Received: 27 October 2020 | Revised: 14 January 2021 | Accepted: 2 February 2021

DOI: 10.1111/cts.13021

BRIEF REPORT

Application of MCAT questions as a testing tool and evaluation metric for knowledge graph–based reasoning systems

Karamarie Fecho[1,*] | James Balhoff[1] | Chris Bizon[1] | William E. Byrd[2] | Sui Hang[3] | David Koslicki[4] | Stefano E. Rensi[5] | Patrick L. Schmitt[1] | Mathias J. Wawer[6] | Mark Williams[7] | Stanley C. Ahalt[1]

[1]Renaissance Computing Institute, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina, USA

[2]University of Alabama at Birmingham, Birmingham, Alabama, USA

[3]Institute for Systems Biology, Seattle, Washington, USA

[4]Penn State University, University Park, Pennsylvania, USA

[5]Stanford University, Stanford, California, USA

[6]Broad Institute, Cambridge, Massachusetts, USA

[7]National Center for Advancing Translational Sciences, Bethesda, Maryland, USA

Correspondence
Karamarie Fecho, Renaissance Computing Institute, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA.
Email: kfecho@copperlineprofessionalsolutions.com

Funding information
Support for this work was provided by the intramural research program within the National Center for Advancing Translational Sciences (NCATS), National Institutes of Health, and the NCATS Biomedical Data Translator program (awards OT3TR002026, OT3TR002020, OT3TR002025,

Abstract

"Knowledge graphs" (KGs) have become a common approach for representing biomedical knowledge. In a KG, multiple biomedical data sets can be linked together as a graph representation, with nodes representing entities, such as "chemical substance" or "genes," and edges representing predicates, such as "causes" or "treats." Reasoning and inference algorithms can then be applied to the KG and used to generate new knowledge. We developed three KG-based question-answering systems as part of the Biomedical Data Translator program. These systems are typically tested and evaluated using traditional software engineering tools and approaches. In this study, we explored a team-based approach to test and evaluate the prototype "Translator Reasoners" through the application of Medical College Admission Test (MCAT) questions. Specifically, we describe three "hackathons," in which the developers of each of the three systems worked together with a moderator to determine whether the applications could be used to solve MCAT questions. The results demonstrate progressive improvement in system performance, with 0% (0/5) correct answers during the first hackathon, 75% (3/4) correct during the second hackathon, and 100% (5/5) correct during the final hackathon. We discuss the technical and sociologic lessons learned and conclude that MCAT questions can be applied successfully in the context of moderated hackathons to test and evaluate prototype KG-based question-answering systems, identify gaps in current capabilities, and improve performance. Finally, we highlight several published clinical and translational science applications of the Translator Reasoners.

Study Highlights

WHAT IS THE CURRENT KNOWLEDGE ON THE TOPIC?

Knowledge graphs (KGs) are common approaches for representing biomedical knowledge. KGs are typically tested and evaluated from a software engineering perspective.

[*]Apart from the first/lead and last/senior authors, all other authors are listed alphabetically.

OT2TR002517, OT2TR002514, OT2TR002515, OT2TR002584, and OT2TR002520). Any opinions expressed in this document are those of the Translator community writ large and do not necessarily reflect the views of NCATS, individual Translator team members, or affiliated organizations and institutions.

WHAT QUESTION DID THIS STUDY ADDRESS?

We explored a team-based approach to evaluate three prototype KG-based question-answering systems through the application of Medical College Admission Test (MCAT) questions within a moderated "hackathon" setting.

WHAT DOES THIS STUDY ADD TO OUR KNOWLEDGE?

We demonstrate that MCAT questions can be applied successfully in the context of moderated hackathons to test and evaluate KG-based question-answering systems, identify gaps, and improve performance.

HOW MIGHT THIS CHANGE CLINICAL PHARMACOLOGY OR TRANSLATIONAL SCIENCE?

We expect that our approach will have broad application in biomedical software development efforts such as ours.

