Content-based Course Recommender System for Liberal Arts Education

Raphaël Morsomme
University College Maastricht
Zwingelput 4
6211 KH Maastricht
raphael.morsomme@maastrichtuniversity.nl

Sofia Vazquez Alferez
University College Maastricht
Zwingelput 4
6211 KH Maastricht
sofia.vazquezalferez@maastrichtuniversity.nl

ABSTRACT

Liberal Arts programs are often characterized by their open curriculum. Yet, the abundance of courses available and the highly personalized curriculum are often overwhelming for students who must select courses relevant to their academic interests and suitable to their academic background. This paper presents the course recommender system that we have developed for the Liberal Arts bachelor of the University College Maastricht, the Netherlands. It aims to complement academic advising and help students make better-informed course selections. The system recommends courses whose content best matches the student's academic interests, issues warnings for courses that are too advanced given the student's academic background and, in the latter case, suggests suitable preparatory courses. We base the course recommendations on a topic model fitted on course descriptions, and the warnings on a sparse predictive model for grade based on students' past academic performance and level of academic expertise. Preparatory courses consist of courses whose content has the best preparatory value according to the predictive model. We find that course recommendations are relevant for a wide range of academic interests present in the student population and that students found recommendations for courses at other departments especially helpful. The preparatory courses often lack coherence with the target course and need to be improved.

Keywords

Education, recommender system, warning, topic model, grade prediction.

1. INTRODUCTION

The Bachelor in Liberal Arts offered at the University College Maastricht, the Netherlands, is an honors program characterized by an open curriculum. The program allows students to design their curriculum in a fairly free fashion: more than 75% of the educational credits are free, the college offers over 150 courses covering a wide range of topics from artificial intelligence, to conflict resolution and to pop songs, and students can take up to one year's worth of courses outside of the college. This freedom allows students to tailor their curriculum to their own interests; but the abundance of courses available makes the selection of courses overwhelming. First, the number of courses offered at the 12 departments of the university is too large for students to have an overview of which ones match their academic interests. Second, since each liberal arts student has a unique curriculum, it can be difficult for them to determine if they have covered the necessary prerequisites for a particular course or if the course's level is too advanced given their academic background. A recommender system that suggests courses whose content matches students' academic interests, issues a warning for courses too advanced and, in the latter case, provides suitable preparatory courses would therefore be extremely beneficial. Not only would it increase the students' information position, thereby improving self-advising, but it would also improve academic advising when used as an agenda-setting tool.

Our course recommender system achieves these three goals: course suggestion, warning issuance and preparatory course advice. To receive course suggestions, the student enters her/his academic interests into the system which returns the 20 courses whose content best matches them. In practice, the student selects key words from a predetermined list that represent her/his academic interests. The course recommender system then uses a topic model to identify the courses whose content best matches the topics corresponding to the selected key words (see Figure 1). To receive warnings, students provide their transcript and indicate which courses they are considering for the following term. The system issues a warning for courses that it identifies as too advanced given the student's academic background. In practice, the student enters her/his student ID with which the system extracts her/his past academic performance and the expertise that she/he has acquired in various topics. From these, the system uses a predictive model to estimate the grade that the student will obtain in the selected courses and issues a warning when the predicted grade is a fail (see Figure 2). Each warning issued is then accompanied by a list of preparatory courses whose content has the best preparatory value according to the predictive model.

