Journal of the American Medical Informatics Association, 0(0), 2018, 1–4
doi: 10.1093/jamia/ocy091
Brief Communication

Brief Communication

Smashing the strict hierarchy: three cases of clinical decision support malfunctions involving carvedilol

Adam Wright,[1,2,3] Aileen P Wright,[1,2] Skye Aaron,[1] and Dean F Sittig[4]

[1]Department of Medicine, Brigham and Women's Hospital, Boston, Massachusetts, USA, [2]Departments of Medicine and Biomedical Informatics, Harvard Medical School, Boston, Massachusetts, USA, [3]Partners eCare, Partners Healthcare, Boston, Massachusetts, USA and [4]Department of Biomedical Informatics, UTHealth – Memorial Hermann Center for Healthcare Quality and Safety, School of Biomedical Informatics, University of Texas Health Science Center at Houston, Houston, Texas, USA

Corresponding Author: Adam Wright, PhD, Brigham and Women's Hospital and Harvard Medical School, 1620 Tremont St., Boston, MA 02115, USA (awright@bwh.harvard.edu)

Received 29 May 2018; Revised 13 June 2018; Editorial Decision 15 June 2018; Accepted 26 June 2018

ABSTRACT

Clinical vocabularies allow for standard representation of clinical concepts, and can also contain knowledge structures, such as hierarchy, that facilitate the creation of maintainable and accurate clinical decision support (CDS). A key architectural feature of clinical hierarchies is how they handle parent-child relationships — specifically whether hierarchies are strict hierarchies (allowing a single parent per concept) or polyhierarchies (allowing multiple parents per concept). These structures handle subsumption relationships (ie, ancestor and descendant relationships) differently. In this paper, we describe three real-world malfunctions of clinical decision support related to incorrect assumptions about subsumption checking for β-blocker, specifically carvedilol, a non-selective β-blocker that also has α-blocker activity. We recommend that 1) CDS implementers should learn about the limitations of terminologies, hierarchies, and classification, 2) CDS implementers should thoroughly test CDS, with a focus on special or unusual cases, 3) CDS implementers should monitor feedback from users, and 4) electronic health record (EHR) and clinical content developers should offer and support polyhierarchical clinical terminologies, especially for medications.

Key words: hierarchies, data models, knowledge management, knowledge representations, clinical decision support, adverse events

BACKROUND

One of the core tasks of biomedical informaticians is representing concepts and knowledge. Key tools to support this task are controlled medical vocabularies, terminologies, and ontologies. These systems allow for the representation of concepts such as medications, diagnoses, or clinical observations. In addition to enumerating concepts, higher-order systems may provide definitions for concepts, or systems for organizing and classifying them.

A classic example of this process is the hierarchical organization of International Classification of Disease (ICD). The Tenth Edition (ICD-10) with clinical modifications (ICD-10-CM) is broken down into 21 chapters, each containing many concepts represented by codes that start with a letter followed by numbers.[1] For example, codes starting with A and B represent infectious and parasitic diseases, while codes starting with C and D represent neoplasms, and so on.

In the landmark 1998 paper "Desiderata for controlled medical vocabularies in the twenty-first century," Cimino lays out 12 qualities of ideal medical vocabularies.[2] One of these qualities is the support for polyhierarchy. In a polyhierarchy, a single concept can be a member of multiple classifications, while in a strict (or mono-) hierarchy, each concept can exist in only a single category. A classic example of this is the ICD-10 code E11.31, "Type 2 diabetes mellitus with unspecified diabetic retinopathy." Because ICD-10 is a strict

