Course Recommender System in a Liberal Arts Context

Raphaël Morsomme
University College Maastricht
Zwingelput 4
6211 KH Maastricht
+32496486716
raphael.morsomme@maastrichtuniversity.nl

Sofia Vazquez Alferez
University College Maastricht
Zwingelput 4
6211 KH Maastricht
+31625246406
sofia.vazquezalferez@maastrichtuniversity.nl

ABSTRACT

This paper describes a direct application of topic modelling and sequential rule mining to provide transparent course recommendations to students of the Liberal Arts and Sciences Bachelor from University College Maastricht, based on their academic interests and performance in previous courses. The system is developed to complement academic advising and help students make well informed decisions. We find that course recommendations based on a topic modeling of course descriptions are useful and that sequence mining provides a rough method to control for prerequisites.

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 at other departments of the university. This freedom allows students to tailor their curriculum to their own interests, but the large number of courses available makes the selection of courses overwhelming (Surpatean et al, 2012). Firstly, 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. Secondly, since each student of the program has a unique curriculum, it can 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 system that identifies courses matching students' academic interests and issues a warning for courses too advanced would therefore be extremely beneficial for the students of the Liberal Arts program. Not only does it increase their information position, thereby improving self-advising, but, used as an agenda-setting tool, it also improve academic advising.

Our course recommender system achieves both goals: courses suggestions and warning issuance. To receive course suggestions, the student enters her/his academic interest into the system which returns the 10 courses that best match the student's interests. 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 X). 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 current academic background of the student. In practice, the student enters her/his student ID with which the system extract 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 X).

Furthermore, in order to help the students plan their curriculum in a well-informed way, each course suggestion includes a list of the selected key words that led to it, and each warning issued has a list of preparatory coursework attached to it.

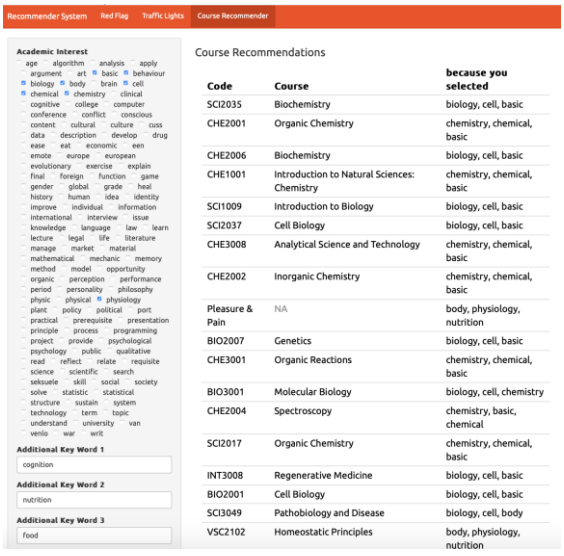

Figure X. Course suggestions in practice.

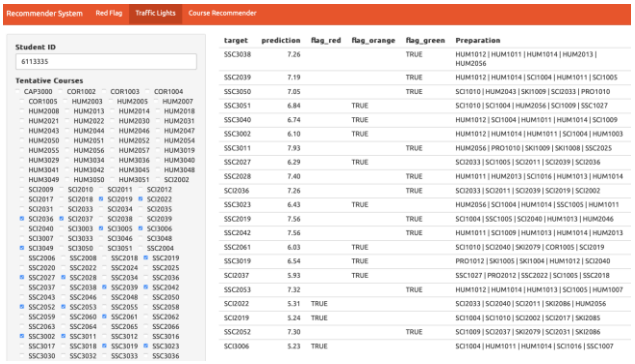

Figure X. Warnings in practice.

2. PREVIOUS WORK

Identifying courses that are both of interest to the (university) students and of an appropriate level is a task that has recently gained attention in the literature. Gulzar, Leema and Deepak (2018) proposed a recommender system that uses information retrieval techniques to select courses based on student interests. Their system uses key words to search the space of possible courses but 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, an Ontological Model is used to expand the search even further and retrieve courses that are related in the Ontological Model to the previously extracted courses. In this context, an Ontological Model is a knowledge model that represents relationships between concepts of a previously specified domain, such as 'Computer Science' (Gulzar, Leema, 2016). The system is considered to be content based because it is the contents of the courses 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 (Gulzar, Leema, 2016). Second, the recommender system is dependent on a well-built database that is not always available at interested institutions.