INTRODUCTION

The Biomedical Data Translator ("Translator") program, funded by the National Center for Advancing Translational Sciences within the National Institutes of Health, was launched in 2016. The program aims to radically change the way that translational research is conducted and increase the speed of discovery through the development of an open informatics platform capable of integrating disparate biomedical data sets and reasoning over them to generate mechanistic insights into disease and ultimately advance clinical care.[1–3]

"Knowledge graphs" (KGs) have become a common approach for knowledge representation in numerous scientific fields, including biomedicine.[4] The Translator program has adopted the use of KGs as a method for biomedical data integration, reasoning, and new knowledge generation. We have developed several Translator "Reasoners" or KG-based question-answering systems as part of the Translator program. Systems such as ours are typically tested and evaluated through the application of tools and approaches drawn from the field of software engineering. In this study, we explore a team-based approach to apply Medical College Admission Test (MCAT) questions as a tool for testing and evaluating the prototype Translator Reasoners and identifying gaps in data sources and software capabilities. We describe the results of three "hackathons,"[5] in which software developers worked together with a moderator to solve MCAT questions using the Translator Reasoners. We report an incremental improvement in system performance and document the lessons learned through our approach.

METHODS

Motivation

The idea for the work described herein was sparked during a presentation in Spring 2018 by a lead developer of IBM Watson Health.[6] The presentation focused on the use of United States Medical Licensing Examination (USMLE) questions to "train" Watson Health, and it described how the approach led to small, but steady improvements in question-answering over time.

Given that USMLE questions are largely narrative-based and are not readily amenable to software implementation without significant manual human processing or natural language processing (NLP), we considered MCAT questions as an alternative. Like USMLE questions, MCAT questions are multiple choice and have one correct answer plus three incorrect answers each, and both are extremely well-vetted by experts and thus are capable of serving as "ground-truth" answers for prototype systems, such as the Translator Reasoners. Moreover, the incorrect answers are intentionally "close" to the correct answer and therefore can provide a benchmark for sensitivity in any predicted answers. Yet, MCAT questions are more simplistic and structured than USMLE questions. Given these considerations, we decided to apply MCAT questions as a testing tool and evaluation metric.

Translator reasoner ecosystem

We focused on three Translator Reasoner applications: ROBOKOP (Reasoning Over Biomedical Objects linked in Knowledge Oriented Pathways)[7–9], RTX (Reasoning Tool X)[10]. and mediKanren.[11] Each application uses the Biolink model[12,13] as an upper-level ontology to express domain knowledge as a KG of relationships between biomedical entities, with nodes representing biomedical entities, such as "gene," "biological pathway," or "chemical substance," and edges representing predicates or relationships between nodes, such as "causes," "increases expression of," or "treats." A variety of biomedical data sources, some highly curated and others less so, are used to populate the Translator Reasoner KGs; these include DrugBank, Comparative Toxicogenomics Database, PubChem, ChEMBL, Chemical Entities of Biological Interest, Monarch, Monarch Disease Ontology, Human Phenotype Ontology, Gene Ontology, ClinGen,

ClinVar, UniProt Knowledgebase, PharmGKB, SemMedDB, and dozens more.[7–11] Reasoning or inference algorithms are applied to the underlying KG, thus allowing users to ask questions, such as: Why is imatinib effective in the treatment of asthma? What genes contribute to disease severity among patients infected with COVID-19? Why does Niemann–Pick C1 disease confer resistance to Ebola? Of note, ROBOKOP, RTX, and mediKanren differ in the biomedical data sources and ontologies that they draw from, as well as the reasoning and inference algorithms that are invoked to answer questions, thus allowing the three applications to collectively serve as a powerful biomedical KG-based ecosystem.

MCAT questions

MCAT questions were freely available from Khan Academy.[14] Three sets of five questions each (15 total questions) were selected by the hackathon moderator (K.F.) prior to the first of three 4-h hackathons (Appendix S1). The questions were semi-randomly selected, in that questions that were known to exceed current Translator capabilities (e.g., temporal sequences of events) were intentionally excluded. In addition, the questions were chosen to have broad representation in terms of types of biological entities and fields of study (e.g., molecular biology, cellular anatomy, and pathophysiology). All questions were compiled without any editing.