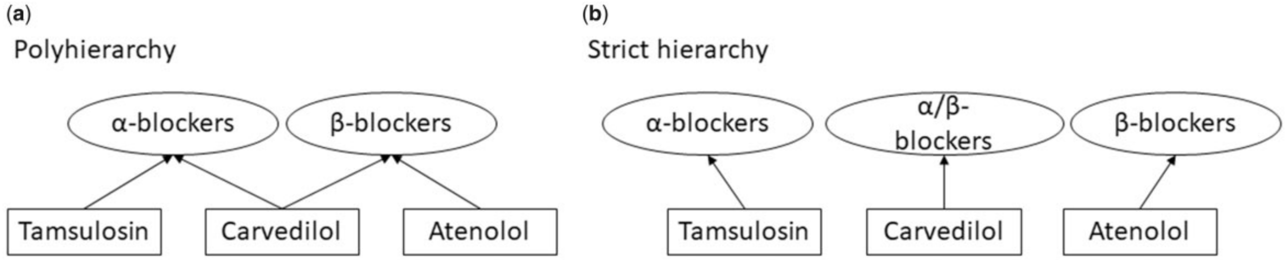

Figure 1. Simplified schematic showing a polyhierarchical drug classification system (panel a) and a strict hierarchy (panel b). In the polyhierarchical system, carvedilol is in both the α-blocker and β-blocker classes, while in the strict hierarchy, each drug can be in only a single class, so it is classified as an α/β-blocker, but is not in the α-blocker or β-blocker class. A real-world classification system would further differentiate between non-selective and selective agents within each class.

hierarchy, this disease falls under chapter E of ICD-10: "endocrine, nutritional and metabolic diseases," but is not listed under chapter H "diseases of the eye and adnexa."

This strict hierarchical limitation imposes certain representational and inferential challenges. For example, if a knowledge engineer sought to identify all patients with eye disease [eg, to support a research study, offer clinical decision support (CDS), or measure quality], it might be tempting to simply locate patients who had a diagnosis in chapter H of ICD-10. However, this would miss patients who had a diagnosis of E11.31, but no other eye disease diagnoses are documented. To surmount this, the knowledge engineer would have to identify additional ICD-10 codes that represented eye diseases outside of chapter H.

SNOMED-CT,[3] by contrast, supports polyhierarchy — a single concept can have multiple parents, making taxonomic inference more straightforward. In the "Desiderata," Cimino observed that, despite the clear benefits of polyhierarchy, "most available standard vocabularies are strict hierarchies."[2] Although the nature of hierarchy in medical vocabularies is often discussed in informatics, there are few concrete reports of clinical issues arising from hierarchical limitations in the informatics literature.

There are distinct features of strict and polyhierarchies, including:

- Subsumption checking (syllogism): It is common to want to ask whether a particular concept is an ancestor or descendant of another concept (either directly or transitively through another concept). This can be done in both strict and polyhieachies, although implementation of subsumption checking may be slightly simpler for strict hierarchies.
- Encoding: In both strict and polyhierarchies, relationships between concepts need to be encoded. In strict hierarchies, this is often done through the identifier (in the ICD-10 example above, E is a parent of E11, which is, in turn, a parent of E11.31). Because each term has only one identifier, using the identifier to encode the parent generally precludes polyhieachy. In polyhierarchies, parent (and other) relationships are encoded through additional attributes or representations.
- Grouping and counting: Certain use cases call for counting the number of objects in a particular class or its descendants (such as the number of patients with diabetes, or taking a beta blocker). In a polyhierarchy, objects may be counted more than once, which can be desirable or undesirable depending on the context.
- Ontologic fidelity: Polyhierarchy allows for more faithful encoding of the real-world relationships between concepts. Strict hierarchies often necessitate compromises, such as assigning a concept to its primary parent, rather than all of its parents.

CASE SERIES

As part of a larger project on CDS malfunctions,[4–12] we have been monitoring academic and industry sources for reports of malfunctions. We have recently observed three separate reports of malfunctions related to the classification of carvedilol. Carvedilol is a non-selective β-blocker, and it also has α-blocker activity. It is approved by the United States Food and Drug Administration (FDA) for the treatment of "mild to severe chronic heart failure, left ventricular dysfunction following myocardial infarction in clinically stable patients and hypertension."[13]

Because it has both α and β-blocker activity, it can be difficult to classify in a strict hierarchy. Figure 1 shows two possible classifications: in the first panel, illustrating a polyhierarchy, its α and β adrenergic antagonisms are both readily represented. In the second panel, illustrating a strict hierarchy, it is not possible for the drug to fall into two classes, so it must be represented either as a β-blocker (its main activity) alone or in a distinct class of α/β-blockers.