2. RELATED WORK

Identifying courses that are both of interest to the students and of an appropriate level is a task that has recently gained attention in the literature. Gulzar et al. [8] propose a recommender system that uses information retrieval techniques to select courses based on students' interests. Their system uses key words to search the space of possible courses and tries to improve the quality of the query by finding synonyms and generating N-grams so that the search returns a higher number of courses. Then, they use an ontological model to expand the search even further and retrieve courses related to the previously extracted courses in the ontological model. In this context, an ontological model is a knowledge model that represents relationships between concepts of a previously specified domain, such as 'Computer Science' [7]. This system is content-based since it is the contents of the courses

Raphael Morsomme and Sofia Vazquez Alferez "Content-based Course Recommender System for Liberal Arts Education" In: *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019)*, Collin F. Lynch, Agathe Merceron, Michel Desmarais, & Roger Nkambou (eds.) 2019, pp. 748 - 753

that are matched to the concepts of the ontological model or the key words of the query. In this manner, the recommender system allows the interest of the students to be matched to the contents of the course. However, the system suffers from several drawbacks: first, the domains (e.g. Computer Science or Medicine) from which the ontological models are built must be defined a-priori [7]. Second, the recommender system is dependent on a well-built database that is not always available at interested institutions.

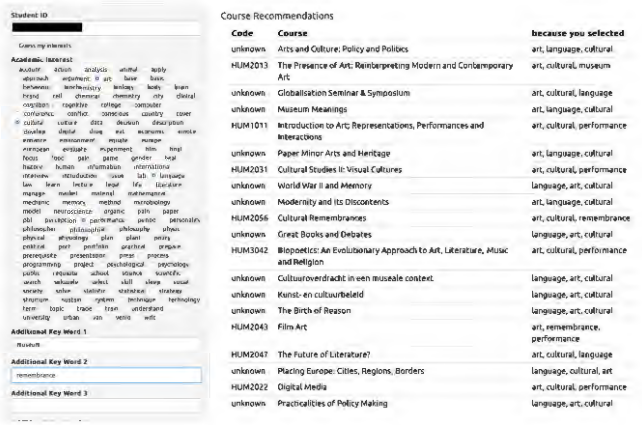

Figure 1. Course suggestions.

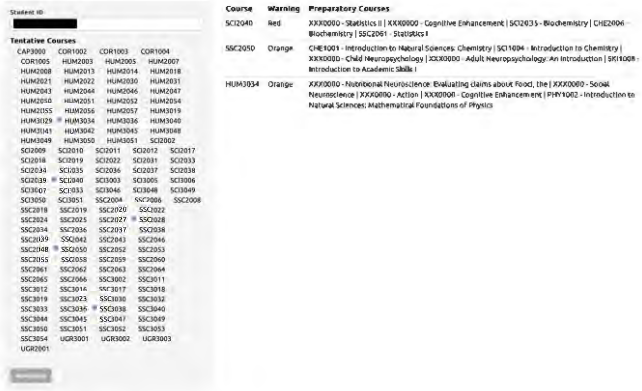

Figure 2. Warnings and preparatory courses.

Bydžovská [3] develops a recommender system that takes into account a student's past performance and interest profile to make course recommendations. Students' interests are defined in a narrow sense, that is, a course is considered of interest if a student has taken the course or marked it as a favorite in the university system. Course recommendations based on interest are then issued via a collaborative filtering approach: the suggested courses are the courses most selected by other students in the same field of study, or those that were taken by the n-most similar students that already graduated. To detect risk of failure, Bydžovská [3] predicts grades using classification and regression, or nearest neighbor, depending on the course. Warnings are issued after binning the predicted grades into excellent, good, or bad. The main innovation of the system is that it proceeds to include social behavior and take into account courses taught by a favorite teacher or taken by similar students. Although the system attempts to handle both interest and appropriateness of a course's level, it suffers from three major disadvantages: first, it does not provide the kind of transparent recommendation that would allow students to reflect on their course selection because the content of the course is not explicitly taken into account. Second, it does not give students suggestions on how to address their deficiencies. Third, it does not permit students to change their interest, which is particularly important in a liberal arts context where students go through a broad exploratory phase before specializing.