Bydžovská (2016), developed 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 made through a collaborative filtering approach: the suggested courses were the most selected courses by other students in the same field of study, or those that were taken by the n-most similar students that had already graduated. To detect risk of failure, Bydžovská (2016) predicted grades of students using classification and regression, or nearest neighbor depending on the course, binned the predicted grades into excellent, good, or bad and then issued warnings accordingly. The main innovation of the system, was that it proceeded to include social behavior and consider courses taught by a favorite teacher or taken by friends of students into the recommendations. Although the system attempts to handle both interest and appropriateness of level for a course, it suffers from a three major disadvantages: firstly, 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. Secondly, it does not give students suggestions of how to address their deficiencies. Thirdly, it does allow for a change in student interests, which is particularly important in a liberal arts context where students go through a broad exploratory phase before specializing.

Bakhshinategh, Spanakis et. al, (2017) addressed the issue of recommending courses that helps 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 taking 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.

Jang, Pardos and Wei (2019) 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 they desire to achieve and the system uses their personal course enrollment history and grades 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 to academic advising on how to improve the curriculum.

[INSERT LINK TO OUR WORK FITS HERE]

- we extend Bydžovská (2016) 's use of student interest by using a topic model.

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, indicating 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. Figure X 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 course catalogues contains a one-page description of 490 courses. Figure X presents the textual data in the tidy format with one row per document-term (Wickham, 2014). We follow common data cleaning procedure in text mining (Meyer, Hornik, & Feinerer, 2008): 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 X presents a flowchart of our course recommender system. At the heart the system is a topic model fitted on the course data with the Latent Dirichlet Allocation algorithm. The topic model is used for three purposes. First, we use it together with the student data to estimate the topic expertise of each student, i.e. how much they know about a particular topic. We then fit a predictive model for each course that takes as input the student's academic performance and topic expertise to estimate her/his grade. The system issues a warning if the student selects a course for which the predictive model predicts a fail grade. Second, we us the topic model to suggest to the student courses whose content match her/his academic interest. The student's academic interests are estimated through the key words that she/he enters into the app and the courses that she/he has taken. The system returns the 10 courses whose topic distribution (as estimated by the topic model) has the shortest KL distance to the academic interests of the student. Third, we use the topic model together with the predictive models to provide a list of preparatory courses accompanying each warning. The list of preparatory courses consists of the 5 courses whose topic distribution has the shortest KL distance to the coefficient estimates of the topic expertise variables in the predictive model.

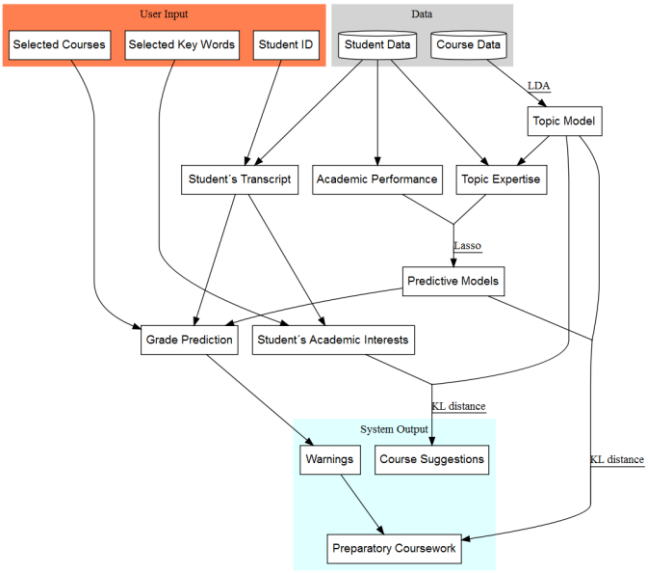

Figure X. Flowchart.

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.

LDA 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. Thus, each word belongs to all topics but with different probabilities, and all topics are present in each course but with different weights. In technical terms, 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 is chosen but choosing a topic $z \sim \text{Multinomial}(\theta)$ and then choosing a word from a multinomial probability distribution conditioned on the topic z.

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 (Gelman et al., 2013). 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. If the documents in our course data were generated with the LDA model, we use Gibbs sampling to learn the distributions β and θ (Phan et al., 2008). In this case, δ and α are the prior distributions for Gibbs sampling, acting as hyper-parameters that respectively affect 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 sampled distribution

approximates the target distribution if the number of iterations is large (Gelman et al., 2013), enabling us to infer β and θ.