The three cases in the literature are as follows:

- A case report by Stone[14] describes a patient who was admitted to the hospital for myocardial infarction and was started on carvedilol. At discharge, a false-positive CDS alert was displayed, which alerted the resident that the patient was not on β-blocker and that one was indicated, given the myocardial infarction. The resident accepted the suggestion, prescribing atenolol, leading the patient to be discharged on two β-blockers. The patient suffered symptomatic bradycardia and hypotension from the therapeutic duplication, and returned to the emergency department.
- A case in a case series by our team[6] of an outpatient CDS alert for patients who have coronary artery disease (CAD) but are not on a β-blocker. The alert fired for patients already on carvedilol (and also labetalol, another α/β-blocker). Although we did not find evidence of any patients who got two beta blockers simultaneously, many clinicians were frustrated by the alert.
- A report by an electronic health record (EHR) vendor[15] about a problem-specific documentation template that shows clinicians whether patients with CAD are taking a β-blocker, but does not recognize carvedilol.

In each of these three cases (which spanned two EHRs and several unrelated clinical sites), the issue arose because the CDS was programmed to look for drugs in a β-blocker class that did not contain carvedilol, either because carvedilol was not a member or descendent of the β-blocker class in the drug vocabulary, or because the EHR did not do subsumption checking. We call these malfunctions subsumption errors, as they are the result of incomplete subsumption checking.

DISCUSSION AND RECOMMENDATIONS

In each of these cases, CDS developers relied on a standard drug classification system. This is actually a best practice,[16] because building CDS with drug classes simplifies knowledge management, particularly as new drugs in a class are released. However, the CDS developers did not realize that the β-blocker class they used did not include carvedilol, leading to real-world clinical issues.

Although all three of these errors involved carvedilol, this problem is certainly not limited to carvedilol. In our collective experience, we have observed subsumption errors related to diagnoses, medications, laboratory results, flowsheet concepts, and even demographics.

To estimate the number of drugs that fall into multiple classes (and thus, could be effected by subsumption errors of this sort), we looked at two drug classification systems that support polyhierarchy: SNOMED and the Established Pharmacologic Class (EPC) system from the FDA. Thirty-nine percent of SNOMED medication concepts (drawn from the FHIR medication value set) had more than one direct parent; however, this may overestimate the frequency, as SNOMED sometimes lists drugs in both functional and chemical classes (eg, SNOMED lists carvedilol as a β-blocker and a propanolamine, but not, interestingly, an α-blocker). EPC is more clinically oriented and lists carvedilol as both an α and β-blocker, exactly as in Figure 1a. Twenty-four percent of drugs in use at Partners HealthCare had multiple EPC classes. There are two common reasons that a drug has multiple classes: first, some drugs contain multiple ingredients with different clinical effects. Second, as in the case of carvedilol, some ingredients have multiple clinical effects. These findings suggest that the phenomenon of drugs being in multiple classes, and thus, at risk for subsumption errors in strict hierarchies, occurs frequently.

The best solution to subsumption errors is careful knowledge management. To this end, we offer four recommendations:

1. CDS implementers should learn about the limitations of tools that they use, such as terminologies, hierarchies, and classification, and should work with pharmacists to carefully analyze third-party clinical content to spot potential gaps in coding schemes, such as the omission of carvedilol from the *β*-blocker class.
2. CDS implementers should thoroughly test CDS, with a focus on special or unusual cases, such as carvedilol. If any of these systems had been tested with carvedilol, the problem could have been found sooner. Of course, identifying and anticipating these special cases can be challenging. We have encountered three common special cases, which we offer as a rule of thumb:
    a. Drugs with multiple effects (such as *α*/*β*-blockers)
    b. Drugs containing multiple ingredients with different effects (such as hydrochlorothiazide-metoprolol)
    c. Drugs in a particular class whose route changes their effect (such as ophthalmic beta blockers)
3. CDS implementers should monitor feedback from users. In the second case described above, many users left comments noting that "the patient is on carvedilol" before the problem was fixed — if these comments had been monitored in real time, the issue could have been corrected sooner.
4. EHR and clinical content developers should offer and support polyhierarchical clinical terminologies, especially for medications. If carvedilol had been in the *β*-blocker classes used by the CDS implementers in these three cases, these issues would have been averted.

Of these recommendations, the fourth is the most complex, and would require development and adoption of new drug terminologies, as well as more sophisticated ontologic inference by the major EHR systems.

CONCLUSION

Using drug classes and other features of terminologies generally allows decision support to be both simpler and more maintainable. However, it can also introduce issues when the expectations of the CDS developer do not match the realities of the terminology. In this paper, we describe three real-world malfunctions of decision support related to failed assumptions about subsumption checking for carvedilol. We recommend that developers of decision support systems, terminologies, and EHR software implement our recommendations to prevent similar events.