Bakhshinategh et al. [1] address the issue of recommending courses that help students overcome their deficiencies whilst accounting for changes over time. They view a study program as a path to obtain graduating attributes (skills, qualities, understandings) and rank the impact that each course has on promoting those graduating attributes for a student who took the course. The ranking is done through self-assessment by students after completing the course. The recommender system then uses collaborative filtering to find courses that score highest on promoting a targeted graduating attribute for a student who wishes to develop it further. Thus, if a student lacks "analytical skills", the system identifies courses that improve these skills so that a student comes closer to the level of "analytical skills" that is required for graduation. This system can be used to find preparatory courses for other courses by shifting from graduating attributes to attributes required to succeed in a course. The main disadvantage is that the impact of each course is found through self-assessment rather than in a data-driven way.

Jiang et al. [11] take a different approach to find preparatory courses by using recurrent neural networks to develop a goal-based course recommender. A student specifies a course that they wish to take, along with the grade that they desire to achieve, and the system uses their transcript to find personalized preparatory courses. Although this approach finds preparatory courses in a data-driven way, it does so at the expense of transparency, which makes a student's reflective decision-making process more difficult and provides no direct insight for academic advising.

We use a topic model to extend Bydžovská's [3] use of students' interest. This provides a more flexible and realistic interpretation of a student's interests and how they change over time. Moreover, we use a topic model to expand the search of relevant courses in the manner that by Gulzar et al. [8] use ontological models. The advantage of a topic model is that topics are learned from the data and must not be known in advance. Our system also supplements recommendations with explanations and additional information to help students make well-informed course selections.

3. DATA

We use two types of data: student data and course data.

The student data consists of anonymized course enrollment information. We use the transcripts of the 2,526 students of the liberal arts program between 2008 and 2019 with a total of 79,245 course enrollments. We exclude enrollments with a missing grade which indicates that the student either dropped the course or fail the attendance requirement. In the latter case, the data set contains an observation corresponding to the resit. Table 1 presents the student data. Each row contains an anonymized student ID, a course ID, a year and semester, and the obtained grade.

The course data consists of the 2018-2019 course catalogues of 5 departments of Maastricht University: European Studies, University College Maastricht, University College Venlo, Psychology and Science Program. These catalogues contain a one-page description of 490 courses. Table 2 presents the textual data in the tidy format with one row per document-term [18]. We process the data following common cleaning procedures [13]: we tokenize the individual terms, stem them with the Hunspell dictionary and remove common stop words, numbers between 1 and 1,000, and terms occurring less than 3 times in the data set.

Table 1. Example of student data

Student ID	Course ID	Academic Year	Period	Grade
44940	CAP3000	2009-2010	4	8.8
37490	SSC2037	2009-2010	4	8.4
71216	HUM1003	2010-2011	4	6.8
44212	SSC2049	2010-2011	2	8.4
85930	SSC2043	2011-2012	1	4.3
14492	COR1004	2012-2013	2	8.5
34750	HUM2049	2013-2014	5	6.0
32316	SSC1001	2013-2014	1	8.5
22092	SCI1009	2014-2015	1	6.4
19512	COR1004	2016-2017	5	7.0

Table 2. Example of course data

Course ID	Course Title	Department	word
HUM3034	World History	UCM	understand
HUM3034	World History	UCM	major
HUM3034	World History	UCM	issue
HUM3034	World History	UCM	episode
HUM3034	World History	UCM	shape
HUM3034	World History	UCM	history
HUM3034	World History	UCM	mankind
HUM3034	World History	UCM	focus
HUM3034	World History	UCM	theme
HUM3034	World History	UCM	topic

4. METHODOLOGY

4.1 Overview

Figure 3 presents a diagram of the course recommender system. We start by fitting models to the data. We use the Latent Dirichlet Allocation statistical model to fit a topic model on the course data and the lasso penalty to fit a series of sparse multiple linear regression models for grade prediction to the student data. A model is fitted to each course. Their inputs consist of the students' past academic performance and level of expertise in the topics which we derive from their transcript with the topic model.

These models generate intermediate results from the user's input. We use the topic model to infer the student's academic interests from the key words that she/he has entered into the system and the regression models to predict the grades that the student will obtain in the course she/he selected based on her/his transcript.