This procedure requires that we fix *a-priori* the number of topics (*k*) to be inferred. We trained 30 models with a number of topics ranging from 5 to 150 by 5. We set α to 50/*k* and δ to 0.1 as suggested by Griffiths & Steyvers (2004). For Gibbs sampling, we run 6,000 iterations with a burn in of 1,000 iterations, and sample every 100 iterations. To explore the model space, we use random initializations and the best model over all runs with respect to the log-likelihood is returned. We select the number of topics yielding the model with the largest log likelihood (Griffiths & Steyvers, 2004). The model with 65 topics has the maximum log likelihood. In order to obtain an even more accurate model, we refit it with more iterations (16,000 iterations with a burn in of 2,000 iterations; the other parameters are kept the same).

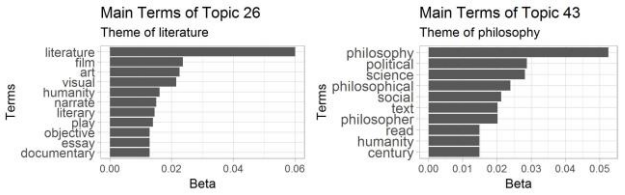

Figure X. Term distribution in two topics.

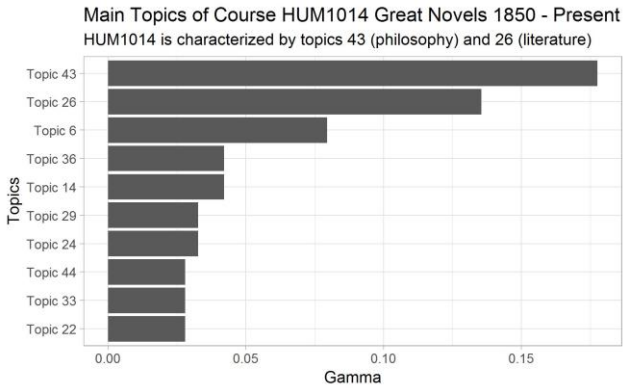

Figure X. Topic distribution in a course.

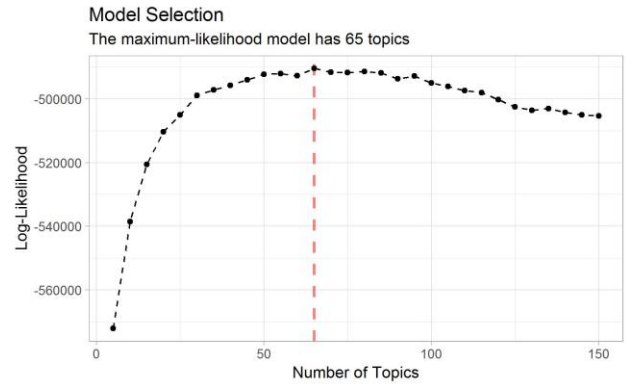

Figure X. Model Selection based on log-likelihood.

4.3 Warnings

We fit a predictive model for grade for each of the 132 courses currently offered at the college that have had more than 20 student enrollments since 2008. The model is a lasso-regularized linear regression model (Tibshirani, 1996). 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 GPA and their concentration-specific GPA (humanities, natural sciences, social science, skills and projects). Students' topic expertise consists of a set of 35 variables (one per topic of the topic model) which indicate how much knowledge of the topic the student has acquired through his curriculum. Topic expertise can be regarded as an approximation of the skills acquired by the student. Concretely, 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. The assumption is that students who obtain 10/10 in a course have acquired all the topic-related knowledge present in the course while those obtaining 5/10 have only acquired half of it. Figure X shows 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 to avoid overfitting. We use the lasso penalty to shrink the coefficient estimates (Tibshirani, 1996). 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 (Hastie et al., 2009). Figure X 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 (1.74 grade point). The mean CV error weighted by the number of students enrolled in the course is 0.77, the median is 0.77 and the standard deviation is 0.28.

