with this ordering is that diseases with many linked features may be given unreasonably high priority.

3.1 Phenotype Suggestions

We use the ready-made entries, together with the annotation files and ontologies, to suggest phenotypes the clinician *should* look for, especially HPO phenotypes that might exclude certain diseases. For phenotype suggestions, we use the existing phenotype selection interface to signal suggested phenotypes. If the phenotype in question is already visible, we highlight it, otherwise we highlight the most specific superclass which is visible. We also show (an excerpt of) the list of diseases which would be more likely if that phenotype is present.

The prioritization of suggested phenotypes is not directly visible, as the output is not ordered. However, a heuristic for which phenotypes to show is important. We order diseases, in increasing order, by the ratio of not selected phenotypes:

$$\rho_i = |\Phi_i - X|/|\Phi_i|$$

We are experimenting with different cutoff values, $\rho_{\max}$, for ρ. For all diseases i with a $\rho_i < \rho_{\max}$, we display/suggest the phenotypes $\Phi_i - X$. Too high $\rho_{\max}$ risk giving the clinician a bias too early. Too low will only give feedback in very specific cases.

It is not a problem to *display* any number of phenotype suggestions, as the interface only shows the uppermost classes. However, the *usefulness* of the feedback is decreased if a few low-ρ diseases drown in a large number of diseases with higher ρ. Therefore we only show phenotypes from those diseases with the lowest ρ, but always at least n_{min} suggestions.

To summarize, the algorithm is to suggest the phenotypes $\Phi_i - X$ if

$$\rho_i < \rho_{\max} \ \texttt{or} \ |\bigcup_{k, \rho_k < \rho_i} \Phi_k - X| < n_{\min} \tag{1}$$

The most useful values of $\rho_{\max}$ and $n_{\min}$ needs further work. However, we expect they will depend on the clinical setting and experience with the interface, and may need to be configurable. In our initial experiments, we have used $\rho_{\max} = 0, 2$ and $n_{\min} = 10$.

3.2   User Interface

We provide help for customizing the UI for use in newborn diagnostics and the ontology provides the basis for prompting consistency in reporting. Similarly, other biomedical semantic technology projects have used ontologies to drive form-based data acquisition [4] [9].

The HPO phenotypes are provided as part of a form, with checkboxes to provide relevant information for the geneticist. By expanding the main header from the Norwegian requistion form, a list of phenotypes for that category is shown, with a selection for "Yes", "No", or "?". We added a new checkbox to distinguish between "No" and "?". Whenever "Yes" is ticked on a box, the direct subclasses in HPO is shown below. When the clinician checks No, a dropdown list of reasons is shown:

- Examined patient, not present
- Age of onset not reached
- Manifestation unrealizable
- Unable to examine patient

The three last items merit some explanation. Some symptoms can only present at certain ages. Especially newborns may often be lacking phenotypes that are only manifested after a certain age (e.g. developmental delays and cognitive impairment). Other phenotypes may be impossible in a patient, e.g. if organs are missing. Finally, in some patients, conducting tests to determine presence of an abnormality might be too invasive or otherwise high risk to the patient.

Fig. 1. Detail from the user interface showing a patient with cutaneous melanoma.

Note that in the existing requisition form implemented in our hospital, with only a box to check, the lack of a check could mean any of the reasons listed, in addition to "Not checked", which is indicated by the "?" in our form. In the case that the clinician provides some reason, this may speed up the communication between geneticist and clinician. There is, for example, no need to ask for clarification of a phenotype which cannot possibly be present at the current age or state of the patient.

Secondly, whenever "No" is clicked, the tool runs a background check for consistency. That is, to see if there any clicks of "Yes" on the same phenotype, or specializations of it. This kind of inconsistency can happen because the HPO

is not a tree, that is, it is possible to navigate to the same phenotype from different top-level entries. We anticipate that the most useful part of this feature is during cooperation. Over time, several clinicians will be working with the same patient, and the perception of the phenotype will change. Interactive feedback about conflicting entries may lead to earlier detection of conflicting views about the patient, such that measures to resolve this can be taken as early as possible.

4   Technical Implementation

Since we expect interactive answers using reasoning on a quite large ontology, it is important with an efficient ontology storage solution. For storing the HPO ontology we tried many different libraries, and the architecture allows switching between different stores. Since this is a technology in development we anticipate switching triple stores. The ontology can either be accessed directly by the application, or be accessed externally in a separate store. For the internal store we tried the Jena in-memory model, TDB, RDFOx and Sequoia. For external access we used a Fuseki triple store and tried with TDB and some others. Currently our application is using Sequoia in-memory (not externally). [1]. This is partially for simplicity, and we expect that in a production environment, having the ontology in an external triple store will be more maintainable. Sequoia is a quite new reasoner developed in Oxford, and supports only a subset of axioms and queries, but enough for HPO, as it mainly uses subclass axioms. The questions we ask are about navigation in this hierarchy: What are the super- and or subclasses of a given class.