The system's outputs are based on these intermediate results. The course suggestions consist of the 20 courses whose content best matches the student's academic interest in terms of Kullback-Leibler distance. Warnings are issued when the predicted grade is a fail. For each warning issued, we indicate the 5 courses whose content has the best preparatory value according to the regression model. The preparatory value of a course is estimated with the Kullback-Leibler distance of its topic distribution to the coefficient estimates of the topic variables in the linear regression.

All computations are realized on the environment for statistical computing and graphics *R* [16, 13, 10, 4, 19, 20].

4.2 Topic Model

We use the Latent Dirichlet Allocation (LDA) generative probabilistic model and the Gibbs sampling algorithm to fit a topic model to the course data.

The LDA model conceptualizes topics as a probability distribution over a finite set of words (in this case, the vocabulary of the course data), and a document (i.e. a course description) as a sequence of *N* words, where each word was generated by drawing from a probability distribution over topics specific to that document [2]. Thus, each word belongs to all topics but with different probabilities, and all topics are present in each course but with different weights. Figure 4 and Figure 5 respectively show the word distribution in two topics and the topic distribution in a course based estimated by the topic model fitted on the course data. Technically, the LDA model generates a document as follows. First, the word distribution β for each topic is determined by $\beta \sim \text{Dirichlet}(\delta)$ and the topic weights θ for each document are determined by $\theta \sim \text{Dirichlet}(\alpha)$. Second, each of the N words of the document is chosen by choosing a topic $z \sim \text{Multinomial}(\theta)$ and then choosing a word from a multinomial probability distribution conditioned on the topic *z*.

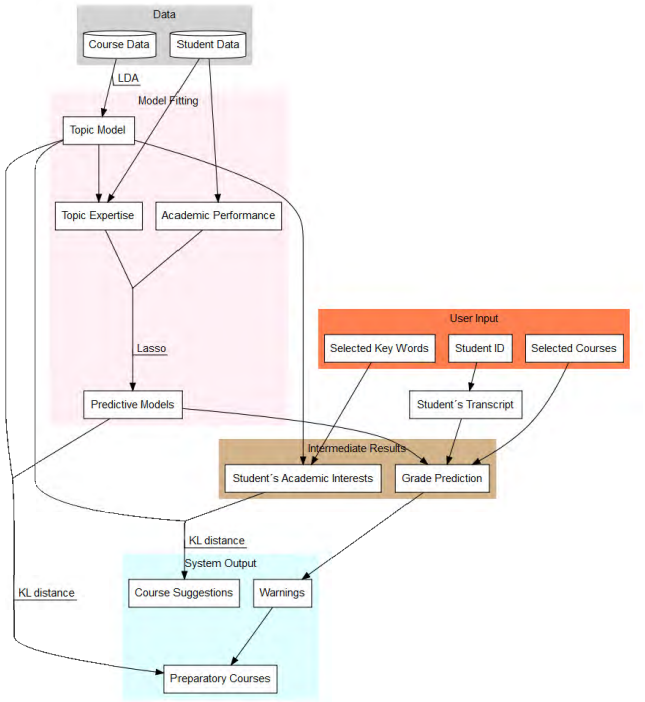

Figure 3. Diagram of the course recommender system.

Gibbs sampling is a Monte Carlo Markov Chain (MCMC) technique for successively sampling conditional distributions of variables whose distribution over states converges to the true distribution in the long run [5]. Gibbs sampling generates posterior samples by sweeping through each variable and sampling from its conditional distribution when the other variables are fixed to their current values. Phan et al. [14] used Gibbs sampling to learn the distributions β and θ for the LDA model. In this case, δ and α are the prior distributions for Gibbs sampling, acting as hyper-parameters that respectively determine how sparse the distributions of words in topics and topics in documents are. Gibbs sampling picks each word in the vocabulary and estimates the probability of assigning the current word to each topic conditioned on the topic assignments of all other words. With this conditional distribution, given a document, a topic is sampled and assigned as the new topic assignment for the current word. Then, with the distribution of words per topic, we compute the conditional probability of the topics given an observed

document. Since Gibbs sampling is a MCMC, the distribution sampled from a large number of iterations approximates the target distribution [5], enabling us to infer β and θ.