Hackathons

The three 4-h hackathons took place in January 2019, July 2019, and September 2019, with an additional 1-h one-on-one follow-up meeting in November 2019. Two of the hackathons were a combination of in-person and remote, and one was in-person only. The follow-up meeting involved the moderator and the lead developer of mediKanren (W.B.) who was unable to join the September 2019 meeting; the other two hackathons included the lead developers of each of the three Translator Reasoners plus other Translator team members.

The hackathons were moderated and intentionally restricted to a small number of team members (5 to 10 team members each) in order to ensure that the events were focused and productive. The participants included the lead developers of the three Translator Reasoners and contributors to those systems or related Translator systems. The hackathon participants had diverse backgrounds, with expertise in software engineering, ontologies, semantic web technologies, analytics, data science, bioinformatics, cheminformatics, and computational biology, and all but one (P.L.S., an undergraduate student) held advanced degrees in these and related fields. Apart from the moderator (K.F.) and a co-moderator for one of the three hackathons (S.H.), none of the hackathon participants had direct experience in clinical or translational science. In contrast, both the moderator and the co-moderator did have direct experience in clinical and translational science, and both held advanced degrees in biomedical fields.

The moderator initiated each hackathon, guided the activities, helped troubleshoot, and recorded extensive notes. The other participants served as "hackers" and actively tested the Translator Reasoners. The composition of each hackathon varied, in terms of participants, although at least one representative for each Translator Reasoner was present at all three events. Other than the participant composition and the fact that one of the hackathons was in-person only, the events were structured identically, although the time devoted to each question varied (data not recorded).

Outcome measures

The primary outcome measure was the number and percentage of MCAT questions that were answered correctly during each hackathon. An answer was deemed to be correct if at least one Translator Reasoner was capable of generating a correct answer by one of the following methods: direct match with the multiple-choice text answer listed as correct for a given MCAT question; process of elimination by ruling out incorrect choices for an MCAT question; or partial match with the multiple-choice text answer listed as correct for a given MCAT question.

The secondary outcome measure was a catalog of lessons learned from each hackathon.

RESULTS

Over the course of three hackathons, we demonstrated improved performance in answering MCAT questions using the prototype Translator Reasoners, with 0 of 5 (0%) questions successfully answered in January 2019, 3 of 4 (75%) in July 2019, and 5 of 5 (100%) in September 2019, albeit with one correct answer from the September 2019 hackathon not confirmed until November 2019 during a one-on-one meeting with the lead developer of mediKanren and the moderator (Table 1).

We did not find an obvious pattern when comparing the performance of each of the three Translator Reasoners. In the second hackathon, correct answers were identified by RTX for one question, ROBOKOP and mediKanren for a second question, and all three Reasoners for the third question. In the third hackathon, correct answers were identified by ROBOKOP for one question, mediKanren for a second question, RTX for a third question, mediKanren and ROBOKOP for a fourth question, and all three Reasoners for the fifth question.

Numerous issues and lessons learned were acquired as a result of the hackathon exercise (Table 1). Some of the issues were anticipated a priori. For example, we found missing or

TABLE 1 Translator performance and lessons learned when applying Translator Reasoners to answer MCAT questions over three 4-h moderated hackathons

Hackathon date	Success rate	Lessons learned
January 2019	0/5 questions (0%)[a]	• Missing/incomplete data sources • Errors with existing data sources • Inadequate specificity with existing data sources • Entity identifier mismatches • "One-hop" graph queries insufficient
July 2019	3/4 questions (75%)	• Missing/incomplete relationships between entities • Limited or absent annotation for certain data sources • Lack of relative/contextual relationships • "Opposites" under-represented or absent in data sources • "Synonymization" or equivalence of text terms challenging • Lack of differentiation or unclear differentiation between data types (e.g., disease vs. phenotype, protein vs. gene) • Multiple implementation strategies (e.g., direct match, process of elimination, and inference) improves success rate
September 2019	5/5 questions (100%)[a]	• "Two-hop" graph queries and other more complex queries more effective than "one-hop" queries • Query directionality and choice of predicate important • Missing or incomplete predicates • Terminology challenges with pluralities • Exact matches to correct answers uncommon • Generalization and inference required for terms that lack specificity • Careful review of supporting evidence improves success rate • Biomedical input facilitates developer identification of correct answer

Abbreviation: MCAT, Medical College Admission Test.

