

Course Recommender > Report Outline

Course Recommender > Report Outline	1
Abstract	1
Introduction/Motivation	2
Use Case	2
Technical Approach	2
Related Work	3
Evaluations	3
Discussion	4
Future Work	5
References	5

Abstract

Abstract - provide a draft abstract - no less than 4 sentences and include 1-3 claims. Abstracts should include a short problem description, short description of why it is important, and some short description of your team contribution. Abstracts should aim to be a half page, and never be more than one page in length.

Students attending a university can benefit from carefully planning out which courses they will take during the course of their academic career. Developing such a plan of study can be difficult because students need to consider factors such as how each course applies to their graduation requirements, whether course schedules have conflicts, whether the courses are interesting to the student, and how registering for particular courses can affect their choices of future courses. In this work, we present a Course Recommendation Ontology, which aims to represent relevant information about courses, schedules, graduation requirements, and student interests to enable a recommender system. Our ontology can be used to enable rule-based recommendations that adhere to course requirements (such as prerequisite requirements) and provide explanations to students about why certain courses are recommended to them.

Introduction/Motivation

Few sentences.

The domain of courses themselves are relatively clear with concrete credit count, course level and subjects. However, when entered into the domain of course registration and graduation requirements, more un-described context of the domain is required to make decisions. Given every semester an average of around 5000 students all need to simultaneously perform the same action of comparing available courses with their own graduation requirements, there is a need for guidance. This guidance at the moment is serviced by academic advisors that are the

subject matter experts that provide this reconciliation of courses to requirements. To alleviate this enormous constraint on these subject matter experts (SMEs) and empower students with the appropriate data, an Ontology can be used to describe the domain and offer a query interface to expose not only data and requirements, but also the less tangible recommendations that advisors often provide.

Use Case

Summary and Pointer to online use case.

The goal of the Course Recommender Ontology is to create a set of recommended courses for an RPI student to enroll in for their next upcoming semester and make the registration process a little less stressful. The Course Recommender Ontology captures three important components of academic planning: graduation requirements, course availability, and student interests in order to complete this goal. The ontology then leverages the hierarchical structure of coursework and from this foundation builds rules/constraints in order to create useful and individualized recommendations. Specifically, for the scope of our project this semester, the majors we are covering from the course catalog are Computer Science (CSCI) and Information Technology and Web Science (ITWS). Additionally, our project focuses solely on undergraduate courses and degree requirements since these aspects are very different when a student is attending graduate school. The stakeholders for this project include students and academic advisors since the ontology both empowers students to understand their academic track as well as promote more effective discourse between an advisor and their advisee. For more information about the Course Recommender Ontology see our [full use case document](#).

Technical Approach

Whatever structure makes sense for your app here. Make sure to include a description of your conceptual model with a link to your online version and also a link to your ontology.

- Describe how we represent students and their plan-of-study.

Students have a few attributes that represent them: their interests, class year, and the courses they take. For their courses, students have a role “hasStudyPlan” that points towards a “Plan of Study” instance. The “Plan of Study” then has multiple links under the roles “hasCompletedCourse”, “hasPlannedCourse”, and “hasOngoingCourse”, which dictate which courses sections a student has taken or plan to take. The relevant object, a “Course Section”, has the necessary information which semester the course occurs on, which enables us to divide the student’s plan of study into different semesters.

- Describe how we encode courses, course sections, and scheduled course sections. Relate them to what is captured in the plan-of-study, and how we use course codes and department codes.

A major modelling decision that was made is the distinction of courses and course sections. In this case, a “Course” is the category of a class you register for. While small details may change over time, the general framework and topics of the course should remain the same. In addition, courses do not change often with regards to the overall prerequisite directed graph. A “Course

Section” is a semester-specific instance of a class that you register for. Within this concept, there are further distinctions between a “Scheduled Course Section” and a “Course Section”, where the scheduled instance is one that has been given an explicit registration number by the administration, essentially a course that is known to occur in the following semester. The benefit of this approach is that the system can estimate when a course is likely to be scheduled in the future and allow students to preemptively plan to take that course without waiting for the administration to release a list of available courses. Additionally, this allows us to treat a “Course” and “Course Section” as data, which improves the reusability of the ontology for other universities and avoids complex modelling patterns, such as punning.

- Describe how we represent graduation requirements. We only handle representing the basics of requirements in the ontology, so an external app would be needed to actually deal with how a student’s completed courses are applied to their requirements.

Modelling requirements for graduation requires breaking down each major and sub requirement. This is because a graduation requirement often requires a total number of credits to be taken as well as a number of credits from courses within a specific subset. To represent these, a major is linked to multiple requirements through the “hasRequirement” role. A requirement is then further broken down into subcomponents by “isFulfilledBy” and “hasSubRequirement”. “isFulfilledBy” enables representing the different paths a student can take to satisfy a requirement. For example, it enables representation of needing to take two courses within the same section. Likewise, “hasSubRequirement” enables composition of requirements.

- Relate each of the components to how they would be used in a recommender system. Requirements would be involved with providing explanations, applying constraints in the system, and filtering courses. Course sections and scheduled course sections would be used to provide insight into actual course offerings for students to sign up for. Student interests would be used to help rank recommended courses (based on interest). Plan-of-study could be used to infer interests as well as constrain the search space of future courses to sign up for.

Related Work

You can fill this in later.

:^)

Evaluations

Point to competency question now

The following competency questions can be used to assess the ability of our ontology to answer questions that would be essential to solve our key use case of recommending courses to students.