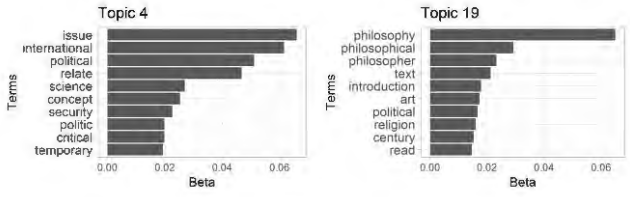

Figure 4. Term distribution in two topics. Topic 4 corresponds to international politics and topic 19 to philosophy.

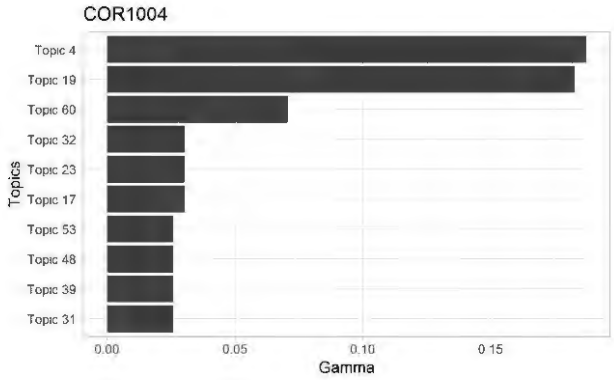

Figure 5. Topic distribution in the core course COR1004 Political Philosophy. The course is characterized by topic 4 (international politics) and topic 19 (philosophy).

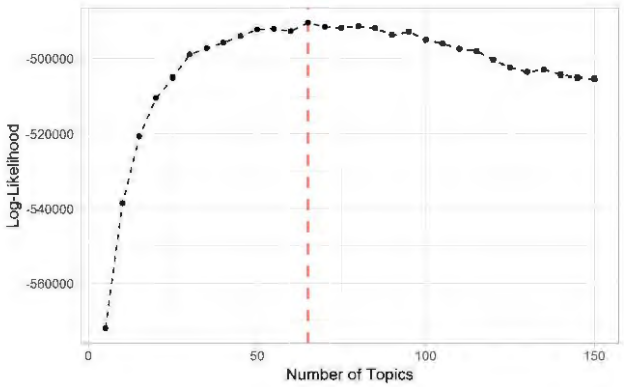

Figure 6. Model selection based on log-likelihood. The maximum-likelihood model has 65 topics.

4.2.1 Model Selection

This procedure requires that we fix *a-priori* the number of topics (*k*) to be inferred. We trained 30 models with $k = 5, 10, \ldots, 150$. We set α to $50/k$ and δ to 0.1 as suggested by Griffiths & Steyvers [6]. For Gibbs sampling, we run 6,000 iterations with a burn in of 1,000 iterations and sample every 100 iterations. To avoid being stuck in a local optimum, we use 10 random initializations to explore the model space and keep the best model with respect to the log-likelihood. We then select the number of topics yielding the model with the largest log-likelihood [6]. Figure 6 shows the log-likelihood of the topic models that we trained. The model with 65 topics has the maximum log likelihood. In order to increase the quality of the selected model, we refit it with more iterations (16,000 iterations with a burn in of 2,000 iterations and 20 random starts; the other parameters are kept the same).