[a]The goal was to tackle five questions for this hackathon session, but only four questions were attempted due to time constraints.

[b]The correct answer to one of the five questions was confirmed during a subsequent November 2019 meeting with the moderator and the lead developer of one of the Translator Reasoners who was unable to attend the September 2019 hackathon.

incomplete data sources, errors with existing data sources, and entity identifier mismatches. Other issues and lessons learned were unexpected. For instance, we found "opposites," such as "spasticity" versus "rigidity," "hypertension" versus "hypotension," and "leukopenia" versus "leukocytosis" to represent a major challenge, due in part to a lack of available data sources and the fact that biomedical ontologies do not typically capture such relations. Additional unexpected terminology challenges related to pluralities (e.g., "increased microglia" vs. "gliosis"), specificity (e.g., "glial cell" vs. "microglia cell"), and differentiation between data types (e.g., "protein" vs. "gene"). Moreover, the text terms that were used in the questions often did not have an obvious equivalent or synonym in the data sources used to populate the Translator KGs (e.g., "adrenaline" vs. "epinephrine), which made it challenging for developers with little to no biomedical background to identify the correct match. In such cases, the moderator, who had a biomedical background, was often able to provide guidance, primarily by translating questions or defining terminologies. Apart from terminology challenges, many questions which, at first glance, appeared to require a simple "one-hop" query (e.g., chemical X interacts with protein Z) in order to identify an answer, actually required more complex "two-hop" queries (e.g., chemical X is associated with gene Y whose product is protein Z) or completely different query strategies. Likewise, we found

that careful review of supporting evidence (i.e., metadata and publications) often led to the identification of a correct answer even when the queries themselves did not return an exact match. Finally, we found it beneficial to invoke multiple strategies (e.g., direct match, process of elimination, and inference) when attempting to answer a question, instead of relying on one approach. This was especially true for questions involving negatives (e.g., Which of the following is NOT innervated by the autonomic nervous system?).

DISCUSSION

We demonstrate improvements in the performance of three Translator Reasoners over three 4-h moderated hackathons, using the ability to identify correct answers to MCAT questions as an evaluation metric. Our findings complement those of researchers at IBM Watson Health,[6] who used USMLE questions to train the prototype Watson Health question-answering system. Indeed, IBM's experience motivated the current work. One difference between IBM's work and that presented here is that we did not use NLP to translate questions, whereas the IBM team incorporated NLP to train their system. This difference reflects differences in the goals of the two projects, with Watson Health focused on clinical decision support, and Translator focused on augmenting (not replacing) human reasoning by providing mechanistic insights into biomedical observations.[1–3] The differing goals also explain the choice in evaluation tools, with the Watson Health team choosing USMLE questions and our team choosing MCAT questions. Nonetheless, the two efforts both resulted in slow and steady improvements in the performance of the prototype question-answering systems through the use of questions designed to test medical students (MCAT questions, Translator) or medical residents (USMLE questions, Watson Health).

A secondary goal of the current work was to catalog the critical issues and lessons learned during each hackathon. This proved to be a useful exercise, as we identified issues that were both anticipated and unexpected. Anticipated issues were largely related to data quality, with missing data sources, identifier mismatches, and errors in existing data sources. Many of these issues were subsequently addressed by team members or reported to the data owners, thereby contributing to the progressive improvements that we observed. Unexpected issues revealed gaps with the prototype Translator system. Many of these issues were related to terminology challenges and mismatches between the terminology used in the MCAT questions and that used in the data sources and ontologies used to populate the Translator Reasoner KGs. The terminology challenges fell largely into five categories: opposites, pluralities, specificities, differentiation, and equivalence. Another unexpected challenge was that the MCAT questions themselves were more challenging than originally anticipated. Indeed, multiple strategies were required to answer the questions, much like the many strategies that humans apply to answer MCAT questions.