For each of the competency questions we can also include more information and explanations based on the content we have included in our use case.

1. What are all the prerequisite courses needed to take the CSCI 4340 Ontologies course?
 - a. This competency question demonstrates that our ontology captures course prerequisite information, which limits a student's ability to sign up for certain courses.
2. I am a rising senior and I want to take the smallest number of courses required to complete my degree. I also want to take "easier" courses whenever possible to allow for more time to plan for a future career. What courses can fulfill my remaining requirements? (This question will use the course history of Jacob Shomstein to produce an answer, but a full list of courses completed is omitted for privacy)
 - a. This competency question addresses problems related to graduation requirements, handling what courses have already been completed, and some subjective notions like "easiness" of courses.
3. I have taken CSCI 4340 Ontologies and CSCI 4020 Design and Analysis of Algorithms. What are some courses like CSCI 4340 Ontologies that I should take next fall?
 - a. This competency question addresses the ability of the ontology to represent schedules of courses (e.g., Fall 2020) and inferring completed courses based on the provided information (CSCI 4340 has prerequisites, so if a student has taken that course we can assume they have the background involved in its prerequisite courses).
4. I have taken CSCI 2300. What are some other courses that I can register for, and why are they recommended to me?
 - a. This competency question once again touches on the representation of prerequisites. It also involves some aspect of preference and explanations - explanations may include pointing out that a recommended course can be used to fulfill graduation requirements, or that a course has a topic that you have indicated as being interested in.
5. How will signing up for CSCI 4350, ITWS 4350, and EARTH 4350 affect my progress towards graduation?
 - a. This competency question mostly related to being able to capture cross-listing information for courses. Multiple cross-listed courses cannot count towards graduation, but they may be relevant for factors like prerequisites.

Discussion

\1 Value of Semantics - make sure to have a separate section in the discussion highlighting the impact of the semantics. You should include a paragraph in this draft outline or up to 3 bullet points. This section is one of the most important in your report

\2 Include a link to your project website

\3 Support your claims

The impact of the semantics is fairly powerful. At the moment, the domain knowledge of which majors require what requirements that link to what courses are obscure mentioned in hard-to-find websites, or held with subject matter experts. This prevents systems to be

responsive to semantic change of requirements and students from having all of the data in one place. By clearly defining the semantics of the domains through clear definition of entities and relationships between the requirements, host institution, courses and properties we not only allow students to get clear answers to the question of “what courses should I enroll in next semester” but also allowing systems like DegreeWorks, SIS and 3rd party applications like YACS to work together and stay responsive to data and semantic changes. For instance, DegreeWorks during COVID was unable to describe and handle the COVID exception that would allow students to take any course Pass/No Credit and have that course apply to graduation requirements if the student passed. This semantic addition of a COVID exception was hard to describe and manage, hence systems like DegreeWorks had to rely on advisors to manually provide comments that would still fully integrate with the system.

Some particular benefits that we gain from this semantics-based approach are

- Inference of interests and completed courses (e.g., you have taken many courses that cover sub-topics of A.I., so you probably are interested in A.I.)
- Distinguishing related concepts based on the ontology’s class hierarchy (e.g., we can filter potential courses to recommend, then filter those courses based on courseSections that are offered in specific semesters, then further filter those using scheduledCourseSections that are concrete instances of a particular course with a schedule, instructor, and registration number)
- Reasoning to fill in incomplete information (e.g., cross-listed courses often don’t show a complete list of cross-listings in the course catalog. Since cross-listing is a transitive and reflexive property, we can use reasoning to expand our available knowledge base)

[Our website](#) describes more about the described conceptual model, use case, queries and further discussion.

Future Work

Future work entails expanding the supported majors beyond those supported in our initial scope (CSCI, ITWS), expanding our coverage of graduation requirements, and opening up the application for use outside of the host institution RPI. To do so, we would want to publicize the requirements through JSON-LD or an alternative shared schema store to allow schools to have a standardized way of describing graduation requirements. This would allow maximum interoperability between the ontology and other systems. We also hope to create and test an implementation to actually recommend courses based on the content in the ontology.

References

[1] <https://schema.org/>

[2] <https://dublincore.org/>

- [3] <http://xmlns.com/foaf/spec/>
- [4] <https://vocab.org/participation/schema>
- [5] https://www.dublincore.org/specifications/lrmi/lrmi_1/
- [6] <http://smiy.sourceforge.net/olo/spec/orderedlistontology.html>
- [7] <https://vocab.org/aiiso-roles/schema>
- [8] Xiaowen Huang National Laboratory of Pattern Recognition, et al. "Explainable Interaction-Driven User Modeling over Knowledge Graph for Sequential Recommendation." Explainable Interaction-Driven User Modeling over Knowledge Graph for Sequential Recommendation | Proceedings of the 27th ACM International Conference on Multimedia, 1 Oct. 2019, dl.acm.org/doi/10.1145/3343031.3350893.
- [9] Zhang, Yongfeng and X. Chen. "Explainable Recommendation: A Survey and New Perspectives." ArXiv abs/1804.11192, 2020.
- [10] Rensselaer Polytechnic Academic Calendar - <https://info.rpi.edu/registrar/academic-calendar>
- [11] "Rensselaer Polytechnic Institute - Acalog ACMS™", *Rensselaer Catalog 2019-2020*, 2019.
[Online]. Available: <http://catalog.rpi.edu/>. [Accessed: 28- Sep- 2020]
- [12] Self-Service Information System. 2020. *User Login*. [online] Available at: <https://sis.rpi.edu/> [Accessed 27 September 2020].