4.3 Warnings

We fit a sparse multiple linear regression model for grade prediction to each of the 132 courses currently offered at the college that have had more than 20 student enrollments since 2008. We regularize the models with the lasso penalty [17]. The set of predictors consists of students' past academic performance and their level of topic expertise at the start of the course. Students' past academic performance consists of 6 variables corresponding to their general and concentration-specific GPA's (humanities, natural sciences, social sciences, skills and projects). Students' topic expertise consists of a set of 65 variables (one per topic of the topic model) which indicate how much knowledge the student has acquired about the topic during her/his studies. A topic expertise variable corresponds to the sum of the topic's importance in the courses taken by the student (as estimated by the topic model) weighted by the grades. We assume that students who obtain 10/10 acquired all the topic-related knowledge present in the course while those obtaining 5/10 acquired half of it. Tables 3a, 3b and 3c and Figure 7 show a toy example of the contribution of individual courses towards a student's topic expertise.

Since the number of predictors is large, we regularize the models with the lasso penalty to increase their accuracy. The lasso penalty shrinks the coefficient estimates of the model, thereby reducing its variance. For each model, we use 10-fold cross-validation (CV) to find the lasso tuning parameter λ that minimizes the CV mean absolute error, a more robust loss function than the squared error [9]. Figure 8 presents the distribution of the CV mean absolute error for the 132 prediction models. The model for the course *PRO2004 Academic Debate* has the smallest prediction error (0.38 grade point) and the model for *SCI3006 Mathematical Modelling* the largest (1.80 grade point). The mean CV error weighted by the number of students enrolled in the course is 0.78, the median is 0.78 and the standard deviation is 0.28.

To receive a warning, the user enters into the system her/his student ID and the list of courses that she/he is considering for the coming term. The system uses the student ID to extract her/his transcript, from which her/his past academic performance and topic expertise are determined. We then use the regression models to predict the grades that the student will obtain in the selected courses and issue a warning for the fail grades (see Figure 2).

4.3.1 Rule-based Warnings

We initially explored an alternative approach for warnings based on association rules. We used the SPADE algorithm [21] to identify sequences in the students transcripts of the type <fail course A> => <fail course B> or <not take course A> => <fail course B> and considered sequences with a support superior to 10 students, a confidence superior to 0.4 and a lift superior to 1.1. Warnings were issued when a student selected a course for which one of the selected rules predicted a failure.

The transparency of this approach motivated its initial adoption; but it turned out to be unsuitable to our case. First, given the small size of the student data and the fact that relatively few students fail courses at the college, only 21 rules met the criteria. Second, this approach ignores the fact that skills necessary to perform well in a particular course can be acquired by taking a *combination* of courses. To tackle the first issue, we considered a relaxed version of the rules that substitutes a <fail course A> with a <obtain less than 6.5 in course A>. The number of rules meeting the relaxed criteria increased to 185. Yet, the second issue remained and led us to consider regression models that use topic expertise as a proxy for the skills necessary to perform well in a course.

Table 3a. Toy example: topic distribution in 3 courses

Course	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Course 1	0.0	0.7	0.2	0.1	0.0
Course 2	0.2	0.2	0.2	0.2	0.2
Course 3	0.0	0.4	0.2	0.1	0.2

Table 3b. Toy example: transcript

Course	Grade
Course 1	6/10
Course 2	9/10
Course 3	2.5/10

Table 3c. Toy example: course contribution to topic expertise

Course	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Course 1	0.00	0.42	0.12	0.060	0.00
Course 2	0.18	0.18	0.18	0.180	0.18
Course 3	0.00	0.10	0.05	0.025	0.05

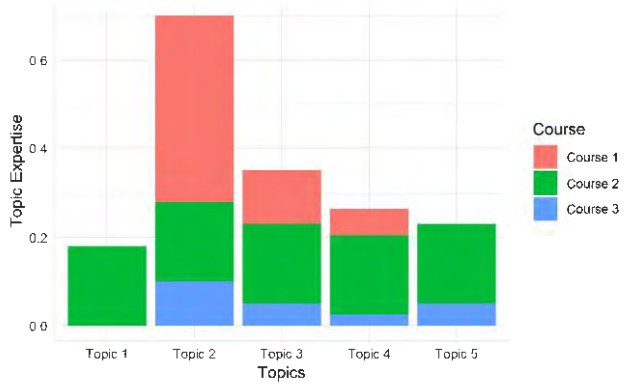

Figure 7. Toy example: course contribution to a student's topic expertise. We use these variables to predict grade.