The following artifacts are parsed and loaded: (1) HPO Ontology; (2) list of HPO entries shown as entry point of navigation (e.g. corresponding to current checkboxes in the requisition form); and (3) annotation files from HPO. These are tab-separated files where each line links a single HPO entry to entries in other ontologies. Some files link to diseases (OMIM, Orphanet and DECIPHER) and other files link to genes. The application must also maintain the current phenotype of the patient. One part of this is to maintain the boxes that are clicked Yes or No. This we currently do in a simple map. It could also be stored as data properties in a triple store that imports the HPO Ontology. The question of consistency could then be answered by a simple query to the triple store. However, none of the triple stores we tried could both store this information and answer the question fast enough, so we currently store the phenotype in-memory as a map and handle the consistency by asking several super- and subclass queries to the ontology and checking for errors. Although not ideal, as functionality that certainly exists in imported libraries is replicated, it is faster and also allows easily obtaining the reasons for inconsistency. The user interface is written in javascript and runs on the web browser. It uses REST calls to connect to the server part, which runs in a web container. All reasoning currently happens in the server. Our phenotyping tool is available [3] under an open source license.

---

[3] From https://gitlab.com/hovlanddag/rekvisisjon

5   Conclusion and Future Work

The project of replacing the Norwegian requisition form led to the need for high flexibility in the setup of the user interface, and we were unable to make use of existing tools such as Phenotips. We have applied existing semantic technologies (Sequoia, HPO ontology) and demonstrated how these can be used to quickly navigate large knowledge sources. We have also applied the use of semantic technology to the task of creating suggestions for pediatricians using the HPO ontology, including different types of heuristics for giving feedback to the pediatrician during data entry time. We implemented mechanisms for avoiding inconsistent entries in reporting clinical features. In conclusion, we use several techniques to suggest HPO codes to pediatricians to improve their ability to provide as much useful information in reporting the patient's clinical characteristics to the geneticists. The next step is to validate the software with formal user testing, including iterative usability testing with pediatricians.

6   Acknowledgements

References

1. Bate, A., Motik, B., Grau, B.C., Simancik, F., Horrocks, I.: Extending consequence-based reasoning to SRIQ. In: Baral, C., Delgrande, J.P., Wolter, F. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25-29, 2016. pp. 187–196. AAAI Press (2016), http://www.aaai.org/ocs/index.php/KR/KR16/paper/view/12882
2. Baynam, G., Walters, M., Claes, P., Kung, S., LeSouef, P., Dawkins, H., Bellgard, M., Girdea, M., Brudno, M., Robinson, P., et al.: Phenotyping: targeting genotype's rich cousin for diagnosis. Journal of paediatrics and child health **51**(4), 381–386 (2015)
3. Girdea, M., Dumitriu, S., Fiume, M., Bowdin, S., Boycott, K.M., Chénier, S., Chitayat, D., Faghfoury, H., Meyn, M.S., Ray, P.N., et al.: P heno t ips: Patient phenotyping software for clinical and research use. Human mutation **34**(8), 1057–1065 (2013)
4. Gonçalves, R.S., Tu, S.W., Nyulas, C.I., Tierney, M.J., Musen, M.A.: An ontology-driven tool for structured data acquisition using web forms. Journal of biomedical semantics **8**(1), 26 (2017)
5. Hombach, D., Schwarz, J.M., Knierim, E., Schuelke, M., Seelow, D., Köhler, S.: Phenotero: Annotate as you write. Clinical genetics **95**(2), 287–292 (2019)
6. Jiménez-Ruiz, E., Hovland, D., Slaughter, L., Håndstad, T., Waaler, A.: On improving the phenotype acquisition process using semantic web technology. In:

Paschke, A., Burger, A., Splendiani, A., Marshall, M.S., Romano, P., Presutti, V. (eds.) Proceedings of the 10th International Conference on Semantic Web Applications and Tools for Health Care and Life Sciences (SWAT4LS 2017), Rome, Italy, December 4-7, 2017. CEUR Workshop Proceedings, vol. 2042. CEUR-WS.org (2017), http://ceur-ws.org/Vol-2042/paper9.pdf

7. Köhler, S., Schulz, M.H., Krawitz, P., Bauer, S., Dölken, S., Ott, C.E., Mundlos, C., Horn, D., Mundlos, S., Robinson, P.N.: Clinical diagnostics in human genetics with semantic similarity searches in ontologies. The American Journal of Human Genetics **85**(4), 457–464 (2009)

8. Liu, C., Kury, P., Sampaio, F., Li, Z., Ta, C., Wang, K., Weng, C.: Doc2hpo: a web application for efficient and accurate hpo concept curation. Nucleic acids research (2019)

9. Stenzhorn, H., Weiler, G., Brochhausen, M., Schera, F., Kritsotakis, V., Tsiknakis, M., Kiefer, S., Graf, N.M.: The obtima system-ontology-based managing of clinical trials. In: Medinfo. pp. 1090–1094 (2010)

10. Tchechmedjiev, A., Abdaoui, A., Emonet, V., Melzi, S., Jonnagaddala, J., Jonquet, C.: Enhanced functionalities for annotating and indexing clinical text with the ncbo annotator+. Bioinformatics **34**(11), 1962–1965 (2018)

11. Wright, C.F., FitzPatrick, D.R., Firth, H.V.: Paediatric genomics: diagnosing rare disease in children. Nature Reviews Genetics **19**(5), 253 (2018)