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

Course	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Course 1	0.0	0.4	0.0	0.4	0.2
Course 2	0.2	0.2	0.2	0.2	0.2
Course 3	0.6	0.3	0.0	0.1	0.0

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	0.240	0.00	0.240	0.12
Course 2	0.18	0.180	0.18	0.180	0.18
Course 3	0.15	0.075	0.00	0.025	0.00

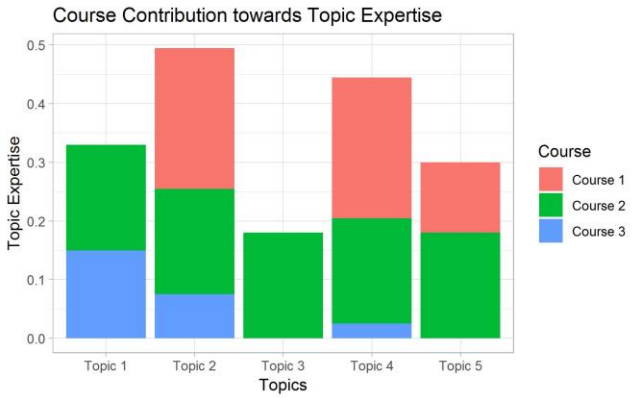

Figure X. Toy example: course contribution to a student's topic expertise. We use these variables to predict grade.

To receive a warning, the user enters into the system her/his student ID and a list of courses that she/he is considering for the coming term. The system uses the student ID to extract the student's transcript, from which her/his past academic performance and topic expertise are established. The predictive models of the selected courses then use these variables to predict the grades the student will receive if she/he enrolls in the selected courses. A warning is issued if the model predicts a fail grade.

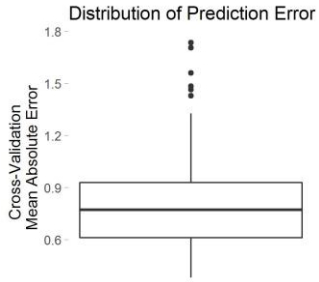

Figure X. Distribution of the cross-validation mean absolute error in the 148 predictive models.

4.3.1 Rule-based Warnings (Exclude to gain space?)

We investigated an alternative approach for warnings based on association rules. We used the CSPADE algorithm (REF) to identify sequences in the students transcripts of the type <fail course A> => <fail course B> and <not take course A> => <fail course B> and considered rules 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 indicate that they considered taking a course for which one the selected rules indicates that she/he is likely to fail it.

Although this approach is very transparent, which motivated its initial adoption, it turned out to be unsuitable to our case. First, given the small size of our sample 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 can be acquired in several 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>. This increased the number of rules meeting the criteria to 185. Yet, the second issue remained and led us to consider a regressive predicting model that uses topic expertise as a proxy for skills that a necessary to perform well in a course.

4.4 Preparatory Coursework

In order to be transparent and help student design their curriculum, each warning is accompanied by a list of preparatory course work. Similarly to the warnings, we fit a lasso-regularized predictive linear regression model for each course which, this time, takes as input students' topic expertise. Large positive coefficient estimates indicates that of good knowledge of the associated topics is associated with a larger grade in the course. For each course, the preparatory coursework consists of the 5 classes whose topic distribution has the smallest KL distance to the course's predictive model's normalized coefficient estimate

4.5 Course Recommendation

To provide Course Recommendations we identify courses whose content best match the academic interests of the students. A student's academic interest profile consists of a numeric vector indicating the importance of each topic for the student. This vector is normalized so we can treat it as a probability distribution over topics and use the KL-distance to find the closest courses to this distribution. To generate the interest topic profile, we ask the student to select key words and to indicate whether to use their transcript to approximate their interest. Then, a probability distribution over topics is created by adding the contribution of all words to each topic and normalizing across all topics. Optionally, if a student wishes to use their transcript, the student's topic profile as defined for the grade predictive model is added to the key words before normalization.

Then KL-distance is used to extract the *n*-closest courses to this distribution over topics and suggest them as courses of interest. To make the system informative we include the key words that led to each recommendation. This is done by calculating the total contribution of each key word for a course, ranking them and displaying the first three with the course recommendation.

5. RESULTS

6. FUTURE WORK

7. ACKNOWLEDGMENTS

Our thanks to the University College Maastricht, the Institute of Data Science, Maastricht University, and the Department of Data Science and Knowledge Engineering, Maastricht University, in particular to Evgueni Smirnov, and Peter Vermeer for supporting this project with their expertise.

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

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

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

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

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

[10] 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.

[11] 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/.

[12] 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.

[13] Surpatean, A., Smirnov, E., & Manie, N. (2012). Master orientation tool. In *Proceedings of the 20th European Conference on Artificial Intelligence* (pp. 995-996). IOS Press.

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

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