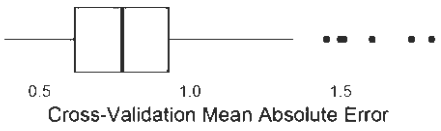

Figure 8. Distribution of the cross-validation mean absolute error in the 148 predictive models.

4.4 Course Recommendation

To provide course recommendations, we identify courses whose content best matches the academic interests of the students. We use the Kullback-Leibler distance, an asymmetric measure of the difference between two probability distributions [12], to estimate the degree to which a course's topic distribution (as estimated in the topic model) corresponds to the normalized academic interests of the student. The system returns the 20 courses with the smallest such distance. A student's academic interests profile consists of a numeric vector indicating the interest of the student in each of the topics. It corresponds to the sum of the selected key words' contribution to the topics in the topic model. In order to assist students in selecting key words, we preselect the 10 terms most relevant to their topic expertise profile (as defined in section 4.3). To make the system as informative and transparent as possible, each recommendation includes the three selected key words with the most relevance to the course. Here, a term's *relevance* corresponds to the sum of the term's contribution across the topics weighted by the importance of the topics in the student's academic interests profile or student's topic expertise profile.

4.5 Preparatory Courses

In order to help students plan their curriculum, each warning is accompanied by a list of suitable preparatory courses. Similarly to the regression models built for the warnings, we fit a lasso-regularized multiple linear regression model for grade prediction to each course; but this time, the input only consists of the students' topic expertise. A positive coefficient estimate indicates that more knowledge of the topic is associated with larger grade in the course. For each course, the preparatory courses consist of the 5 courses (excluding advanced courses) whose topic distribution has the smallest KL distance to the course's regression's normalized coefficient estimates.

5. RESULTS

We used expert validation to evaluate the system's usefulness: current students, alumni and members of academic advising interacted with the system and commented on their experience.

We find that the system recommends course that are potentially useful to the students, thereby helping them make better-informed course selections. First, students value the system's ability to consider multiple interpretations of the same term e.g. the term *function* in mathematics and in biology. This feature of the system stems from the possibility for a term to have a large weight in several topics. Second, many users were surprised that they do not need to enter key words present in the course description for the course to be recommended. Since topics act as a buffer between the key words entered in the system and the course descriptions, students merely need to choose key words that characterize topics present in the course. This allows them to focus on their academic interests when selecting key words as opposed to thinking about the courses that might interest them. Third, they found that self-selected key words yield recommendations that are more useful than those stemming from the preselected key words. This pattern is due to the presence of topics related not to the content but the structure of the courses. For instance, topic 25 is dominated by the terms *paper*, *write* and *assessment*. A student's topic expertise profile therefore contains topics related to the *structure* of the courses that they have taken, which, for a student focusing on film art, leads to the preselection of the terms *research*, *method*, *period*, and *skill* along with *film*, *gender*, *literature* and *culture*, and the recommendation of the courses *Research Methods: Interviewing*, *Research skills*, and *Research Project*. Excluding structured-related key words solves the issue and results in suggestions of potentially interesting courses: *Narrative Media*, *Pop songs and poetry*, and *Cultural Studies II*. We therefore include an *opt-out* option to cancel key word preselection. Fourth, students found recommendations for courses at other departments particularly useful: in most cases, they ignored that these courses existed or that their content matched their academic interests.

Students wished that warnings also included low grades. We therefore provide *red* warnings for predicted fail grades (< 5.5/10) and *orange* warnings for low ones (between 5.6/10 and 6.5/10).