Although the work described in this manuscript focused on the use of MCAT questions as a tool to evaluate and improve the Translator Reasoners, we note that the Reasoners themselves have broad application in clinical and translational science. Specifically, these systems are being used to propose mechanistic insights into clinical and translational observations. For instance, we are using the Translator Reasoners to generate testable hypotheses regarding genes and biological pathways that might causally explain real-world associations between chemical workplace exposures and immune-mediated diseases, such as asthma, celiac disease, and multiple sclerosis.[15] We are also applying the Reasoners to suggest drug targets for rare diseases, such as Fanconi anemia and cyclic vomiting syndrome.[7,16] In addition, the Translator Reasoners are being applied to delineate clinical outcome pathways and adverse outcome pathways.[8] We anticipate many additional applications as the Reasoners mature and their usability improves, with improved interfaces and documentation to support their use by clinical and translational scientists who may have limited technical skills. We encourage interested users to contact the developers for assistance with the tools.

In addition to the technical benefits of using MCAT questions as testing and evaluation tools for KG-based reasoning systems and their application in clinical and translational science, our experience supports the use of hackathons as a means to promote goal-oriented multiteam collaboration and productive team science. We found this to be true with both the two mixed remote/in-person hackathons and the one in-person hackathon that we held as part of the work described here. Our prior experience likewise supports the use of hackathons to promote software development and foster a collaborative team culture.[2,17] In all cases, we found that hackathons are most productive and successful when they involve small, focused groups and are guided by a moderator. In the hackathons described herein, the moderator had a biomedical background, which likely contributed to the success of the events and the progressive performance improvements in Translator Reasoners. We acknowledge that hackathons are not always viewed favorably,[18,19] but our experience[2,17] and that of others[5] suggests that if carefully planned and implemented, hackathons can provide a highly productive environment to support both software development and team science.

One limitation of the hackathon exercise is that it is difficult to distinguish between improvements related to data sources versus those related to the software engineers themselves. For instance, as we identified gaps related to the data sources that were used to populate the KGs and/or the way that the Translator Reasoners treated those data sources, we worked to address them, so data quality necessarily improved over time. However, at the same time, the software engineers may have simply become more

familiar with the MCAT questions over each hackathon, thus resulting in performance improvements that were independent of improvements in data quality and/or software capabilities. Regardless, our experience suggests that MCAT questions can be used to evaluate KG-based software applications, such as the Translator Reasoners, and that moderated hackathons can facilitate the process. Indeed, we did not identify any obvious patterns in the performance of the three Translator Reasoners or the ability of one system to perform more effectively than the others, thus supporting the idea that our approach can be applied more generally to KG-based reasoning systems.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the intellectual input, useful discussions, and general camaraderie provided by the Biomedical Data Translator Consortium. We also thank Translator leadership, namely, Drs. Christopher P. Austin, Christine M. Colvis, and Noel T. Southall.

CONFLICT OF INTEREST

The authors declared no competing interests for this work.

AUTHOR CONTRIBUTIONS

K.F. wrote the manuscript and designed the research. K.F., J.B., C.B., W.B., S.H., D.K., S.E.R., P.L.S., M.J.W., M.W., and S.C.A. performed the research and analyzed the data.

DATA AVAILABILITY STATEMENT

An instance of the ROBOKOP UI for exploration of the ROBOKOP KG can be found at http://robokop.renci.org; the ROBOKOP KG can be accessed and downloaded at http://robokopkg.renci.org. Code and instructions for ROBOKOP and the ROBOKOP KG are available under the MIT open software license at https://github.com/NCATS-Gamma/robokop. The RTX GitHub repository can be found at https://github.com/RTXteam/RTX; the RTX KG UI can be found at https://arax.ncats.io/. The mediKanren GitHub repository can be found at https://github.com/webyrd/mediKanren.