FUNDING

The research reported in this publication was supported by the National Library of Medicine of the National Institutes of Health under award number R01LM011966. The content of this article is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

Conflict of interest statement. The authors have no competing interests.

CONTRIBUTORSHIP

Conception and design of the study: AW, DFS;
Acquisition, analysis, and interpretation of data: AW, APW, SA, DFS
Drafting of the manuscript: AW
Critical revision of the manuscript for important intellectual content: AW, APW, SA, DFS
Final approval of the manuscript: AW, APW, SA, DFS

ACKNOWLEDGMENTS

We would like to acknowledge the support of the National Library of Medicine of the National Institutes of Health under award number R01LM011966 for funding our research into CDS malfunctions.

REFERENCES

1. Centers for Disease Control and Prevention. International Classification of Diseases, Tenth Revision, Clinical Modification (ICD-10-CM); 2018. https://www.cdc.gov/nchs/icd/icd10cm.htm. Accessed May 22, 2018.
2. Cimino JJ. Desiderata for controlled medical vocabularies in the twenty-first century. *Methods Inf Med* 1998; 37 (4–5): 394–403.
3. SNOMED International. SNOMED CT The Global Language of Healthcare. https://www.snomed.org/snomed-ct. Accessed May 22, 2018.
4. Ray S, McEvoy DS, Aaron S, *et al*. Using statistical anomaly detection models to find clinical decision support malfunctions. *J Am Med Inform Assoc* 2018; doi:10.1093/jamia/ocy041.
5. Liu S, Wright A, Sittig DF, *et al*. Change-point detection for monitoring clinical decision support systems with a multi-process dynamic linear model. *Proceedings (IEEE Int Conf Bioinformatics Biomed)* 2017; 2017: 569–72.
6. Wright A, Ai A, Ash J, *et al*. Clinical decision support alert malfunctions: analysis and empirically derived taxonomy. *J Am Med Inform Assoc* 2018; 25 (5): 496–506. doi: 10.1093/jamia/ocx106.
7. Wright A, Aaron S, Seger DL, *et al*. Reduced effectiveness of interruptive drug-drug interaction alerts after conversion to a commercial electronic health record. *J Gen Intern Med* 2018; doi:10.1007/s11606-018-4415-9.

8. Liu S, Wright A, Hauskrecht M. Change-point detection method for clinical decision support system rule monitoring. *Artif Intell Med (2017)* 2017; 10259: 126–35.
9. Wright A, Hickman TT, McEvoy D, *et al*. Analysis of clinical decision support system malfunctions: a case series and survey. *J Am Med Inform Assoc* 2016; 23 (6): 1068–76.
10. McEvoy DS, Sittig DF, Hickman TT, *et al*. Variation in high-priority drug-drug interaction alerts across institutions and electronic health records. *J Am Med Inform Assoc* 2017; 24 (2): 331–8.
11. Schreiber R, Sittig DF, Ash J, *et al*. Orders on file but no labs drawn: investigation of machine and human errors caused by an interface idiosyncrasy. *J Am Med Inform Assoc* 2017; 24 (5): 958–63.
12. Wright A, Aaron S, Sittig DF. Testing electronic health records in the "production" environment: an essential step in the journey to a safe and effective health care system. *J Am Med Inform Assoc* 2017; 24 (1): 188–92.
13. GlaxoSmithKline. Highlights of prescribing information coreg® (carvedilol) tablets; 2008. https://www.accessdata.fda.gov/drugsatfda_docs/label/2009/020297s029lbl.pdf. Accessed May 22, 2018.
14. Stone EG. Unintended adverse consequences of a clinical decision support system: two cases. *J Am Med Inform Assoc* 2018; 25 (5): 564–7.
15. NextGen Healthcare Information Systems. Carvedilol is not recognized as beta blocker therapy - LVSD on the HPI – CAD template; 2018. http://pages.nextgen.com/yToSfH4f0k0xTCa0oi00000. Accessed May 22, 2018.
16. Osheroff JA, Teich JM, Levick D, *et al*. *Improving Outcomes with Clinical Decision Support: An Implementer's Guide*. Chicago, IL: Healthcare Information and Management Systems Society; 2012.