Users were enthusiastic about the preparatory courses; they found it very beneficial to receive suggestions of how to prepare for a particular course. Unfortunately, the preparatory courses returned by the system often lack coherence with the target course. For instance, the list of preparatory courses for the course *World*

History contains the course *Nutritional Neuroscience*. These incongruences may stem from the presence of structure-related topics in the topic model combined with the fact that the lasso penalty shrinks most coefficient estimates to 0. Hence, it is possible that the regression model for grade prediction of some courses only has non-zero coefficient estimates for structure-related topics, hence yielding preparatory courses characterized by these structure-related topics, and not the content-related topics.

6. FUTURE WORK

This course recommender system is a work in progress and the difficulties detailed above indicate three pathways for future work.

First, we need to differentiate structure-related and content-related topics. This seems particularly difficult to do. One approach is to manually inspect the topics that are most prevalent in the corpus of documents and reduce the weight of the structure-related ones.

Second, in order to increase the coherence between preparatory courses and target course, we could impose that their content must be related. The KL distance could be used to accomplish this. We could also take the personalized approach of Jiang et al. [11].

Third, since the topic model has a central place in the system, we plan to improve it by (i) expanding the course data to course manuals (20-page document offering a detailed description of a course's content) and the material covered in the course e.g. academic articles, textbook chapters, and (ii) using a structural topic model that uses covariates to build the model and calibrate topic prevalence and topic content depending on metadata [15] to take into account the origins (department) of the course data.

7. ACKNOWLEDGMENTS

Our thanks to the University College Maastricht, Maastricht University, the Institute of Data Science, and the Department of Data Science and Knowledge Engineering, in particular to Evgueni Smirnov for his technical support and Peter Vermeer for initiating the project and enabling collaboration with the University College Maastricht.

8. REFERENCES

[1] Bakhshinategh, B., Spanakis, G., Zaïane, O. R., & ElAtia, S. (2017). A Course Recommender System based on Graduating Attributes. In CSEDU (1) (pp. 347-354).

[2] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*(Jan), 993-1022.

[3] Bydžovská, H. (2016). Course Enrollment Recommender System. *International Educational Data Mining Society*.

[4] Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.

[5] Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

[6] Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences*, *101*(suppl 1), 5228-5235.

[7] Gulzar, Z., & Leema, A. A. (2016). An ontology based approach for exploring knowledge in networking domain. In *2016 International Conference on Inventive Computation Technologies (ICICT)* (Vol. 1, pp. 1-6). IEEE.

[8] Gulzar, Z., Leema, A. A., & Deepak, G. (2018). PCRS: Personalized course recommender system based on hybrid approach. *Procedia Computer Science*, *125*, 518-524.

[9] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*, Springer Series in Statistics.

[10] Hornik, K., & Grün, B. (2011). topicmodels: An R package for fitting topic models. *Journal of Statistical Software*, *40*(13), 1-30.

[11] Jiang, W., Pardos, Z. A., & Wei, Q. (2019, March). Goal-based Course Recommendation. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*(pp. 36-45). ACM.

[12] Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, *22*, 79-86.

[13] Meyer, D., Hornik, K., & Feinerer, I. (2008). Text mining infrastructure in R. *Journal of statistical software*, *25*(5), 1-54. URL http://www.jstatsoft.org/v25/i05/.

[14] Phan, X. H., Nguyen, L. M., & Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web* (pp. 91-100). ACM.

[15] Roberts, M. E., Stewart, B. M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S. K. & Rand, D. G. (2014). Structural topic models for open-ended survey response. *American Journal of Political Science*, *58*(4), 1064-1082.

[16] Team, R. C. (2013). A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing. *URL http://www. R-project. org*.

[17] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267-288.

[18] Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, 59(10), 1-23. doi:10.18637/jss.v059.i10.

[19] Wickham, H., Francois, R., Henry, L., & Müller, K. (2015). dplyr: A grammar of data manipulation.

[20] Wickham, H. (2016). *ggplot2: elegant graphics for data analysis*. Springer.

[21] Zaki, M. J. (2001). SPADE: Efficient algorithm for mining frequent sequences. *Machine learning*, *42*(1-2), 31-60.