REFERENCES

1. Austin CP, Colvis CM, Southall NT. Deconstructing the translational tower of babel. *Clin Transl Sci*. 2019;12(2):85.
2. The Biomedical Data Translator Consortium. The biomedical data translator program: conception, culture, and community. *Clin Transl Sci*. 2019;12(2):91-94.
3. The Biomedical Data Translator Consortium. Toward a universal biomedical data translator. *Clin Transl Sci*. 2019;12(2):86-90.
4. Nicholson DN, Greene CS. Constructing knowledge graphs and their biomedical applications. *Comput Struct Biotechnol J*. 2020;18:1414.
5. Tauberer J. How to run a successful hackathon. 2017. https://hackathon.guide/. Accessed August 21, 2020.
6. Prager JM. From question answering to clinical decision support: Electronic medical record analysis activities in the Watson lab. Presentation, April 11, 2018, Renaissance Computing Institute, University of North Carolina, UNC, Chapel Hill, North Carolina. [Also see IBM Watson Health; https://www.ibm.com/watson-health.] https://apps.research.unc.edu/events/index.cfm?event=events.eventDetails&event_key=5590D9117267E6B9FEE286722A3CDB6474EEB580. Accessed March 23, 2021.
7. Bizon C, Cox S, Balhoff J, et al. ROBOKOP KG and KGB: integrated knowledge graphs from federated sources. *J Chem Inf Model*. 2019;59(12):4968-4973.
8. Morton K, Wang P, Bizon C, et al. ROBOKOP: an abstraction layer and user interface for knowledge graphs to support question answering. *Bioinformatics*. 2019;35(24):5382-5384.
9. ROBOKOP. http://robokop.renci.org. http://robokopkg.renci.org; https://github.com/NCATS-Gamma/robokop. Accessed February 24, 2021.
10. RTX. https://github.com/RTXteam/RTX. Accessed February 24, 2021.
11. mediKanren. https://github.com/webyrd/mediKanren. Accessed February 24, 2021.
12. Biolink model, undated. https://github.com/biolink/biolink-model. Accessed February 24, 2021.
13. Mungall CM. Standardized biological knowledge graphs: the Biolink model. https://www.slideshare.net/cmungall/introduction-to-the-biolink-datamodel. Accessed April 13, 2018.
14. Khan Academy. MCAT test prep. https://www.khanacademy.org/test-prep/mcat. Accessed February 24, 2021.
15. Fecho K, Bizon C, Miller FW et al. Use of the open ROBOKOP knowledge graph–based application to provide mechanistic explanations for observed associations between environmental exposures and immune-mediated diseases. AMIA 2020 Annual Symposium, November 2020, conference paper. https://knowledge.amia.org/72332-amia-1.4602255/t005-1.4604904?qr=1.
16. Shefchek KA, Harris NL, Gargano M, et al. The Monarch Initiative in 2019: an integrative data and analytic platform connecting phenotypes to genotypes across species. *Nucleic Acids Res*. 2020;48(D1):D704-D715.
17. Fecho K, Ahalt SC, Arunachalam S, et al. Sex, obesity, diabetes, and exposure to particulate matter: scientific insights revealed by analysis of open clinical data sources during a five-day hackathon. *J Biomed Inform*. 2019;100:103325.
18. Siva V. Are hackathons good, bad, or overrated? *hackerearth blog*. https://www.hackerearth.com/blog/innovation-management/hackathons/good-bad-overrated/. Accessed June 6, 2018.
19. Swanner N. Is it time to rethink the hackathon? Dice, https://insights.dice.com/2018/02/21/time-rethink-hackathon/. Accessed February 21, 2018.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section.

How to cite this article: Fecho K, Balhoff J, Bizon C, et al. Application of MCAT questions as a testing tool and evaluation metric for knowledge graph–based reasoning systems. *Clin Transl Sci*. 2021;00:1–6. https://doi.org/10.1111/cts.13021